# Effectively Closed Sets
# $\Pi^0_1$ Classes
# DRAFT May 2017


Douglas Cenzer
Department of Mathematics, University of Florida
Gainesville, FL 32605-8105, USA
email: cenzer ufl.edu

Jeffrey B. Remmel
Department of Mathematics
University of California, San Diego


May 3, 2017

# Contents

# Preface

Effectively closed sets have been a central theme in computability theory, algorithmic randomness and applications to computability and effectiveness in mathematics. This book is intended to be a self-contained introduction to the theory and applications of effectively closed sets, or $\Pi^0_1$ classes. It may be used for a graduate-level course and also as reference for researchers in computability theory and related areas.

Part A begins with some basic facts from computability theory which will be needed. The members of a $\Pi^0_1$ class are real numbers, often represented by infinite strings of natural numbers, or by sets of natural numbers. Background is taken from the classic book of Soare [181] on computably enumerable (c.e.) sets and degrees. The fundamental problem, going back to work of Kleene [97] in the period 1940-1960, is to determine the complexity of the members of a $\Pi^0_1$ class, as measured by the Turing degree, or by the definition in the hyperarithmetic hierarchy, or by the amount of resources in time and space required. The Kleene basis theorem showed that every $\Pi^0_1$ class contains a member which is recursive in some $\Sigma^1_1$ set and the Kreisel-Shoenfield basis theorem [173], which showed that every c. b. $\Pi^0_1$ class contains a member of degree $< 0'$. Two fundamental papers in this area are [91, 90] by Jockusch and Soare. They show, among other things, that there is a $\Pi^0_1$ class with no recursive members and such that any two members have mutually incomparable Turing degree.

The Cantor-Bendixson derivative which reduces a closed set to its perfect kernel, plays an important role here going back to the 1959 paper of Kreisel [105], who first noticed that the degree of a member $x$ of a $\Pi^0_1$ class is related to the Cantor-Bendixson rank of $x$ in $P$ and that any countable class has a computable member. Countable $\Pi^0_1$ classes were closely examined by Soare and others [19, 42] in the 1980's. $\Pi^0_1$ classes are given an enumeration as $P_0, P_1, \dots$ and index sets for families of $\Pi^0_1$ classes are then studied in the manner that index sets for c.e. sets are studied in [181]. These can measure the complexity of certain properties of $\Pi^0_1$ classes, related in particular to cardinality and measure. $\Pi^0_1$ classes may be defined as sets of infinite paths through computable trees.

Part B presents some applications of $\Pi^0_1$ classes in logic, mathematics and theoretical computer science. The solution sets of many mathematical problems may be represented by $\Pi^0_1$ classes and the complexity of the problem can then be determined. The more difficult representation problem is to show that every $\Pi^0_1$ class (or every bounded or c. b. $\Pi^0_1$ class) can represent the solution set of a certain problem. For example, in 1960, Shoenfield [174] showed that the family of complete consistent extensions of an axiomatizable theory is a c. b. $\Pi^0_1$ class and Ehrenfeucht [63] showed that any c. b. $\Pi^0_1$ class can represent such a family.

The family of complete consistent extensions of an axiomatizable theory is of course closely related to the Lindenbaum algebra of the theory and Boolean algebras are an important topic for $\Pi^0_1$ classes. A number of articles in the area use the notion of a computably enumerable ideal of the computable dense Boolean algebra as an equivalent notion to that of a $\Pi^0_1$ class. This concept will be discussed in detail in the section on Boolean algebras.

Non-monotonic logic [122] is a general form of reasoning where certain "default" assumptions are made and may later be rescinded. The set of stable models of a logic program is a non-monotonic generalization of the (unique) closure under consequence of a set of axioms and rules. Different versions of a logic program may be used to represent c. b., bounded and unbounded $\Pi^0_1$ classes. Another area of theoretical computer science where $\Pi^0_1$ classes have application is the study of $\omega$-languages. This refers to a sets of infinite words which is *accepted*, in some fashion, by a program.

The surjective matching problem of Philip and Marshall Hall [76] was analyzed by Manaster and Rosenstein, who showed that the set of bijective matchings in a symmetrically highly recursive society is always a c. b. $\Pi^0_1$ class, and can represent an arbitrary c. b. $\Pi^0_1$ class. Bean [7] showed in 1976 that the family of $k$-colorings of a highly computable graph is a c. b. $\Pi^0_1$ class and Remmel [161] showed that any c. b. $\Pi^0_1$ class can represent, up to a permutation of the colors, such a family.

The reason that $\Pi^0_1$ classes arise so naturally in the study of recursive combinatorics is that many combinatorial theorems about finite graphs and partially ordered sets (posets) can be extended to countably infinite graphs and posets by applying König's Lemma, which states that every infinite finitely branching tree $T$ has an infinite path through it. Now König' Lemma, and also the so-called Weak König's Lemma play an important role in the Reverse Mathematics program of Friedman and Simpson [176]. Thus the study of $\Pi^0_1$ classes can be related to the study of König's Lemma. For example, Simpson [176] showed that Lindenbaum's lemma (that every countable consistent set of sentences has a complete consistent extension) and Gödel's completeness theorem are both equivalent to Weak König's Lemma over a certain subsystem ($RCA_0$) of second order arithmetic. For another example, Hirst [81] showed that a version of Hall's symmetric matching theorem is equivalent to König's Lemma over $RCA_0$.

The role of $\Pi^0_1$ classes in computable algebra and computable analysis is also presented.

Part C examines recent results on the family of $\Pi^0_1$ classes. One very important topic is the connection between effectively closed sets and algorithmic randomness, as developed by many researchers from Kucera [106, 107, 108] to Lewis [1, 6, 5] and surveyed in the books of Downey-Hirschfeldt [58] and Nies [150]. The lattice $\mathcal{E}_\Pi$ of $\Pi^0_1$ classes under inclusion is compared and contrasted with the lattice $\mathcal{E}$ of c.e. sets under inclusion. This includes results of Downey and others [22, 46, 45] on thin classes and automorphisms and work of Cenzer and Nies [28, 29] on intervals and on definability in $\mathcal{E}_\Pi$. The degree of difficulty of a class was defined by Medvedev [136] and refers to the difficulty of finding a member of the class. The Medvedev lattice of degrees of difficulty was studied later by Sorbi [183] and then the study of the Medvedev and also the related Muchnik degrees of $\Pi^0_1$ classes was developed further by Simpson [177] and others. Here we also examine $\Pi^0_1$ classes which arise from trees with a specified complexity, such as polynomial time computable.

## Acknowledgements

The authors owe a great debt to their advisors, Peter Hinman and Anil Nerode, to the founders of the subject of computability theory, in particular Stephen Kleene, and also to Carl Jockusch and Robert Soare, whose seminal papers established the subject of $\Pi^0_1$ classes back in the 1970's. We would also like to thank some others for enlightening discussions on the subject over many years, including Peter Cholak, Peter Clote, Rod Downey, Andre Nies, Richard Shore, Stan Wainer, Rebecca Weber. The first author would like to thank several people for their careful reading of drafts. This includes Chris Alfeld, Paul Brodhead, Ali Dashti, Sergey Melikhov, Diego Rojas, Rebecca Smith and Ferit Toska.

# Part A

# Computability Theory and $\Pi^0_1$ Classes

# Chapter 1

# Background

This chapter contains some of the definitions and notations needed for the study of effectively closed sets. We begin with objects under study: numbers, functions, sequences (or strings) and trees.

The set $\{0, 1, 2, ...\}$ of natural numbers is denoted by $\mathbb{N}$ and also by $\omega$ when we view $\mathbb{N}$ as an ordered set. Here $n = \{0, 1, \ldots, n-1\}$ is identified with the set of smaller natural numbers. Lower-case Latin letters $a, b, c, d, e, i, j, k, l, m, n$ denote integers; $p, q, r, s, t$ denote rational numbers; $u, v, w, x, y, z$ denote real numbers. The letters $f, g, h$ (and occasionally other lower-case Latin letters) denote *total* functions from $\mathbb{N}^k$ to $\mathbb{N}$ for $k \geq 1$; the Greek letters $\phi, \psi, \theta$ (and occasionally other lower-case Greek letters) denote (possibly) *partial* functions on $\mathbb{N}^k$ (functions whose domain is a subset of $\mathbb{N}^k$ for some $k$). Lower case Greek letters $\rho, \sigma, \tau, \nu$ denote finite sequences of natural numbers; $\alpha, \beta, \delta, \gamma$ denote ordinals. Upper-case Latin letters $A, B, C, D, E, I, J, K, L, M$ denote subsets of $\mathbb{N}$; $S, T$ denote trees; $P, Q, U, V, W, X, Y, Z$ denote sets of real numbers. Upper-case Latin letters $F, G, H$ denote *total* functions of real variables (with domain and range included in $\mathbb{N}^m \times \Re^n$); Upper-case Greek letters $\Phi, \Psi, \Theta$ (and occasionally others) denote (possibly) partial functions of real variables. In our usage, a *set* usually refers to a set of natural numbers.

The composition of two functions $f$ and $g$ is denoted by $f \circ g$; $f^n$ denotes the function $f$ composed with itself $n$ times. For a partial function $\phi$, $\phi(x) \downarrow$ denotes that $\phi(x)$ is defined and $\phi(x) \uparrow$ denotes that $\phi(x)$ is not defined. $dom(\phi) = \{x : \phi(x) \downarrow\}$ and $ran(\phi) = \{\phi(x) : x \in dom(\phi)\}$ denote the domain and range of $\phi$, respectively. If $F : X \to Y$, then $F[U]$ denotes $\{F(x) : x \in U\}$ for $U \subseteq X$ and $F^{-1}[V]$ denotes $\{x : F(x) \in V\}$ for $V \subseteq Y$. $\chi_A$ denotes the characteristic function of $A$, which is often identified with $A$ and written simply as $A(x)$. $\phi \lceil m$ denotes the restriction of $A$ to $x$.

For two sets $X$ and $Y$, $X \times Y$ denotes the direct product of $X$ and $Y$, that is, the set of ordered pairs $(x, y)$ with $x \in X$ and $y \in Y$. The direct product $X_1 \times X_2 \times \ldots \times X_k$ of a sequence $X_1, \ldots X_k$ of sets is similarly defined. $X^k$ is the product of $k$ copies of $X$.

The power $X^Y$ of two sets denotes the set of (total) functions with domain

$Y$ and range a subset of $X$. In particular, $\{0,1\}^{\mathbb{N}}$ is the usual Cantor space and may be identified with the family of subsets of $\mathbb{N}$. $\mathbb{N}^{\mathbb{N}}$ is the Baire space. $\Re$ denotes the space of real numbers. The Cantor space may be identified with a (compact) subset of $\Re$ and the Baire space may be identified with the set of irrational numbers. For us a *class* refers to a subset of $\Re$ (or of the Cantor space or Baire space). A class in the Cantor space may be called a "class of sets" since its elements are the characteristic functions of sets of natural numbers.

## 1.1 Trees

Let $\Sigma$ be a set of symbols (an *alphabet*), usually an initial segment of $\mathbb{N}$. Then for a natural number $n$, $\Sigma^n$ denotes the set of strings $\sigma = (\sigma(0), \sigma(1), \ldots, \sigma(n-1))$ of $n$ letters from $\Sigma$; the length $n$ of $\sigma$ is denoted by $|\sigma|$. The empty string has length 0 and will be denoted by $\emptyset$. $\Sigma^*$ (or sometimes $\Sigma^{<\omega}$) denotes the set $\cup_{n\in\omega}\Sigma^n$ and $\Sigma^\omega$ denotes the set of infinite sequences. Strings may be coded by natural numbers in the usual fashion. First let $[x, y]$ denote the standard pairing function $\frac{1}{2}(x^2 + 2xy + y^2 + 3x + y)$ and in general $[x_0, x_1, \ldots, x_n] = [[x_0, \ldots, x_{n-1}], x_n]$. Then we can code strings of arbitrary length $n > 0$ by $\langle\sigma\rangle = [n, [\sigma(0), \sigma(1), \ldots, \sigma(n-1)]]$ and also $\langle\emptyset\rangle = 1$. A string may be identified with its code, so that functions on $\mathbb{N}^*$ are represented by functions on $\mathbb{N}$. A constant string $\sigma$ of length $n$ will be denoted $k^n$. For $m < |\sigma|$, $\sigma \upharpoonright m$ is the string $(\sigma(0), \ldots, \sigma(m-1))$; $\sigma$ is an *initial segment* of $\tau$ (written $\sigma \prec \tau$) if $\sigma = \tau | m$ for some m. Initial segments are also referred to as *prefixes*. Similarly $\tau$ is said to be a *suffix* of $\sigma$ if $|\tau| \leq |\sigma|$ and, for all $i < |\tau|$, $\sigma(|\sigma| - |\tau| + i) = \tau(i)$. The *concatenation* $\sigma^\frown\tau$ (or sometimes $\sigma * \tau$ or just $\sigma\tau$) is defined by $\sigma^\frown\tau = (\sigma(0), \sigma(1), \ldots, \sigma(m-1), \tau(0), \tau(1), \ldots, \tau(n-1))$, where $|\sigma| = m$ and $|\tau| = n$; in particular we write $\sigma^\frown a$ for $\sigma^\frown(a)$ and $a^\frown\sigma$ for $(a)^\frown\sigma$. Thus we may also say that $\sigma$ is a prefix of $\tau$ if and only if $\tau = \sigma^\frown\rho$ for some $\rho$ and that $\tau$ is a suffix of $\sigma$ if and only if $\sigma = \rho^\frown\tau$ for some $\rho$.

For any $x \in \Sigma^*$ and any finite $n$, the *initial segment* $x\lceil n$ of $x$ is $(x(0), \ldots, x(n-1))$. We write $\sigma \preceq x$ if $\sigma = x\lceil n$ for some $n$. For any $\sigma \in \Sigma^n$ and any $x \in \Sigma^*$, we have $\sigma^\frown x = (\sigma(0), \ldots, \sigma(n-1), x(0), x(1), \ldots)$.

For a sequence $a_0 < a_1 < \cdots < a_n$, we denote by $\lfloor a_0, \ldots, a_n \rfloor$ the string $\sigma \in \{0,1\}^{a_n}$ such that $\sigma(k) = 1$ if and only if $k = a_i$ for some $i < n$. Thus $\lfloor a_0, a_1, \ldots, a_n \rfloor = 0^{a_0}10^{a_1-a_0-1}1 \cdots 0^{a_{n-1}-a_{n-2}-1}10^{a_n-a_{n-1}-1}$.

For any $x, y \in \mathbb{N}^{\mathbb{N}}$, the *join* $x \oplus y = z$, where $z(2n) = x(n)$ and $z(2n+1) = y(n)$. For two classes $P$ and $Q$, the product $P \otimes Q = \{x \oplus y : x \in P \ \&\ y \in Q\}$. An infinite sequence $x_0, x_1, \ldots$ may be coded as $\langle x_0, x_1, \ldots\rangle = y$, where $y(\langle m, n\rangle) = x_m(n)$. For an infinite family $\{P_i : i \in \omega\}$ of sets, the product may then be defined as $\{\langle x_0, x_1, \ldots\rangle : (\forall i) x_i \in P_i\}$. We can also define the disjoint union $P \oplus Q = \{0^\frown x : x \in P\} \cup \{1^\frown y : y \in Q\}$.

A *tree* $T$ over $\Sigma$ is a set of finite strings from $\Sigma^*$ which is closed under initial segments. The set $\Sigma$ is sometimes called an *alphabet*. We say that $\tau \in T$ is an *immediate successor* of a string $\sigma \in T$ if $\tau = \sigma^\frown a$ for some $a \in \Sigma$. Since our alphabet will always be countable and effective, we may assume that $T \subseteq \mathbb{N}^*$.

For any tree $T$ and any $\sigma$, $T(\sigma) = \{\tau : \sigma \preceq \tau \text{ or } \tau \preceq \sigma\}$.

A tree $T$ is said to be a *shift* if it is also closed under suffixes.

**Example 1.1.1.** *Define $T \subset \{0,1\}^*$ so that $\sigma \in T$ if and only if $\sigma$ does not have 3 consecutive 0's, that is, if $\sigma$ has no consecutive substring of the form (000). Clearly if $\sigma$ does not have 3 consecutive 0's then no initial segment of $\sigma$ can have 3 consecutive 0's either. Furthermore, if $\sigma$ has no consecutive substring* (000)*, then no suffix of $\sigma$ can have a consecutive substring* (000)*. Thus $T$ is a shift.*

We say that a tree $T$ is *finite-branching* if for every $\sigma \in T$, there are only finitely many immediate successors of $\sigma$ in $T$. Certainly any tree $T$ over a finite alphabet is finite-branching.

**Example 1.1.2.** *Define the tree $T \subset \mathbb{N}^*$ so that for strings $\sigma$ of length $n$, $\sigma \in T \iff \sigma(n-1) \leq 1 + \sigma(0) + \sigma(1) + \ldots \sigma(n-2)$. Then for any $\sigma \in T$, $\sigma(0) \leq 1$, $\sigma(1) \leq 2$, and by induction $\sigma(n) \leq 2^n$; it follows that $\sigma$ can have at most $2^n$ immediate successors.*

We will see later that a tree $T$ is finite-branching if and only if there is a function $f$ such that for all strings $\sigma \in T$ of length $n$, $\sigma$ has at most $f(n)$ immediate successors. The problem of computing the function $f$ will be a very important one. More generally, we will look at the problems of computing list of these successors, or an upper bound on the size of the successors, or an upper bound on the number of successors.

## 1.2 Topology and Measure

The topology of the real line has a basis of open intervals $(x, y) = \{u : x < u < y\}$ where $x = -\infty$ and $y = \infty$ are allowed; $[x, y]$ denotes the closed interval $\{u : x \leq u \leq y\}$; $[x, y)$ and $(x, y]$ are similarly defined. The topology on the spaces $\Sigma^{\mathbb{N}}$, where $\Sigma$ is either a finite alphabet or equals $\mathbb{N}$, is determined by a basis of intervals $I(\sigma) = \{x : \sigma \prec x\}$ and has a sub-basis of sets of the form $\{x : x(m) = n\}$ for fixed $m, n$. Notice that each interval is also a closed set and is therefore said to be *clopen* and that the clopen subsets of the Cantor space $\{0,1\}^{\mathbb{N}}$ are just the finite unions of intervals.

For a tree $T \subseteq \Sigma^*$, we define the set $[T]$ of infinite paths through $T$ by letting

$$x \in [T] \iff (\forall n) x \upharpoonright n \in T.$$

A subset $P$ of $\mathbb{N}^{\mathbb{N}}$ is closed if and only if $P = [T]$ for some tree $T$. This justifies the description of a $\Pi^0_1$ class as an effectively closed subset of $\mathbb{N}^{\mathbb{N}}$. A function $F : X \to Y$ is *continuous* if $F^{-1}[V]$ is open for every open set $V \subseteq Y$. Then a function $F : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ is continuous if, for all $m, n$, $\{x : F(x)(m) = n\}$ is open.

Let $X$ be either $\Re$, $\mathbb{N}^{\mathbb{N}}$ or $\{0,1\}^{\mathbb{N}}$. A subset $Y$ of $X$ is *dense* in an interval $I$ if it meets every subinterval of $I$; $Y$ is *nowhere dense* if it is dense in no interval.

$Y$ is *meager (first category)* if it is a countable union of nowhere dense sets; $Y$ is *non-meager (second category)* if it is not meager. $Y$ is *comeager (residual)* if $\overline{Y}$ is meager.

An element $x \in Y$ is *isolated* in $Y$ if there exists an open set $U$ such that $Y \cap U = \{x\}$. A closed, non-empty set $Y$ is *perfect* if it has no isolated elements. Each of the spaces $\Re$, $\mathbb{N}^{\mathbb{N}}$ and $\{0,1\}^{\mathbb{N}}$ are perfect.

**Definition 1.2.1.** *The* Cantor-Bendixson derivative $D(P)$ *of a compact set* $P$ *is the set of nonisolated points in* $P$.

Note that $D(P)$ is empty if and only if $P$ is finite.

The iterated Cantor-Bendixson derivative $D^\alpha(P)$ of a closed set $P$ is defined for all ordinals $\alpha$ by the following transfinite induction.

$D^0(P) = P$; $D^{\alpha+1}(P) = D(D^\alpha(P))$ for any $\alpha$; $D^\lambda(P) = \bigcap_{\alpha<\lambda} D^\alpha(P)$ for any limit ordinal $\lambda$.

The *Cantor-Bendixson (C.B.) rank* of a closed set $P$ is the least ordinal $\alpha$ such that $D^{\alpha+1}(P) = D^\alpha(P)$. If $\alpha$ is the C-B rank of $P$, then $D^\alpha(P)$ is the *perfect kernel* of $P$ and is a perfect closed set. For an element $x \in P$ which is not in the perfect kernel, the *Cantor-Bendixson (C.B.) rank* of $x$ in $P$ is the least ordinal $\alpha$ such that $x \notin D^{\alpha+1}(P)$.

The standard Lebesgue measure $\mu$ on $\{0,1\}^\omega$ is determined by letting $\mu(I(\sigma)) = 2^{-|\sigma|}$. A product measure on $\mathbb{N}^{\mathbb{N}}$ may be defined (with $\lambda(\mathbb{N}^{\mathbb{N}}) = 1$) by setting the measure of $\{x : x(m) = n\}$ to be $2^{-n-1}$, so that $I(\sigma)$ has measure $2^{-(m_0+m_1+\cdots+m_{k-1}+k)}$.

## 1.3 Structures

We shall use the logical symbols &, $\vee$, $\neg$, $\rightarrow$ and $\iff$ to denote as usual "and", "or", "not", "implies" and "if and only if". The symbols $\exists$ and $\forall$ denote the quantifiers "there exists" and "for all". In addition, $(\exists m < p)$ and $(\forall m < p)$ denote *bounded quantifiers* where the range of the quantifier is restricted to numbers less than $p$, and $(\exists^\infty x)$ denotes "there exist infinitely many $x$ such that".

As usual, a *first-order language* $\mathcal{L}$ is given by a set $\{R_i\}_{i\in S}$ of relation symbols, a set $\{f_j\}_{j\in T}$ of function symbols, and a set $\{c_i\}_{i\in U}$ of constant symbols, together with functions $m(i)$ and $n(i)$ such that $R_i$ is an $m(i)$-ary relation symbol and $f_i$ is an $n(i)$-ary function symbol. We assume here that $S$, $T$ and $U$ are subsets of $\omega$. The language also includes variables and both existential and universal quantifiers using these variables. The set of terms of $\mathcal{L}$ and the set $Sent(\mathcal{L})$ of sentences of $\mathcal{L}$ are defined as usual by induction. A propositional language is given by a set of 0-ary relation symbols, or propositional variables. The reader is referred to Shoenfield [175] for details.

We shall consider structures over an effective first-order language

$$\mathcal{L} = \langle \{R_i^{m(i)}\}_{i\in S}, \{f_i^{n(i)}\}_{i\in T}, \{c_i\}_{i\in U} \rangle,$$

where S, T and U are initial segments of $\omega$, for all $i \in U$, $c_i$ is a constant symbol and there are partial recursive functions $s$ and $t$ such that, for all $i \in S$, $R_i$ is an $s(i)$-ary relation symbol and, for all $i \in T$, $f_i$ is a $t(i)$-ary function symbol.

Let $\Gamma$ be some complexity class of sets (and functions), such as partial recursive, primitive recursive, exponential time, polynomial time (or p-time). We say that a set or function is $\Gamma$-computable if it is in $\Gamma$.

A *model* or *structure*, $\mathcal{A} = (A, \{R_i^{\mathcal{A}}\}_{i\in S}, \{f_i^{\mathcal{A}}\}_{i\in T}, \{c_i^{\mathcal{A}}\}_{i\in U})$, for the language $\mathcal{L}$ is given by a set $A$ together with interpretations of the relation, function and constant symbols.

**Definition 1.3.1.** *(a) A structure (where the universe $A$ of $\mathcal{A}$ is a subset of $\Sigma^*$) is a $\Gamma$-structure if*

*(i) $A$ is a $\Gamma$-computable subset of $\Sigma^*$*

*(ii) for each $i \in S$, $R_i^{\mathcal{A}}$ is a $\Gamma$-computable relation on $A^{m(i)}$.*

*(iii) for each $j \in T$, $f_j^{\mathcal{A}}$ is a $\Gamma$-computable function from $A^{n(j)}$ into $A$.*

*(iv) If $S = \omega$, then there is a $\Gamma$-computable relation $R$ such that, for all $i \in S$ and all $(x_0, \dots, x_{m(i)})$,*

$$R_i^{\mathcal{A}}(x_0, \dots, x_{m(i)}) \iff R(i, \langle x_0, \dots, x_{m(i)}\rangle).$$

*(v) If $T = \omega$, then there is a $\Gamma$-computable function $f$ such that, for all $j \in T$ and all $(x_0, \dots, x_{n(j)})$,*

$$f_i^{\mathcal{A}}(x_0, \dots, x_{n(j)}) = f(i, \langle x_0, \dots, x_{n(j)}\rangle).$$

For any complexity class $\Gamma$, we say that two structures $\mathcal{A}$ and $\mathcal{B}$ are $\Gamma$-isomorphic if there is an isomorphism $f$ from $\mathcal{A}$ onto $\mathcal{B}$ and $\Gamma$-computable functions F and G such that $f = F\lceil A$ (the restriction of $F$ to $A$) and $f^{-1} = G\lceil B$.

## 1.4 Orderings and Ordinals

The results of this book are all theorems of Zermelo-Fraenkel Set Theory with the Axiom of Choice. The (Generalized) Continuum is not assumed.

Our set-theoretic conventions are standard and we refer the reader to (for example) Jech [83] for further background. The inclusion relation $X \subseteq Y$ denotes $(\forall x)(x \in X \to x \in Y)$ and $X \subset Y$ denotes $X \subseteq Y$ and $X \neq Y$. The symbols $\cup$, $\cap$ and $\setminus$ denote the binary operations of union, intersection and difference; $\overline{A}$ denotes the complement of $A$.

A set $X$ is *transitive* if $(\forall y)(y \in X \to y \subseteq X)$ and $X$ is an *ordinal (number)* if $X$ and all of its elements are transitive. For ordinals $\alpha$ and $\beta$, $\alpha < \beta$ if and only if $\alpha \in \beta$. For any ordinal $\alpha$, $\alpha + 1 = \alpha \cup \{\alpha\}$ is the *successor ordinal* of $\alpha$. $\alpha$ is a *limit ordinal* if it is neither 0 nor a successor, which implies that $(\forall \beta < \alpha)(\beta + 1 < \alpha)$. For any set $X$ of ordinals, $\inf X$ denotes the least element of $X$ and $\sup X$ denotes the least ordinal greater than or equal to every element of $X$.

An ordinal $\alpha$ is said to be a *recursive ordinal* if there is a recursive well-ordering of $\omega$ of order type $\alpha$. The least non-recursive ordinal is denoted by $\omega_1^{\mathrm{C-K}}$, and was introduced by Church and Kleene [47].

The natural, or Hessenberg sum, $\alpha \oplus \beta$, of two ordinals $\alpha$ and $\beta$, may be defined as follows. Let $\alpha = \omega^{\gamma_1} a_1 + \omega^{\gamma_2} a_2 + \cdots + \omega^{\gamma_k} a_k$ and $\beta = \omega^{\gamma_1} b_1 + \omega^{\gamma_2} b_2 + \cdots + \omega^{\gamma_k} b_k$ be the Cantor normal forms of $\alpha$ and $\beta$, where we have inserted $a_i = 0$ and $b_j = 0$ to obtain expressions with the same powers of $\omega$. Then
$\alpha \oplus \beta = \omega^{\gamma_1}(a_1 + b_1) + \omega^{\gamma_2}(a_2 + b_2) + \cdots + \omega^{\gamma_k}(a_k + b_k)$.

Thus we treat ordinals as polynomials over $\omega$ with natural number coefficients. This natural addition is commutative. For any ordinals $\alpha$ and $\beta$, $\alpha + \beta \le \alpha \oplus \beta$. See [110] (p. 253) for details.

An ordinal $\kappa$ is a *cardinal number* if there is no one-to-one correspondence between $\kappa$ and any $\alpha < \kappa$. It follows from the Axiom of Choice that for every set $X$, there is a unique cardinal $\kappa$ and a one-to-one correspondence between $X$ and $\kappa$; $\kappa$ is the *cardinality* ($Card(X)$)of $X$. The natural numbers are exactly the finite cardinals and $\omega$ is the least infinite cardinal. A set $X$ is *countable* if $Card(X) \le \omega$ and *countably infinite* if $Card(X) = \omega$. The infinite cardinal $\omega$ is also denoted by $\aleph_0$ and the least uncountable cardinal by $\aleph_1$.

For any set $X$, $\mathcal{P}(X)$ denotes the power set of $X$, the set of all subsets of $X$ and $2^\kappa$ denotes $Card(\mathcal{P}(\kappa)$. Since there is a one-to-one correspondence between $\mathcal{P}(\mathbb{N})$ and the continuum $\Re$, $Card(\Re) = 2^{\aleph_0}$.

A relation $R$ on a set $X$ is a subset of $X \times X$; the domain of $R$ is $dom(R) = \{x : (\exists y)(x,y) \in R\}$ and the range is $ran(R) = \{y : (\exists x)(x,y) \in R\}$. $R(x,y)$ and also $xRy$ are sometimes used in place of $(x,y) \in R$. $R$ is *reflexive* if $R(x,x)$ for all $x$ and is *irreflexive* if $\neg R(x,x)$ for all $x$. $R$ is *symmetric* if $R(x,y)$ implies $R(y,x)$ for all $x,y$ and is *antisymmetric* if $R(x,y)$ & $R(y,x)$ implies $y = x$ for all $x,y$. $R$ is *transitive* if $R(x,y)$ & $R(y,z))$ implies $R(x,z)$ for all $x,y,z$. $R$ is *total* or *connected* if $R(x,y) \ \vee \ R(y,x)$ for all $x,y$. $R$ is an *equivalence relation* if it is symmetric, reflexive and transitive.

$R$ is a *pre-partial-ordering* if it is reflexive and transitive. A pre-partial-ordering $R$ is a *pre-linear-ordering* if it is total. A pre-partial-(linear-)ordering is a partial (linear) ordering if it is antisymmetric.

$R$ is is *well-founded* if every subset $A$ of $X$ has a *minimal element*, that is, some $m$ such that for all $x$, $R(x,m) \to R(m,x)$. Assuming the Axiom of Dependent Choice (DC), this is equivalent to the following

$$(\forall f \in \mathbb{N}^X)[(\forall m)(R(f(m+1), f(m)) \to (\exists m)R(f(m), f(m+1)).$$

A (pre-)linear ordering is a *(pre-)well-ordering* if it is well-founded.

In this chapter, we present some basic definitions and results from classical computability theory which are needed for the study of $\Pi_1^0$ classes. The key notion here is that of a computable functional, or function with domain a subset of $\mathbb{N}^\mathbb{N}$.

We begin with a brief review of computable functions and computably enumerable (c. e.) sets. Formal definitions of the set of computable functions have been given in many different ways. The computable functions are the functions mapping natural numbers (or more generally finite strings of symbols taken from a finite alphabet) which are computable by a Turing machine, register machine, or other idealized computer. These are the functions which can be computed by a program in Maple, or Matlab, or some other fixed programming language. The set of computable functions is the smallest which includes certain basic functions and is closed under primitive recursion, composition, and unbounded search.

All of these approaches are known to lead to the same family of functions, and Church's Thesis proclaims that any other attempt to formalize the notion of a computable function will lead to the same family of functions.

We refer the reader to Soare [181] and to Odifreddi [151] for full details on the basic definitions and results of computability theory.

## 1.5 Formal definitions of the computable functions

Since index sets will be a central topic in our work, we will give a definition in the spirit of Kleene [99] and Hinman [80] based on the index or code for a computable function. We will give the general definition for a computable function or functional with both natural number inputs and real number inputs (that is, functions from $\mathbb{N}^{\mathbb{N}}$). It is crucial that our functions may be *partial*, that is, defined on a proper subset of $\mathbb{N}^k \times (\mathbb{N}^{\mathbb{N}})^l$. The second crucial observation is that the (partial) computable functions may be enumerated as $\Phi_0, \Phi_1, \ldots$ so that the *universal* function $U(e, \overrightarrow{m}, \overrightarrow{x}) = \Phi_e(\overrightarrow{m}, \overrightarrow{x})$ is itself partial computable.

An index $e = \langle i, k, \ell, \ldots \rangle$ for a computable function is the code for a function $\Phi_e$ of $k$ natural numbers and $\ell$ real numbers. $\Phi_e$ is a function on natural numbers if $\ell = 0$ and will then be denoted also by $\phi_e$. Here $\overrightarrow{m} = (m_0, \ldots, m_{k-1})$ and $\overrightarrow{x} = (x_0, \ldots, x_{\ell-1})$.

The basic indices and functions are the following:

**(0)** Constant Functions: $\Phi_e(\overrightarrow{m}, \overrightarrow{x}) = n$ when $e = \langle 0, k, \ell, n \rangle$.

**(1)** Projection Functions: $\Phi_e(\overrightarrow{m}, \overrightarrow{x}) = m_i$ when $e = \langle 1, k, \ell, i \rangle$ and $i < k$.

**(2)** Successor Functions: $\Phi_e(\overrightarrow{m}, \overrightarrow{x}) = m_i + 1$ when $e = \langle 2, k, \ell, i \rangle$ and $i < k$.

**(3)** Application Functions: $\Phi_e(\overrightarrow{m}, \overrightarrow{x}) = x_j(m_i)$ when $e = \langle 3, k, \ell, i, j \rangle$, $i < k$ and $j < \ell$.

The primitive recursive functions are obtained from the basic functions by closure under composition and primitive recursion, which are defined as follows.

**(4)** Composition: $\Phi_e(\overrightarrow{m}, \overrightarrow{x}) = \Phi_a(\Phi_{b_1}(\overrightarrow{m}, \overrightarrow{x}), \ldots, \Phi_{b_r}(\overrightarrow{m}, \overrightarrow{x}))$ when $e = \langle 4, k, \ell, a, b_1, \ldots, b_r \rangle$ when $(a)_1 = r$, $(a)_2 = 0$ and, for each $t$, $(b_t)_1 = k$ and $(b_t)_2 = \ell$.

**(5)** Primitive Recursion: $\Phi_e(0, \overrightarrow{m}, \overrightarrow{x}) = \Phi_a(\overrightarrow{m}, \overrightarrow{x})$ and, for each $n$, $\Phi_e(n + 1, \overrightarrow{m}, \overrightarrow{x}) = \Phi_b(\Phi_e(n, \overrightarrow{m}, \overrightarrow{x}), n, \overrightarrow{m}, \overrightarrow{x}))$ when $e = \langle 5, k{+}1, \ell, a, b \rangle$, $(a)_1 = k$, $(a)_2 = \ell$, $(b)_1 = k + 2$ and $(b)_2 = \ell$.

A *set* $A \subseteq \mathbb{N}^k$ is primitive recursive if the characteristic function is primitive recursive. It is worth noting that the set of indices for primitive recursive functions is itself a primitive recursive set. Thus we may define an enumeration $\Pi_e$ of the primitive recursive functions by letting $\Pi_e(\overrightarrow{m}, \overrightarrow{x}) = \Phi_e(\overrightarrow{m}, \overrightarrow{x})$ if $e$ is a primitive recursive index and otherwise $\Pi_e(\overrightarrow{m}, \overrightarrow{x}) = 0$.

**Lemma 1.5.1.** *There is a partial recursive function $\pi$ such that for each $e$, $\Pi_e = \Phi_{\pi(e)}$.* $\square$

Details are left to the exercises.

The computable functions are obtained from the basic functions by closure under composition, primitive recursion and search, which is defined as follows. Here we let "(least $p)R(p)$" denote the least $p$ such that $R(p)$.

**(6)** Search: $\Phi_e(\overrightarrow{m}, \overrightarrow{x}) = (\text{least } p)\Phi_a(p, \overrightarrow{m}, \overrightarrow{x}) = 0$ when $e = \langle 6, k, \ell, a\rangle$, where this means as usual that $\Phi_e(\overrightarrow{m}, \overrightarrow{x}) = q$ if $\Phi_a(q, \overrightarrow{m}, \overrightarrow{x}) = 0$ and for all $p < q$, $\Phi_a(p, \overrightarrow{m}, \overrightarrow{x})$ is defined and not equal to zero.

If $\Phi_e(\overrightarrow{m}, \overrightarrow{x})$ is defined by the above, we say that $\Phi_e(\overrightarrow{m}, \overrightarrow{x})$ *converges* and write $\Phi_e(\overrightarrow{m}, \overrightarrow{x}) \downarrow$. If $\Phi_e(\overrightarrow{m}, \overrightarrow{x})$ is not determined by this definition, then $\Phi_e(\overrightarrow{m}, \overrightarrow{x})$ is undefined. We say that $\Phi_e(\overrightarrow{m}, \overrightarrow{x})$ *diverges* and write $\Phi_e(\overrightarrow{m}, \overrightarrow{x}) \uparrow$. If $e$ is not an index of a computable function, then of course $\Phi_e(\overrightarrow{m}, \overrightarrow{x}) \uparrow$ for all $\overrightarrow{m}, \overrightarrow{x}$, so that $\Phi_e$ is the empty function.

If we replace the real variables $x_j$ with finite sequences $\sigma_j$, then the definition of $\Phi_e(\overrightarrow{x}, \overrightarrow{\sigma})$ is obtained as above when we begin with $\Phi_e(\overrightarrow{m}, \overrightarrow{\sigma}) = \sigma_j(m_i)$ provided that $m_i < |\sigma_j|$.

Then the computation of $\Phi_e(\overrightarrow{m}, \overrightarrow{x}) = q$ is coded by $c = \langle e, \overrightarrow{m}, \overrightarrow{\sigma}, q\rangle$, where $\sigma_j$ is the shortest initial segment of $x_j$ needed.

We will next define the notions of a *computation tree* and a *derivation* for a computation.

For the constant, projection and successor functions, the computation tree of $\Phi_e(\overrightarrow{m}, \overrightarrow{x}) = n$ has a single node $\langle e, \overrightarrow{m}, \overrightarrow{\emptyset}, n\rangle$ and this is also the derivation.

For the application function, the computation tree for $\Phi_e(\overrightarrow{m}, \overrightarrow{x}) = x_i(m_j) = n$ also has a single node $\langle e, \overrightarrow{m}, \overrightarrow{\emptyset}, (x_i(0), \dots, x_i(m_j)), \overrightarrow{\emptyset}, n\rangle$ and this is the derivation.

The other cases are more complicated.

(4) Composition:

The computation tree for $\Phi_e(\overrightarrow{m}, \overrightarrow{x}) = \Phi_a(\Phi_{b_0}(\overrightarrow{m}, \overrightarrow{x}), \dots, \Phi_{b_{r-1}}(\overrightarrow{m}, \overrightarrow{x})) = q$ has a top node $c = \langle e, \overrightarrow{m}, \overrightarrow{\sigma}, q\rangle$ and has immediate predecessors $c_0, \dots, c_{r-1}, c'$, where $c_t$ is the top node of the computation tree for $\Phi_{b_t}(\overrightarrow{m}, \overrightarrow{x})$ for $t < r$, and $c'$ is the top node of the computation tree for $\Phi_a(\Phi_{b_0}(\overrightarrow{m}, \overrightarrow{x}), \dots, \Phi_{b_{r-1}}(\overrightarrow{m}, \overrightarrow{x}))$. For each $j$, $\sigma_j$ is the union of the initial segments of $x_j$ used in $c_t$. The derivation is $\langle d_1, \dots, d_{r-1}, d, c\rangle$ where $d_t$ is the derivation of $\Phi_{b_t}(\overrightarrow{m}, \overrightarrow{x})$ for $t < r$ and $d$ is the derivation of $\Phi_a(\Phi_{b_0}(\overrightarrow{m}, \overrightarrow{x}), \dots, \Phi_{b_{r-1}}(\overrightarrow{m}, \overrightarrow{x}))$.

(5) Primitive Recursion:

The computation tree for $\Phi_e(0, \overrightarrow{m}, \overrightarrow{x}) = \Phi_a(\overrightarrow{m}, \overrightarrow{x}) = q_0$ has top node $d_0 = \langle e, 0, \overrightarrow{m}, \overrightarrow{\sigma}, q_0\rangle$ with a single immediate predecessor $c_0 = \langle a, \overrightarrow{m}, \overrightarrow{\sigma}, q_0\rangle$. The derivation is $\langle c_0, d_0\rangle$.

The computation tree for $\Phi_e(n+1, \overrightarrow{m}, \overrightarrow{x}) = \Phi_b(\Phi_e(n, \overrightarrow{m}, \overrightarrow{x}), n, \overrightarrow{m}, \overrightarrow{x})) = q_{n+1}$ has top node $d_{n+1} = \langle e, n+1, \overrightarrow{m}, \overrightarrow{\sigma}, q_{n+1}\rangle$ with two immediate predecessors, the top node $d_n$ of the computation tree for $\Phi_e(n, \overrightarrow{m}, \overrightarrow{x})$ and the top node $c_n$ of the computation tree for $\Phi_b(q_n, n, \overrightarrow{m}, \overrightarrow{x}))$. For each $j$, $\sigma_j$ is the union of the initial segments of $x_j$ used in $c_n$ and in $d_n$. The derivation is $\langle d_n, c_n, d_{n+1}\rangle$.

(6) Search:

The computation tree for $q = \Phi_e(\overrightarrow{m}, \overrightarrow{x}) = (\text{least } p)\Phi_a(p, \overrightarrow{m}, \overrightarrow{x}) = 0$ has top node $d = \langle e, \overrightarrow{m}, \overrightarrow{\sigma}, q\rangle$ with immediate predecessors $c_0, \dots, c_p$ where $c_t$ is the top node of the computation tree for $\Phi_a(t, \overrightarrow{m}, \overrightarrow{x})$ for $t \leq p$. For each $j$, $\sigma_j$ is the union of the initial segments of $x_j$ used in $c_t$ for some $t \leq p$. The derivation is $\langle c_0, \dots, c_p, d\rangle$.

We will often write $\Phi_e^y(\overrightarrow{m}, \overrightarrow{x})$ for $\Phi_e(\overrightarrow{m}, \overrightarrow{x}, y)$ and refer to the function $\Phi_e^y$ as being computable from the *oracle* $y$.

**Lemma 1.5.2.** *The set of derivations is primitive recursive and, furthermore, the relation $T(e, \langle\overrightarrow{m}, \overrightarrow{\sigma}\rangle, d)$ which indicates that $d$ is the derivation of $\Phi_e(\overrightarrow{m}, \overrightarrow{\sigma})$ is also primitive recursive.*

*Sketch.* The set of derivations may be defined by course-of-values recursion using coding and decoding of finite sequences, all of which is primitive recursive. Then the values of $e$, $\overrightarrow{m}$, $\overrightarrow{\sigma}$ and $\Phi_e(\overrightarrow{m}, \overrightarrow{\sigma})$ can be obtained from the last entry of the finite sequence coded by the derivation $d$. See Chapter II of Hinman [80] for details. □

### 1.5.1 Turing machines

The classic Turing machine, defined by Alan Turing, provides a very useful approach to computable functions. It has a simple elegant format but nevertheless has a strength equal to any other model of computing.

Our model of the Turing machine will be as follows. Let $\Sigma_0$ be a finite alphabet, let $B$ denote the blank symbol (not included in $\Sigma_0$), and let $\Sigma = \Sigma_0 \cup \{B\}$. A Turing machine *tape* consists of a potentially infinite sequence of squares, on which symbols from the alphabet $\Sigma$ may be stored, and possibly erased or written over during a computation.

Each tape comes equipped with a *pointer* or *reading head*, which will be pointing at one of the entries during any step of a Turing machine computation. The entries on a tape are ordered as $a_0, a_1, \dots$ beginning with a leftmost square. Initially each reading head points at the leftmost square of its tape. Turing machine computations are based on two fundamental operations, the following. Say that the pointer on a tape is located over $a_i$. The Turing machine can replace the symbol $a_i$ with any other symbol. Then it can move from the current square to $a_{i+1}$ or to $a_{i-1}$ (if $i > 0$) or remain at the current square.

A Turing machine $M$ which defines a function $\varphi_M : \Sigma_0^k \to \Sigma$ for some finite $k$ will have $k$ *input tapes*, an *output tape*, and a fixed finite number $m$ of *work* or *scratch* tapes. The inputs $\sigma_0, \ldots, \sigma_{k-1}$ are written on the input tapes at the start of the computation and the other tapes are initially empty. We will assume that the input tapes are *read-only*, that is, $M$ does not ever write over any symbol on the input tapes and does not write any new symbols onto the empty squares of an input tape. The output tape is assumed to be *write-only*, that is, once a symbol is written onto the output tape, it cannot be changed.

The instructions for a Turing machine $M$ to compute the function $\varphi_M$ are given by a finite set $Q$ of *states*, including some *initial* state $s$ and a *halting state* $h$, together with a *transition function*

$$\delta_M : Q \times \Sigma^{k+m+1} \to Q \times \Sigma^{m+2} \times \{\leftarrow, \rightarrow, \vdash\}^{k+m+1}.$$

The state of the machine together with the symbols on the scanned squares, are used via the transition function to determine the operation of the machine as follows. Let the tapes be numbered so that tapes 0 through $k-1$ are the input tapes, tapes $k$ through $k+m-1$ are the scratch tapes, and tape $k+m$ is the output tape. Suppose that $M$ is in state $q$ and that, for each $i < k + m$, pointer on tape $i$ is scanning the symbol $a_i$. Let $\delta_M(s, a_0, \ldots, a_{k+m}) = (q', b_0, \ldots, b_{k+m}, X_0, \ldots, X_{k+m})$, where each $X_i \in \{\leftarrow, \rightarrow, \vdash\}$. Here we assume that, for $i < k$, $b_i = a_i$ and that, if $a_{k+m+1} \neq B$, then $b_{k+m+1} = a_{k+m+1}$. We also assume that if $b_{k+m} \neq B$, then $X_{k+m} =\rightarrow$ and otherwise $X_{k+m} =\vdash$. Then the symbol $a_i$ is replaced on tape $i$ by the symbol $b_i$. The pointer on tape $i$ moves right if $X =\rightarrow$, moves left if $X =\leftarrow$ and it is not the leftmost square which is being scanned, and otherwise remains pointing at the same square. Finally, the machine transitions into state $q'$. If $q' = h$, then the computation is finished and the *output* $\varphi_M(\sigma_0, \ldots, \sigma_{k-1})$ is the sequence of entries on the output tape. The *length* of the computation is the number of steps until the halting state is reached, if any, and also represents the amount of *time* used in the computation for the purpose of complexity theory. The amount of space used is the total number of squares on the work tapes which were ever written on during the computation.

**Example 1.5.3.** *Natural numbers are usually represented in reverse binary form, so that* 6 *is represented as* 011. *(This is due to having a leftmost square on each tape.) The function $\varphi(x) = x + 1$ may be computed by the following Turing machine $M$. $M$ has three states, $s$, $q$ and $h$ and just two tapes, the input tape and the output tape. The transition function has the following values.*

$$\begin{aligned}
\delta(s, 0, B) &= (q, 0, 1, \rightarrow)\\
\delta(s, 1, B) &= (s, 1, 0, \rightarrow)\\
\delta(s, B, B) &= (h, B, B, \vdash)\\
\delta(r, 0, B) &= (r, 0, B, \rightarrow)\\
\delta(r, 1, B) &= (r, 1, B, \rightarrow)\\
\delta(r, B, B) &= (h, B, B, \vdash)
\end{aligned}$$

*Here we omit any transition where the output tape is not scanning a blank square, since that situation cannot occur.*

*The computation* $\varphi(101) = 011$ *(that is,* $5+1 = 6$*) takes three steps, remaining in state* $s$ *after the first step, moving to state* $r$ *after the second step and finishing in the halting state* $h$ *after scanning the blank at the third step.*

Frequently, we use computations to test whether a given input $\sigma$ meets certain criteria, that is, belongs to some set $A$. Then our Turing machine $M$ might output *Yes* or *No* if the input does or does not meet the criteria, or $M$ might halt if $\sigma$ meets the criteria and not halt otherwise. In the first case, $M$ demonstrates that the set $A$ is computable, and in the second case, $M$ demonstrates that $A$ is computably enumerable.

**Example 1.5.4.** *Let* $A = \{\sigma \in \{0,1\}^* : (\exists n)\sigma(n) = 0 = \sigma(n+1)\}$. *We can show that A is computably enumerable with the following simple Turing machine. Here we do not need any work tapes or even an output tape.*

$$\begin{aligned}
\delta(s,0) &= (q,\rightarrow)\\
\delta(s,1) &= (s,\rightarrow)\\
\delta(s,B) &= (s,\rightarrow)\\
\delta(q,0) &= (h,\vdash)\\
\delta(q,1) &= (s,\rightarrow)\\
\delta(q,B) &= (s,\rightarrow)
\end{aligned}$$

*If the input string* $\sigma$ *is in* $A$ *and* $n$ *is the least such that* $\sigma(n) = \sigma(n+1) = 0$*, then the Turing machine takes* $n+1$ *steps to read through the first* $n+1$ *entries of* $\sigma$ *and then halts. If* $\sigma$ *is not in* $A$*, then the machine take* $|\sigma|+1$ *steps to read through* $\sigma$ *(without finding* $00$ *and find the blank at the end of* $\sigma$. *Then it simply continues to read blanks and thus never halts.*

**Example 1.5.5.** *Let* $A = \{0^n 1^n : n \in \mathbb{N}\}$. *We will give an informal description of a Turing machine* $M$ *which outputs* $Y$ *if* $\sigma \in A$ *and otherwise outputs* $N$. *The machine* $M$ *has one work tape where it copies the* $0$*s from the input tape until either a* $1$ *or a* $B$ *is read. The reading head on the work tape will be pointing to the final* $0$. *When a* $1$ *is read on the input tape,* $M$ *transitions to a new state and begins erasing the* $0$*s from the work tape. When a* $B$ *is now read in the input tape,* $M$ *checks to see whether there is a* $B$ *or a* $0$ *on the work tape. If it is a* $B$*, then* $\sigma$ *is accepted by writing* $Y$ *on the output tape. If it is a* $0$*, then* $\sigma$ *is rejected by writing* $N$ *on the output tape (in this case there are not enough* $1$*s to match the initial sequence of* $0$*s). If* $M$ *finds a* $0$ *after some sequence of* $1$*s, then again* $\sigma$ *is rejected. For the remaining case, if* $B$ *is read on the input tape after a sequence of* $0$*s but before any* $1$*s are read, then* $\sigma$ *is also rejected.*

### Exercises

1.5.1. Show that the set of primitive recursive indices is itself a primitive recursive set. (You may assume here that the coding functions mapping $(a_0, a_1, \ldots, a_n)$ to $a = \langle a_0, a_1, \ldots, a_n\rangle$ and the decoding functions $(a)_i = a_i$ are primitive recursive.)

1.5.2. Prove Lemma 1.5.1.

1.5.3. Show that the *universal* sequence $\{\Pi_e\}_{e\in\omega}$ is not uniformly primitive recursive, that is, the function $f$ defined by $f(e,m) = \Pi_e(m)$, is not itself primitive recursive.

## 1.6 Basic results

In this section, we state a number of results about computable functions which will be needed later. Most proofs are omitted; the reader is referred to Hinman [80] and Soare [181]. For simplicity of expression, we will generally write $\phi_e(m)$ for a function of $k$ variables rather than $\phi_e(m_1, \ldots, m_k)$ or $\phi_e(\overrightarrow{m})$. Thus the results given here apply to functions taking any number of variables.

**Lemma 1.6.1** (Padding Lemma)**.** *Each partial computable function $\phi_e$ has an infinite set of indices, and furthermore, there is a primitive recursive, one-to-one function $f$ such that, for all $e$ and $n$, $f(e,n)$ is an index for $\phi_e$.*

*Sketch.* Let $f(e,n)$ be an index for the function which first computes $\phi_e(m)$, then adds $n$ to the output, and finally subtracts $n$ from the output. □

**Theorem 1.6.2** (Normal Form Theorem)**.** *(Kleene) There is a primitive recursive predicate $T_1(e, \overrightarrow{m}, \overrightarrow{\sigma}, q)$ and a primitive recursive function $U$ such that*

$$\Phi_e(\overrightarrow{m}, \overrightarrow{x}) = U((\textit{least } q)\ T_1(e, \overrightarrow{m}, \overrightarrow{x}\lceil q, q))$$

*Sketch.* Let the $T$ predicate be given by Lemma 1.5.2 and define the predicate $T_1$ so that, for any $e, \overrightarrow{m}, \overrightarrow{\sigma}, q$, $T_1(e, \overrightarrow{m}, \overrightarrow{\sigma}, q)$ if and only if there exists initial segments $\tau_j$ of each $\sigma_j$ such that $T(e, \langle \overrightarrow{m}, \overrightarrow{\tau} \rangle, q)$ and $U$ outputs $\Phi_e(\overrightarrow{m}, \overrightarrow{\sigma})$ from the derivation $q$. □

**Theorem 1.6.3** (Enumeration Theorem)**.** *For any $k, \ell < \omega$, there is a partial computable function $\Phi$ such that, for all $e$, $\overrightarrow{m}$ and $\overrightarrow{x}$, $\Phi(e, \overrightarrow{m}, \overrightarrow{x}) = \Phi_e(\overrightarrow{m}, \overrightarrow{x})$.*

*Proof.* Just let $\Phi(e, \overrightarrow{m}, \overrightarrow{x}) = U(\text{least } q)\ T_1(e, \overrightarrow{m}, \overrightarrow{x}\lceil q, q))$, where $T_1$ and $U$ are given by Theorem 1.6.2. □

The finite approximation $\Phi_{e,s}$ at *stage* $s$ of a partial computable function $\Phi_e$ is defined as follows.

**Definition 1.6.4.** *(i)* $\Phi_{e,s}(\overrightarrow{m}, \overrightarrow{x}) = p$ *if and only if*

$$(\exists q < s)[T_1(e, \overrightarrow{m}, \overrightarrow{x}\lceil q, q)\ \&\ U(q) = p].$$

*(ii)* $\Phi_{e,s}(\overrightarrow{m}, \overrightarrow{x})$ converges *(written $\Phi_{e,s}(\overrightarrow{m}, \overrightarrow{x})\downarrow$) if $\Phi_{e,s}(\overrightarrow{m}, \overrightarrow{x}) = p$ for some $p$ and otherwise* $\Phi_{e,s}(\overrightarrow{m}, \overrightarrow{x})$ diverges *( $\Phi_{e,s}(\overrightarrow{m}, \overrightarrow{x})\uparrow$). Similar definitions apply for $\Phi_{e,s}(\overrightarrow{m}, \overrightarrow{\sigma})$.*

*(iii)* $\Phi_e(\overrightarrow{m}, \overrightarrow{\sigma}) = \Phi_{e,s}(\overrightarrow{m}, \overrightarrow{\sigma})$*, where* $s = |\overrightarrow{\sigma}|$.

The following results are immediate from the definitions and the Normal Form Theorem above. For simplicity of expression, the results are written only for a function of one real variable but applies to functions of several variables as well.

**Theorem 1.6.5** (Master Enumeration Theorem)**.** *$\{\langle e, \overrightarrow{m}, \sigma, s\rangle : \Phi_{e,s}(\overrightarrow{m}, \sigma)\downarrow\}$ and $\{\langle e, \overrightarrow{m}, \sigma, p, s\rangle : \Phi_{e,s}(\overrightarrow{m}, \sigma) = p\}$ are both primitive recursive sets.*

**Theorem 1.6.6.** *(Use Principle)*

*(a)* $\Phi_e(\overrightarrow{m}, x) = n \Longrightarrow (\exists s)(\exists \sigma \subset x)\Phi_{e,s}(\overrightarrow{m}, \sigma) = n.$

*(b)* $\Phi_{e,s}(\overrightarrow{m}, \sigma) = n \Longrightarrow (\forall t \geq s)(\forall \tau \supset \sigma)\Phi_{e,t}(\overrightarrow{m}, \tau) = n.$

*(c)* $\Phi_{e,s}(\overrightarrow{m}, \sigma) = n \to (\forall x \supset \sigma)\Phi_e(\overrightarrow{m}, x) = n.$

**Theorem 1.6.7** (s-m-n Theorem)**.** *For every $m, n \geq 1$, there exists a one-to-one primitive recursive function $S^m_n$ such that, for all $e, i_1, \ldots, i_m, j_1, \ldots, j_n$,*

$$\Phi_{S^m_n(e,i_1,\ldots,i_m)}(j_1, \ldots, j_n, \overrightarrow{x}) = \Phi_e(i_1, \ldots, i_m, j_1, \ldots, j_n, \overrightarrow{x})$$

*Proof.* For $m = 1$, we want $S^1_1(e, i)$ to be the index for the function $\phi$ such that $\phi(j_1, \ldots, j_n, \overrightarrow{x}) = \phi_e(i, j_1, \ldots, j_n, \overrightarrow{x})$. Let $u$ be given by the Enumeration Theorem so that $\phi_u(e, i, j_1, \ldots, j_n, \overrightarrow{x}) = \phi_e(i, j_1, \ldots, j_n, \overrightarrow{x})$. Let $C_k$ denote the constant function $C_k(\overrightarrow{m}, \overrightarrow{x}) = k$ and let $P_i$ denote the projection function $P_i(\overrightarrow{m}, \overrightarrow{x}) = m_i$, both with $n$ number and $\ell$ real variables. Then

$$\begin{aligned}\phi(j_1, \ldots, j_n, \overrightarrow{x}) &= \phi_u(e, i, j_1, \ldots, j_n, \overrightarrow{x}) \\ &= \phi_u(C_e(\overrightarrow{j}, \overrightarrow{x}), C_i(\overrightarrow{j}, \overrightarrow{x}), P_0(\overrightarrow{j}, \overrightarrow{x}), \ldots, P_{n-1}(\overrightarrow{j}, \overrightarrow{x})),\end{aligned}$$

so that

$$S^1_1(e, i) = \langle 4, n+1, \ell, u, \langle 0, n, \ell, e\rangle, \langle 0, n, \ell, i\rangle, \langle 1, n, \ell, 0\rangle, \ldots, \langle 1, n, \ell, n-1\rangle\rangle.$$

Then $S^{m+1}_n$ may be defined recursively by

$$S^{m+1}_n(e, i_0, \ldots, i_m) = S^m_n(S^1_{m+n}(e, i_0), i_1, \ldots, i_m).$$

□

This result is very useful. Here is an example.

**Proposition 1.6.8.** *There is a primitive recursive function $g$ such that, for all $a$ and $b$, $W_{g(a,b)} = W_a \cup W_b$.*

*Proof.* Let $\phi(a, b, m) = U((\text{least } q)[T_1(a, m, q) \vee T_1(b, m, q)])$ and let $\phi$ have index $e$. Then let $g(a, b) = S^2_1(e, a, b)$. □

More importantly, we will need the following.

**Theorem 1.6.9** (Substitution Theorem)**.** *There is a primitive recursive function $f$ such that, for all $e, m, A$ such that $\Phi^A_b$ is total, $\Phi_e(m, \Phi^A_b) = \Phi_{f(b,e)}(m, A)$.*

*Proof.* Let $R(e, b, m, \sigma)$ if $\Phi_e(m, \sigma)\downarrow$; $R$ is primitive recursive by the Master Enumeration Theorem. Now let $g(e, b, m, A) = (\text{least } s)R(e, b, m, A\lceil s)$ and

$$\Phi_c(e, b, m, A) = A\lceil g(e, b, m, A),$$

where we identify a finite sequence with its code. Then

$$\Phi_e(m, \Phi_b^A) = \Phi_e(m, \Phi_c(e, b, m, A)) = \Phi_d(e, b, m, A),$$

where

$$d = \langle 4, 3, 1, \langle 1, 3, 1, 2\rangle, c\rangle.$$

Now apply the s-m-n Theorem to get $f(b, e) = S_1^2(d, e, b)$. □

**Theorem 1.6.10** (Recursion Theorem)**.** *For any partial computable function $\Phi$, there exists an index $e$ such that, for all $\overrightarrow{m}$, $\Phi_e(\overrightarrow{m}, \overrightarrow{x}) = \Phi(e, \overrightarrow{m}, \overrightarrow{x})$. Furthermore, there is a primitive recursive function $g$ such that if $\Phi = \Phi_i$, then $e = g(i)$.*

*Proof.* Given $\Phi$, let $\Phi_b(a, \overrightarrow{m}, \overrightarrow{x}) = \Phi(S_1^{k+1}(a, a), \overrightarrow{m}, \overrightarrow{x})$ and let $e = S_1^{k+1}(b, b)$. Then

$$\Phi_e(\overrightarrow{m}, \overrightarrow{x}) = \Phi_b(b, \overrightarrow{m}, \overrightarrow{x}) = \Phi(S^{k+1}(b, b), \overrightarrow{m}, \overrightarrow{x}) = \Phi(e, \overrightarrow{m}, \overrightarrow{x}).$$

□

This leads to the following.

**Theorem 1.6.11** (Fixed Point Theorem)**.** *For any computable function $f$, there exists an index $e$ such that $\Phi_e = \Phi_{f(e)}$. Furthermore, there is a primitive recursive function $h$ such that if $f = \Phi_i$, then $e = h(i)$.*

*Proof.* Let $\Phi(a, \overrightarrow{m}, \overrightarrow{x}) = \Phi_{f(a)}(\overrightarrow{m}, \overrightarrow{x})$ and let $e$ be given by the Recursion Theorem such that $\phi_e(\overrightarrow{m}, \overrightarrow{x}) = \phi(e, \overrightarrow{m}, \overrightarrow{x})$. □

**Corollary 1.6.12.** *For any computable function $f$, there exists an index $e$ such that $W_e = W_{f(e)}$. Furthermore, there is a primitive recursive function $h$ such that if $f = \phi_i$, then $e = h(i)$.*

**Definition 1.6.13.** *A function $F : (\mathbb{N}^{\mathbb{N}})^{\ell} \to \mathbb{N}^{\mathbb{N}}$ is (partial) computable (or* computably continuous*) if there is a (partial) computable functional $\Phi$ such that, for all $\overrightarrow{x}$ and $n$, $\Phi(n, \overrightarrow{x}) = F(\overrightarrow{x})(n)$.*

**Theorem 1.6.14.** *Let $F : (\mathbb{N}^{\mathbb{N}})^{\ell} \to \mathbb{N}^{\mathbb{N}}$ be total. Then $F$ is continuous if and only if $F$ is computable in some oracle $A \subseteq \mathbb{N}$.*

*Proof.* ($\longleftarrow$). We give the proof for $\ell = 1$. Let $A$ be the given oracle and suppose that $F(x)(m) = \Phi(m, x, A)$. It suffices to show that, for any $m$ and $n$, $\{x \in \mathbb{N}^{\mathbb{N}} : F(x)(m) = n\}$ is an open set. Suppose that $F(x)(m) = n$. By the Use Principle (Theorem 1.6.6) there is some $s$ and some finite $\sigma \subset x$ and finite $\tau \subset \chi_A$ such that $\Phi_{e,s}(m,\sigma,\tau) = n$. It follows that for all $x \in I(\sigma)$, $\Phi_{e,s}(m,x,\tau) = n$ and hence $I(\sigma) \subseteq F^{-1}(\{y : y(m) = n\})$.

($\longrightarrow$): Let $F : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ be continuous. Then for each $m$ and $n$, $U_{m,n} = \{x : F(x)(m) = n\}$ is open and thus for each $x \in U_{m,n}$, there exists a finite $\sigma \prec x$ such that $I(\sigma) \subseteq U_{m,n}$. Let

$$A = \{\langle m, n, \sigma\rangle : I(\sigma) \subseteq U_{m,n}\}.$$

To compute $F(x)(m)$ from $x$ simply fix $m$ and search for the least $\langle m, n, \sigma\rangle \in A$ such that $\sigma \prec x$; then $F(x)(m) = n$. □

**Exercises**

1.6.1. Use induction to prove Theorem 1.6.6 (a).

1.6.2. Use the Recursion Theorem to show that the Fibonacci sequence $1, 1, 2, 3, 5, 8, \ldots$ is computable.

## 1.7 Computably enumerable sets

**Definition 1.7.1.** *(i) A subset $A$ of $\mathbb{N}$ is* computably enumerable *(*c. e.*) if $A$ is the domain of some partial computable function.*

*(ii) The c. e. sets can be enumerated in the form*

$$W_e = \{m : \phi_e(m)\downarrow\} = \{m : (\exists q)T(e, m, q)\}.$$

*(iii)* $W_{e,s} = \{m : \phi_{e,s}(m)\downarrow\}$.

The following lemma is immediate from Theorem 1.6.5.

**Lemma 1.7.2.** $\{\langle e, m, s\rangle : m \in W_{e,s}\}$ *is primitive recursive.* □

There are several equivalent definitions.

**Definition 1.7.3.** *A set $A \subseteq \mathbb{N}^k$ is $\Sigma^0_1$ (resp. $\Sigma^B_1$) if there is a computable relation $R$ (resp. computable in $B$) such that, for all $\overrightarrow{m}$, $\overrightarrow{m} \in A \iff (\exists p)R(p, \overrightarrow{m})$.*

**Theorem 1.7.4** (Normal Form Theorem for c. e. sets)**.** *A set $A$ is c. e. if and only if it is $\Sigma^0_1$.* □

The proof is left as an exercise. Observe that any computable set is trivially $\Sigma^0_1$ and hence also is computably enumerable.

**Theorem 1.7.5** (Quantifier Contraction Theorem)**.** *If $W$ is a c. e. set, then $\{m : (\exists p)\langle p, m\rangle \in W\}$ is a c. e. set.*

*Proof.* Let $V = \{m : (\exists p)\langle m, p\rangle \in W\}$. Then $m \in V \iff (\exists q)\langle m, (q)_0\rangle \in W_{e,(q)_1}$. Thus $V$ is c. e. by the Normal Form Theorem for c. e. sets. □

The intended meaning of the term "computably enumerable set" is that there is an effective listing $a_0, a_1, \ldots$ of the set.

**Theorem 1.7.6** (Listing Theorem)**.** *A set $A$ is c. e. if and only if either $A = \emptyset$ or $A$ is the range of a total computable function.*

*Proof.* ($\Longleftarrow$): If $A = \emptyset$, then $A$ is c. e. If $A = \{\phi_e(m) : m \in \mathbb{N}\}$ where $\phi$ is a total computable function, then

$$n \in A \iff (\exists n)(\exists s)\phi_{e,s}(p) = n.$$

Thus $A$ is c. e. by Theorem 1.6.5 and the Quantifier Contraction Theorem.

($\Longrightarrow$): Let $A = W_e \neq \emptyset$ and choose $a \in A$. Then $A$ is the range of the following computable function.

$$f(\langle m, s\rangle) = \begin{cases} m, & \text{if } \ m \in W_{e,s+1} \setminus W_{e,s}; \\ a, & \text{otherwise.} \end{cases}$$

□

**Theorem 1.7.7** (Complementation Theorem)**.** *A set $A$ is computable if and only if both $A$ and $\mathbb{N} \setminus A$ are c. e.*

*Proof.* ($\Longleftarrow$): If $A$ is computable, then $\mathbb{N} \setminus A$ is also computable and hence both sets are c. e..

($\Longrightarrow$): Suppose that $A = W_a$ and $\mathbb{N} \setminus A = W_b$ and let $\phi(m) = (\text{least } s)[m \in W_{a,s} \ \vee \ m \in W_{b,s}]$. Then $\phi$ is a total computable function and $m \in A \iff m \in W_{a,\phi(m)}$, so that $A$ is computable. □

There are natural noncomputable c. e. sets.

**Definition 1.7.8.** *(a) $K = \{e : e \in W_e\}$;*

*(b) $K_0 = \{\langle m, e\rangle : m \in W_e\}$.*

**Proposition 1.7.9.** *$K$ and $K_0$ are noncomputable c. e. sets.*

*Proof.* It follows from Lemma 1.7.2 that $K_0$ and $K$ are c. e. sets. Suppose now that $K$ were computable, so that $\mathbb{N} \setminus K$ is c. e., by the Complementation Theorem, and choose $a$ such that $\mathbb{N} \setminus K = W_a$. Then, for any $m$,

$$m \in W_m \iff m \in K \iff m \notin W_a$$

and when $m = a$ we obtain the contradiction

$$a \in W_a \iff a \in K \iff a \notin W_a.$$

Now $a \in K \iff \langle a, a\rangle \in K_0$, so that $K$ would be computable if $K_0$ were computable. □

**Exercises**

1.7.1. Prove lemma 1.7.2 and the Normal Form Theorem for c.e. sets. Hint: Use the corresponding results for partial computable functions.

1.7.2. Show that a partial function is partial computable function if and only if the graph is $\Sigma^0_1$.

1.7.3. Use the s-m-n Theorem to obtain a primitive recursive function such that for any $e$, $\{m : (\exists p)\langle m, p\rangle \in W_e\} = W_{f(e)}$.

1.7.4. Show that if $A$ is a $\Sigma^0_1$ relation and $B = \{\langle m, p\rangle : (\forall n < p)\langle m, n\rangle \in A\}$, then $B$ is also $\Sigma^0_1$.

## 1.8 Computability of real numbers

Any set $A$ of natural numbers represents a real number $r_A \in [0, 1]$ where $r_A = \sum_{n\in A} 2^{-n-1}$. For every real $r$ in $[0, 1]$, there exists $A \subseteq \mathbb{N}$ such that $r = r_A$ and $r$ has a unique representation except for dyadic rationals $r$, which have exactly two such representations. The real $r_A$ is said to be computable if $A$ is a computable set. For an arbitrary real $x$, we have $x = i + r$, where $i$ is an integer and $r \in [0, 1]$, so we will say that $x$ is computable if and only if $r$ is computable.

The unit interval $[0, 1] \subset \mathbb{R}$ has a natural linear ordering and this corresponds to the lexicographic ordering on $\{0, 1\}^{\mathbb{N}}$.

**Definition 1.8.1.** *For $x, y \in \mathbb{N}^{\mathbb{N}}$, $x <_{lex} y$ if $x(n) < y(n)$ where $n$ is the least such that $x(n) \neq y(n)$.*

It is easy to see that $<_{lex}$ is a linear ordering on $\mathbb{N}^{\mathbb{N}}$. We sometimes say that "$x$ is left of $y$" if $x <_{lex} y$, since this fits the picture of the tree $\mathbb{N}^*$. For $x, y \in \{0, 1\}^{\mathbb{N}}$, if $r_x \neq r_y$, then $r_x < r_y \iff x <_{lex} y$. If $r$ is a dyadic rational, then there are two representations $x \neq y$ such that $r_x = r_y = r$ and these are successors under $<_{lex}$.

Another useful way of determining the complexity of a real number is by means of Dedekind cuts of rationals. Rational numbers may be represented as quotients of integers and thereby as finite sequences of natural numbers. Thus we may view the set $\mathbb{Q}$ of rational numbers as a computable structure equipped with a computable ordering and computable operations of addition, subtraction, multiplication and division. The Dedekind cut $L(r)$ of a real number is defined by

$$L(r) = \{q \in \mathbb{Q} : q \leq r\}.$$

It turns out that the complexity of the Dedekind cut is quite useful in computable analysis.

**Proposition 1.8.2.** *For any real $r$, $r$ is computable if and only if $L(r)$ is computable.*

*Proof.* It suffices to consider $r \in [0,1]$, so let $r = r_x$ for some $x \in \{0,1\}^{\mathbb{N}}$. If $r$ is rational, then both $r$ and $L(r)$ are computable. So let $r$ be irrational and suppose first that $r$ is computable. Then, for any rational $q$,

$$q < r \iff (\exists n) q < \sum_{i=0}^{n} x(i) 2^{-i-1}$$

and

$$q > r \iff (\exists n) q > 2^{-n-1} + r \iff (\exists n) q > 2^{-n-1} + \sum_{i=0}^{n-1} x(i) 2^{-i-1}.$$

Next suppose that $L(r)$ is computable. Then we can recursively define $x$ so that $r = r_x$ as follows. Let $x(0) = 0$, if $r < \frac{1}{2}$ and $x(0) = 1$ otherwise. Given $x(n)$, let

$$x(n+1) = \begin{cases} 0, & \text{if } r < 2^{-n-2} + \sum_{i=0^n} x(i) 2^{-i-1}, \\ 1, \text{otherwise.} \end{cases}$$

$\square$

There is a nice characterization for $\Sigma_1^0$ and $\Pi_1^0$ Dedekind cuts.

**Proposition 1.8.3.** *(a) $L(r)$ is $\Sigma_1^0$ if and only if $r = \lim_n q_n$, where $\{q_n\}_{n\in\omega}$ is a computable, increasing sequence of rationals.*

*(b) $L(r)$ is $\Pi_1^0$ if and only if $r = \lim_n q_n$, where $\{q_n\}_{n\in\omega}$ is a computable, decreasing sequence of rationals.*

*Proof.* (a) Suppose first that $r = lim_n q_n$ where $\{q_n\}_{n\in\omega}$ is a computable, increasing sequence of rationals. Then, for any rational $q$, $q < r \iff (\exists n) q < q_n$.

Suppose now that $L(r)$ is $\Sigma_1^0$ and let $L(r)$ have a computable enumeration as $p_0, p_1, \ldots$. Then we can define a computable nondecreasing sequence $q_n$ of rationals with limit $r$ by

$$q_n = max\{p_0, p_1, \ldots, p_n\}.$$

It is routine to convert this into an increasing sequence.

(b) If $L(r)$ is $\Pi_1^0$, then $L(1-r)$ is $\Sigma_1^0$, since $q < 1 - r \iff r < 1 - q$. Thus $1-r = lim_n p_n$ where $\{p_n\}_{n\in\omega}$ is a computable, increasing sequence of rationals. It follows that $r = lim_n(1-p_n)$ is the limit of a computable, decreasing sequence. Conversely, if $r = lim_n q_n$ where $\{q_n\}_{n\in\omega}$ is a computable, increasing sequence of rationals, then $1 - r$ is the limit of a decreasing sequence so that $L(1-r)$ is $\Sigma_1^0$ and hence $L(r)$ is $\Pi_1^0$. $\square$

The following notions are important in computable analysis.

**Definition 1.8.4.** *Let $r$ be a real number. Then*

*(a) $r$ is* lower semicomputable *if it is the limit of an increasing computable sequence of rationals;*

(b) $r$ *is* upper semicomputable *if it is the limit of a decreasing computable sequence of rationals;*

(c) $r$ *is* weakly computable *if it is either lower semicomputable or upper semicomputable.*

It would be natural to say that $r_x$ is $\Sigma^0_1$ if the set with characteristic function $x$ is $\Sigma^0_1$, but there is no corresponding equivalence as in Proposition 1.8.2. One direction only holds. In example 1.9.3 below we will construct a lower semicomputable real $r$ which is not the characteristic function of a c. e. set.

**Proposition 1.8.5.** *(a) If $A$ is $\Sigma^0_1$, then $L(r_A)$ is $\Sigma^0_1$;*

*(b) If $A$ is $\Pi^0_1$, then $L(r_A)$ is $\Pi^0_1$.*

*Proof.* (a) If $A$ is finite, then of course $r_A$ is rational and therefore $L(r_A)$ is computable. Suppose therefore that $A$ is $\Sigma^0_1$ and infinite and let $A$ have computable enumeration $a_0, a_1, \dots$ without repetition. Then for any rational $q$,

$$q < r_A \iff (\exists n) q < \sum_{i=0}^{n} 2^{-a_i - 1}.$$

(b) If $A$ is $\Pi^0_1$, then $\mathbb{N} \setminus A$ is $\Sigma^0_1$ and $r_{\mathbb{N}\setminus A} = 1 - r_A$, so that $L(1 - r_A)$ is $\Sigma^0_1$ and therefore $L(r_A)$ is $\Pi^0_1$. □

**Exercises**

1.8.1. Show that a real number $r$ is computable if and only if there is a computable sequence $q_n$ of rationals such that $|q_n - r| < 2^{-n}$ for all $n$.

## 1.9 Turing, many-one, and truth-table reducibility

**Definition 1.9.1.** *(i) $A$ is* many-one reducible *(*$m$-reducible*) to $B$ ($A \leq_m B$ if there is a computable function $f$ such that $a \in A \iff f(a) \in B$*

*(ii) $A$ is* one-one reducible *to $B$ ($A \leq_1 B$ if there is a one-to-one computable function $f$ such that $a \in A \iff f(a) \in B$*

*(iii) $C$ is* $m$-complete *(or* $\Sigma^0_1$ complete*) if $A \leq_m C$ for all c. e. sets $A$.*

For example, any c. e. set is $m$-reducible to $K_0$, since $m \in W_e \iff \langle m, e\rangle \in K_0$; here the function $f$ is given by $f(m) = \langle m, e\rangle$. Thus $K_0$ is $m$-complete. The following useful lemma is left as an exercise.

**Lemma 1.9.2.** *If every $\Sigma^0_1$ (respectively $\Pi^0_1$) set is $m$-reducible to $A$, then $A$ is not $\Pi^0_1$ (resp. $\Sigma^0_1$).* □

**Example 1.9.3.** *Let $K$ be a noncomputable c. e. set and let $A = \{2n : n \in K\} \cup \{2n+1 : n \notin K\}$. Then $A$ is a difference of c. e. sets and is $m$-complete for both $\Pi^0_1$ and $\Sigma^0_1$ sets. It follows from Lemma 1.9.2 that $A$ is not $\Sigma^0_1$, but $r_A$ is the limit of the nondecreasing computable sequence $\{q_s\}_{s\in\omega}$ defined as follows. Let $K = \cup_s K_s$ where $K_s$ is a uniformly computable finite subset of $\{0, 1, \ldots, s-1\}$ and let*

$$q_s = \sum\{2^{-2n-1} : n \in K_s\} + \sum\{2^{-2n-2} : n < s \;\&\; n \notin K_s\}.$$

*Observe that if $n \in K_s \setminus K_{s-1}$, then $2^{-2n-1}$ is added to the first part of $q_s$ and $2^{-2n-2}$ is subtracted from the second part, so that $q_{s-1} < q_s$.*

**Definition 1.9.4.** *(i) $A \equiv_m B$ if $A \leq_m B$ and $B \leq_m A$.*

*(ii) $A \equiv_1 B$ if $A \leq_1 B$ and $B \leq_1 A$.*

**Proposition 1.9.5.** *Suppose that $A \leq_m B$. If $B$ is c. e., then $A$ is c. e. and if $B$ is computable, then $A$ is computable.*

The proof is left as an exercise.

**Definition 1.9.6.** *$A$ is* computably isomorphic *to $B$ (written $A \equiv B$) if there is a computable permutation $\pi$ of $\mathbb{N}$ such that $\pi[A] = B$.*

The following is an effective version of the classic Cantor-Schröder-Bernstein Theorem.

**Theorem 1.9.7** (Cantor-Schröder-Bernstein Theorem)**.** *Let $A$ and $B$ be sets and let $f$ and $g$ be injections, $f : A \to B$ and $g : A \to B$; then there exists an isomorphism $h : A \to B$.* □

Banach's version of the Cantor-Schröder-Bernstein Theorem adds the requirement that, for all $a \in A$, either $h(a) = f(a)$ or $h(a) = g^{-1}(a)$.

**Theorem 1.9.8** (Myhill Isomorphism Theorem)**.** *$A \equiv B \iff A \equiv_1 B$.*

*Proof.* The direction ($\Longrightarrow$) is trivial. Suppose therefore that $A \leq_1 B$ via $f$ and $B \leq_1 A$ via $g$. We will define $\pi$ in stages $\pi_s = \{\langle m_0, n_0\rangle, \ldots, \langle m_{2s-1}, n_{2s-1}\rangle\}$ so that for all $m < s$, $m \in Dom(\pi_s)$ and $m \in Ran(\pi_s)$ and such that

$$m_i \in A \iff n_i \in B.$$

We begin with $\pi_0 = \emptyset$.

**Stage** $s+1$: Let $\pi_s$ be given as above and let $m = m_{2s}$ be the least $m \notin Dom(\pi_s)$. $\pi(m)$ is computed as follows. First compute $b_0 = f(m)$ and check if $b_0 \in Ran(\pi_s)$. If not, then $b_0 = \pi_{s+1}(m)$. If so, then compute $b_1 = f(\pi_s^{-1}(b_0))$ and again check whether $b_1 \in Ran(\pi_s)$ and let $b_1 = \pi_{s+1}(m)$ if not and $b_2 = f(\pi_s^{-1}(b_0))$ if so. Observe that after we reach $b_{2s-1}$, $Ran(\pi_s) = \{b_0, b_1, \ldots, b_{2s-1}\}$ is exhausted, so that $b_{2s} = \pi_{s+1}(m)$.

Next let $n = n_{2s+1}$ be the least not in $Ran(\pi_s) \cup \{\pi_{s+1}(m)\}$ and similarly define $a_0 = g(n)$, $a_1 = g(\pi_s(a_0))$, and so on to obtain $\pi^{-1}_{s+1}(n)$. □

We now show that, in the setting of $m$-reduciblity, Banach's version of the Cantor-Schröder-Bernstein Theorem is not effective,

**Theorem 1.9.9.** *There exist computable injections $f : \mathbb{N} \to \mathbb{N}$ and $g : \mathbb{N} \to \mathbb{N}$ such that, for any computable permutation $h$ of $\mathbb{N}$, there is some $i$ such that $h(i) \neq f(i)$ and $h(i) \neq g^{-1}(i)$.*

*Proof.* Let $K$ be a noncomputable c. e. set and let $\psi$ be a one-one computable function with range $K$. We define injections $f$ and $g$ as follows.

$$\textbf{(1)}\quad f(2^m(2n+1)) = \begin{cases} 2n+1, & \text{if } \psi(m-2) = n, \\ 2^m(2n+1), & \text{if } (\exists j < m-2)\psi(j) = n, \\ 2^{m+1}(2n+1), & \text{otherwise.} \end{cases}$$

$$\textbf{(2)}\quad g(2^m(2n+1)) = \begin{cases} 2^{m+1}(2n+1), & \text{if } (\exists j \leq m-2)\psi(j) = n, \\ 2^m(2n+1) & \text{otherwise.} \end{cases}$$

Now by way of contradiction, let $h$ be a computable permutation such that, for all $i$, either $h(i) = f(i)$ or $h(i) = g^{-1}(i)$. We claim that

$$n \notin K \iff h(2n+1) = 2n+1.$$

This would contradict the assumption that $K$ is not computable. It remains to verify the claim. Suppose first that $n \notin K$. Then $g(2n+1) = 2n+1$ and $2n+1$ is not in the range of $f$, so that $h(2n+1) = g^{-1}(2n+1) = 2n+1$. Next suppose that $n \in K$ and let $n = \psi(m)$. Then $2^{m+2}(2n+1)$ is not in the range of $g$, so $h(2^{m+2}(2n+1)) = f(2^{m+2}(2n+1) = 2n+1$, so that $h(2n+1) \neq 2n+1$.

□

Let *Sent* be the set of propositional sentences on variables $a_0, a_1, \ldots$. There is a an effective enumeration $\psi_0, \psi_1, \ldots$ of these sentences, so that we may identify the sentence $\psi_n$ with $n$ in context. For any set $B \subseteq \mathbb{N}$ and any sentence $\psi \in \mathit{Sent}$, we say that $B \models \psi$ if $\psi$ is true under the truth assignment which makes $a_i$ true if and only if $i \in B$. Let $B^{tt}$ denote the set of $\psi \in \mathit{Sent}$ such that $B \models \psi$. Then we say that $A$ is truth-table reducible to $B$ ($A \leq_{tt} B$) if $A \leq_m B^{tt}$. Equivalently, $A \leq_{tt} B$ if and only if there exists a computable relation $R$ and a computable function $f$ such that for any $n$, $n \in A \iff R(\langle B \upharpoonright f(n) \rangle)$. It is immediate that many-one reducibility implies truth-table reducibility.

**Theorem 1.9.10.** *(Trakhtenbrot-Nerode [194, 143]) $A \leq_{tt} B$ if and only if there is a total, computable function $\Phi : \{0,1\}^{\mathbb{N}} \to \{0,1\}^{\mathbb{N}}$ such that $\Phi(B) = A$.*

*Proof.* Suppose first that there is a total computable function $\Phi$ such that $\Phi(B) = A$. We will define a function $f : \mathbb{N} \to \mathit{Sent}$ such that $a \in A \iff B \models f(a)$. Given $a$, use the Master Enumeration Theorem 1.6.5 to compute the least $n$ such that $\Phi(a, \sigma)\downarrow$ for all $\sigma \in \{0,1\}^n$. For each $\sigma$ such that $\Phi(a, \sigma) = 1$, let $\varphi_\sigma$ be the conjunction $\bigwedge_{i=0}^{n-1} b_i$, where $b_i = a_i$ if $\sigma(i) = 1$ and $b_i = \neg a_i$ if

$\sigma(i) = 0$. Thus $B \models \varphi_\sigma$ if and only if, for all $i < n$, $\sigma(i) = 1 \iff i \in B$. Finally, let $\varphi$ be the disjunction of $\{\varphi_\sigma : \Phi(a, \sigma) = 1\}$. Then

$$a \in A \iff \Phi(a, B) = 1 \iff B \models \varphi.$$

Next suppose that $A \leq_{tt} B$ and let $f$ be given so that $a \in A \iff B \models f(a)$. Then in general, define $\Phi$ so that

$$\Phi(c, X) = 1 \iff X \models f(c)$$

Here we can compute for each propositional variable $a_i$ occuring in $f(c)$, whether $X \models a_i$ immediately from $X$ and then use truth tables to check whether $X \models f(c)$. $\square$

This leads naturally to *Turing reducibility*, where we allow *partial* computable functions.

**Definition 1.9.11.** *(i) A is* Turing reducible *to $B$ (written $A \leq_T B$) if there is a functional $\Phi$ such that, for all $m$, $A(m) = \Phi(m, B)$.*

*(ii) $A$ is* Turing equivalent *to $B$ if both $A \leq_T B$ and $B \leq_T A$.*

*(iii) The Turing degree* $\mathbf{a}$ *of $A$ is the equivalence class of $A$ under Turing equivalence.*

Informally, this means that $A \leq_T B$ if $A$ can be computed using $B$ as an oracle. It follows from Theorem 1.9.10 that truth-table reducibity implies Turing reducibility. It is easy to see that $\equiv_T$ is an equivalence relation. The Turing degrees are partially ordered by $\leq_T$ with least element $\mathbf{0}$, which is the Turing degree of a recursive set.

It is possible that $A \leq_T B$ but $A$ is not truth-table reducible to $B$. (See the exercises below.) However, there is a family of sets $B$ for which the two reducibilities are equivalent.

**Definition 1.9.12.** *A function $g \in \mathbb{N}^{\mathbb{N}}$ is* almost computable *(or* hyperimmune-free*) if for all $f \leq_T g$, there exists a computable function $h$ such that $f(n) \leq h(n)$ for all $n$.*

**Theorem 1.9.13.** *Suppose $g$ is almost computable. Then for all $f$, if $f \leq_T g$, then $f \leq_{tt} g$.*

*Proof.* Suppose that $y$ is almost computable and that $f(n) = \Phi(n, g)$ where $\Phi$ is computable. Define $u(n)$ to be the least $k$ such that $\Phi(n, g\lceil k) \downarrow$. Then $u$ is computable from $g$ and hence there is a computable function $h$ such that $g(n) \leq h(n)$ for all $n$. It follows that $f(n) = \Phi(n, g\lceil h(n))$ for all $n$, so that $\Phi$ can be extended to a total function $\Psi$ by letting $\Psi(n, x) = 0$ whenever $\Phi(n, x\lceil h(n)) \uparrow$. Hence $f$ is truth-table reducible to $g$, as desired. $\square$

**Exercises**

1.9.1. Prove Lemma 1.9.2 and show that $A$ is $\Sigma^0_1$ complete if and only if $K \leq_m A$.

1.9.2. Prove Proposition 1.9.5.

1.9.3. Show that $\leq_m$ and $\leq_T$ is transitive.

1.9.4. Show that the two definitions of truth-table reducibility are equivalent.

1.9.5. Give an example to show that $A \leq_T B$ but not $A \leq_{tt} B$. (Hint: let $A$ be the set of $e$ such that $\Phi_e$ defines a total functional $F_e$, where $y = F_e(x)$ means that $y(m) = \Phi_e(m, x)$. Then let $e \in B \iff (e \in A \ \& \ \Phi_e(e, A) = 0)$.

## 1.10 The jump and the arithmetical hierarchy

**Definition 1.10.1.** *1.* $W_e^A = \{m : \Phi_e(m, A)\downarrow\}$.

*2. $B$ is c. e. in $A$ if $B = W_e^A$ for some $e$.*

*3. The* jump *of $A$ is $K_0^A = \{\langle e, m\rangle : m \in W_e^A\}$ and is denoted by $A'$.*

*4. $A^{(n)}$ is the $n$th jump of $A$, that is, $A^{(0)} = A$ and $A^{(n+1)} = (A^{(n)})'$.*

The following two theorems generalize from results of Section 1.7 and 1.9.

**Theorem 1.10.2.** *The following are equivalent:*

*(a) $B$ is c. e. in $A$;*

*(b) $B = \emptyset$ or $B = Ran(\phi_e^A)$ for some $e$;*

*(c) $B$ is $\Sigma_1^A$.* □

**Theorem 1.10.3.** *$B \leq_T A$ if and only if $B$ and $\mathbb{N} \setminus B$ are both c. e. in $A$.* □

**Theorem 1.10.4** (Jump Theorem)**.** *(a) $A' \not\leq_T A$.*

*(b) $B$ is c. e. in $A$ if and only if $B \leq_1 A'$.*

*(c) $B \leq_T A$ if and only if $B' \leq_1 A'$.*

*Proof.* Parts (a) and (b) relativize from results of sections 1.7 and 1.9. For part (c), suppose first that $B \leq_T A$ and let $B(n) = \Phi_b(n, A)$. Then for any $e$, $\Phi_e(m, B) = \Phi_e(m, \Phi_b^A)$ and by Theorem 1.6.9, there is a primitive recursive function such that $\Phi_e(m, B) = \Phi_{f(e)}(m, A)$. Thus $\langle e, m\rangle \in B' \iff \langle f(e), m\rangle \in A'$. For the other direction, suppose that $B' \leq_1 A'$. Then $B$ and $\mathbb{N} \setminus B$ are both c. e. in $A$ and therefore $B \leq_T A$ by Theorem 1.10.3. □

In particular, there is an infinite hierarchy of degrees $\mathbf{0^{(n)}} = deg(\emptyset^{(n)})$.

The arithmetical hierarchy of sets of natural numbers may be defined as follows.

**Definition 1.10.5.** *Let $R \subseteq \mathbb{N}^k \times (\mathbb{N}^{\mathbb{N}})^\ell$ and let $n > 0$ be a natural number.*

1. *$R$ is $\Sigma^0_0$ if it is computable.*
2. *$R$ is $\Pi^0_n$ if $\mathbb{N}^k \times (\mathbb{N}^{\mathbb{N}})^\ell \setminus R$ is $\Sigma^0_n$.*
3. *$R$ is $\Sigma^0_{n+1}$ if it is the projection of a $\Pi^0_n$ set, that is, if there exists a $\Pi^0_n$ relation $B$ such that, for all $\overrightarrow{m}$ and $\overrightarrow{x}$:*

$$R(\overrightarrow{m}, \overrightarrow{x}) \iff (\exists j)B(j, \overrightarrow{m}, \overrightarrow{x}).$$

4. *$R$ is $\Delta^0_n$ if it is both $\Sigma^0_n$ and $\Pi^0_n$.*

Note of course that the $\Sigma^0_1$ sets are just the computably enumerable sets. These definitions can be relativized to any oracle $C$ to define the $\Sigma^0_n[C]$, $\Pi^0_n[C]$ and $\Delta^0_n[C]$ sets and relations.

Here are some basic facts about the arithmetical hierarchy. Part (e) refers to *bounded quantification*. See [181] for proofs.

**Theorem 1.10.6.** *(a) $A \in \Sigma^0_n \cup \Pi^0_n$ and $m > n$ implies $A \in \Delta^0_m$;*

*(b) $A, B \in \Sigma^0_n(\Pi^0_n) \Longrightarrow A \cap B, A \cup B \in \Sigma^0_n(\Pi^0_n)$;*

*(c) If $R \in \Sigma^0_n$ for $n > 0$ and $A = \{m : (\exists p)R(m, p)\}$, then $R$ is $\Sigma^0_n$;*

*(d) If $B \leq_m A$ and $A \in \Sigma^0_n$, then $B \in \Sigma^0_n$;*

*(e) If $R \in \Sigma^0_n$ and $A = \{\langle m, p\rangle : (\forall i < p)R(i, m, p)\}$, then $A \in \Sigma^0_n$.* □

An important result is the following.

**Theorem 1.10.7** (Post's Theorem)**.** *For any subset $A$ of $\mathbb{N}$:*

*(a) $A$ is $\Sigma^0_{n+1}$ if and only if it is c. e. in $\emptyset^{(n)}$.*

*(b) $A$ is $\Delta^0_{n+1} \iff A \leq_T \emptyset^{(n)}$.*

*Proof.* The proofs are by induction on $n$. For $n = 0$, both parts are immediate. Now suppose by induction that (a) and (b) are true for $n$ and for all subsets of $\mathbb{N}$.

First we show that $\emptyset^{(n+1)}$ is $\Sigma^0_{n+1}$. That is, by induction assume that $\emptyset^{(n)}$ is $\Sigma^0_n$. Then

$$\langle e, m\rangle \in \emptyset^{(n+1)} \iff (\exists s)(\exists \sigma)[\sigma \subset \emptyset^{(n)} \ \& \ \Phi_e(m, \sigma)\downarrow].$$

But in general, $\sigma \subset C$ if and only if $(\forall i < |\sigma|)[\sigma(i) = 1 \iff i \in C]$, so that for $C = \emptyset^{(n)}$, this condition is a disjunction of $\Sigma^0_n$ and $\Pi^0_n$ clauses. It follows that the quantified condition is $\Delta^0_{n+1}$ and hence $\emptyset^{(n+1)}$ is $\Sigma^0_{n+1}$ by Theorem 1.10.6.

Now suppose that $A$ is c. e. in $\emptyset^{(n)}$. Then $A \leq_1 \emptyset^{(n+1)}$ and therefore $A$ is $\Sigma^0_{n+1}$.

Conversely, suppose that $A$ is $\Sigma^0_{n+1}$ and let $R$ be $\Pi^0_n$ such that $m \in A \iff (\exists p)R(m,p)$. Then $\mathbb{N} \setminus R$ is $\Sigma^0_n$ and hence c. e. in $\emptyset^{(n-1)}$ by induction. It follows from the Jump Theorem that $\mathbb{N} \setminus R \leq_1 \emptyset^{(n)}$ and therefore $R \leq_T \emptyset^{(n)}$. Since $A$ is c. e. in $R$, it follows that $A$ is also c. e. in $\emptyset^{(n)}$.

For part (b), $A$ is $\Delta^0_{n+1}$ if and only if both $A$ and $\mathbb{N} \setminus A$ are $\Sigma^0_{n+1}$, which is if and only if $A$ and $\mathbb{N} \setminus A$ are c. e. in $\emptyset^{(n)}$ (by (a)). But this is if and only if $A \leq_T \emptyset^{(n)}$ by the Jump Theorem. □

**Definition 1.10.8.** *$A$ is said to be $\Sigma^0_n$ ($\Pi^0_n$)* complete *if $A$ is $\Sigma^0_n$ ($\Pi^0_n$) and every $\Sigma^0_n$ ($\Pi^0_n$) set is m-reducible to $A$.*

It follows from Post's Theorem that $\emptyset^{(n)}$ is $\Sigma^0_n$ complete for all $n > 0$.

$\Delta^0_2$ sets and functions will be of particular interest.

**Definition 1.10.9.** *Let $\{f_s\}_{s\in\omega}$ be a sequence of total functions from $\mathbb{N}^{\mathbb{N}}$.*

*(i)* $\lim_s f_s = f$ *means that, for all $m$, there exists $s$ such that $f(m) = f_t(m)$ for all $t \geq s$;*

*(ii) $h$ is a* modulus of convergence *for $\{f_s\}_{s\in\omega}$, if, for all $m$ and all $s \geq h(m)$, $f_s(m) = f(m)$.*

Given a uniformly computable sequence $\{f_s\}_{s\in\omega}$ of functions with limit $f$ and modulus of convergence $h$, $f$ is always computable in $h$.

**Lemma 1.10.10** (Modulus Lemma)**.** *If $A$ is c. e. and $f \leq_T A$, then there is a uniformly computable sequence $\{f_s\}_{s\in\omega}$ such that* $\lim_s f_s = f$ *and a modulus of convergence $h \leq_T A$.*

*Proof.* Let $A = W_i$ be c. e., let $\sigma_s = W_{i,s}\lceil s$ and let $f = \Phi^A_e$. Now let

$$f_s(m) = \begin{cases} \Phi_{e,s}(m, \sigma_s), & \text{if convergent,} \\ 0, & \text{otherwise.} \end{cases}$$

and

$$h(m) = (\text{least } s)(\exists z \leq s)[\Phi_{e,s}(m, \sigma_s\lceil z)\downarrow \ \& \ \sigma_s\lceil z = A\lceil z].$$

Then $\{f_s\}_{s\in\omega}$ is a computable sequence with limit $f$ and $h$ is a modulus of convergence which is computable from $A$. □

In particular, this implies that any $\Delta^0_2$ function is the limit of a computable sequence.

**Lemma 1.10.11** (Limit Lemma)**.** *$f \leq_T A'$ if and only if there exists an $A$-computable sequence $\{f_s\}_{s\in\omega}$ such that $f = \lim_s f_s$.*

*Proof.* ($\Longrightarrow$): This follows from the Modulus Lemma relativized to $A$, since $f \leq_T A'$ if and only if $f$ is c. e. in $A$.

($\Longleftarrow$): Let $f = \lim_s f_s$ and let $h(m) = (\text{least } s)[(\forall t \geq s) f_t(m) = f_s(m)]$. Since $\{f_s\}_{s\in\omega}$ is computable in $A$, it follows that $h \leq_T A'$, so that $f \leq_T A'$ as well. □

For real numbers, we say that $r_x$ is *computably approximable* if $x$ is $\Delta^0_2$, so that $r$ is computably approximable if and only if it is the limit of a computable sequence of rationals.

For any $\Delta^0_2$ set $A$, it follows from the Jump Theorem that $\emptyset' \leq_T A' \leq_T \emptyset''$.

**Definition 1.10.12.** *Let $A \leq_T \emptyset'$; $A$ is* low *if $A' = \emptyset'$; $A$ is* high *if $A' = \emptyset''$.*

Clearly any computable set is low, whereas $\emptyset'$ is high. A c.e., noncomputable low set is constructed in Soare [181] (p. 111).

**Exercises**

1.10.1. The difference $B \setminus C$ of two c. e. sets is said to be a d. r. e. set. More generally, a set $C$ is $n$-r.e. if there is a computable sequence $\{A_s\}_{s\in\mathbb{N}}$ such that $A = \lim_s A_s$ and such that (i) $A_0 = \emptyset$ and (ii) for each $m$, $card(\{s : A_{s+1}(m) \neq A_s(m)\}) \leq n$. Show that for each $n$, there is an $(n+1)$-r.e. set which is not $n$-r.e.

## 1.11 The lattice of c. e. sets

The lattice $\mathcal{E}$ of c. e. sets is ordered by inclusion and has the natural operations of union and intersection. The lattice $\mathcal{E}^*$ is the quotient of $\mathcal{E}$ under equality modulo finite difference.

In this section, we consider properties of c. e. sets related to the lattice.

**Definition 1.11.1.** *(i) A set is* immune *if it is infinite but contains no infinite c. e. set;*

*(ii) A c. e. set $A$ is* simple *if $\mathbb{N} \setminus A$ is immune.*

Simple sets were first constructed by Post [155] as a partial solution to *Post's Problem*, which was to find natural intermediate c. e. degrees. It is easy to see that simple sets are neither computable nor $m$-complete.

**Definition 1.11.2.** *Let $R$ be a property of c. e. sets, that is $R \subseteq \mathcal{E}$.*

*(i) $R$ is* lattice-theoretic *or* invariant *in $\mathcal{E}$ ($\mathcal{E}^*$) if it is invariant under all automorphisms of $\mathcal{E}$ ($\mathcal{E}^*$).*

*(ii) $R$ is* elementary lattice-theoretic *or* definable *in $\mathcal{E}$ (respectively, $\mathcal{E}^*$) if there is a first-order formula $\varphi$ with one free variable in the language $\{\leq, \vee, \wedge, 0, 1\}$ of lattice theory such that $R(A)$ if and only if $\mathcal{E} \models \varphi(A)$ (resp. $\mathcal{E}^* \models \varphi(A)$).*

Clearly any definable property is also invariant.

**Lemma 1.11.3.** *The properties of computability and of finiteness are both definable in $\mathcal{E}$.*

*Proof.* $A$ is computable if and only if

$$\mathcal{E} \models (\exists y)[A \vee y = 1 \ \& \ A \wedge y = 0].$$

$A$ is finite if and only if

$$\mathcal{E} \models (\forall y)[y \subseteq A \longrightarrow A \text{ is computable}].$$

□

In the lattice $\mathcal{E}^*$, $A$ is simple if, for all $B \neq 0$, $A \cap B \neq 0$. Thus, the property of being simple is elementary lattice-theoretic in $\mathcal{E}^*$. The following lemma will imply that simplicity is also definable in $\mathcal{E}$.

**Lemma 1.11.4.** *If a property $R$ is preserved under finite differences, then $R$ is definable in $\mathcal{E}$ if and only if $R$ is definable in $\mathcal{E}^*$.*

*Proof.* Let $R$ be preserved under finite differences. If $R$ is definable in $\mathcal{E}$ by a formula $\varphi$, then the same formula works in $\mathcal{E}^*$. Suppose next that $R$ is definable in $\mathcal{E}^*$. Since finiteness is definable in $\mathcal{E}$, the relation $=^*$ of equality modulo finite difference is also definable in $\mathcal{E}$. Thus the definition from $\mathcal{E}^*$ may be rewritten in $\mathcal{E}^*$ by replacing all occurrences of $=$ with $=^*$. □

**Definition 1.11.5.** *(i) If $e = \sum_{i=0}^{k} e_i 2^i$, then $D_e = \{i \leq k : e_i = 1\}$. (Thus $D_0, D_1, \ldots$ effectively enumerates the finite sets of natural numbers.)*

*(ii) A sequence $\{F_n\}_{n \in \mathbb{N}}$ of finite sets is a* strong (weak) array *if there is a computable function $f$ such that $F_n = D_{f(n)}$ ($F_n = W_{f(n)}$).*

*(iii) An infinite set $B$ is* hyperimmune (h-immune) *(respectively,* hyperhyperimmune (hh-immune)*) if for any pairwise disjoint strong (respectively, weak) array, $F_n \cap B = \emptyset$ for some $n$.*

*(iv) A c. e. set $A$ is* hypersimple (h-simple) *(respectively,* hyperhypersimple (hh-simple)*) if $\mathbb{N} \setminus A$ is h-immune (respectively, hh-immune).*

It is easy to see that hh-simple implies h-simple and that h-simple implies simple.

**Definition 1.11.6.** *(i) A function $f$* majorizes *a function $g$ if $f(n) \geq g(n)$ for all $n$ and $f$* dominates *$g$ if $f(n) \geq g(n)$ for all but finitely many $n$.*

*(ii) The* principal function *$p_A$ of an infinite set $A$ is defined by $p_A(n) = a_n$, where $a_0 < a_1 < \cdots$ enumerates $A$ is increasing order.*

The following result is easy to prove.

**Theorem 1.11.7.** *An infinite set $A$ is hyperimmune if and only if no computable function majorizes $p_A$.* □

Dekker used this characterization to show that every nonzero c. e. degree contains a h-simple (hence simple) set.

A degree **a** is said to be *hyperimmune-free* if it does not contain any hyperimmune sets. A set $A$ is sometimes said to be *almost recursive* if its degree is hyperimmune-free, that is, any function computable in $A$ is dominated by some recursive function.

We will need the following result from Soare [181] (p. 85) in our study of $\Pi^0_1$ classes.

**Theorem 1.11.8.** *For any noncomputable c. e. set $B$, there is a simple, non-hypersimple c. e. set $A \equiv_T B$. Furthermore, for any c. e. set $C$ and any infinite set $D \leq_T C$, if $A \cap D = \emptyset$, then $B \leq_T C$.*

*Proof.* (based on [181], p. 85). The requirements are fourfold:

**(i)** $A$ is simple;

**(ii)** $A \leq_T B$;

**(iii)** $A$ is not hypersimple;

**(iv)** $B \leq_T A$.

Let $f$ be a one-to-one computable function with range $B$ and let $B_s = \{f(0), \dots, f(s)\}$. $A$ is enumerated in stages $A_s$, beginning with $A_0 = \emptyset$. Let $\mathbb{N} \setminus A_s = \{a^s_0 < a^s_1 < \dots\}$. There are two actions which may be taken at stage $s+1$.

*Step 1.* Here we take action to satisfy requirements (i) and (ii). For all $e \leq s$, attention is required if $W_{e,s} \cap A_s = \emptyset$ and

$$(\exists n)[n > 3e \ \&\ n \in W_{e,s} \ \&\ f(s+1) < n].$$

In this case, put into $A_{s+1}$ the least such $n$ corresponding to $e$. If no such $e$ exists, do nothing.

*Step 2* Here we take action to satisfy requirements (iii) and (iv) by putting $a_{3f(s+1)+1}$ into $A$.

We show that the requirements are satisfied.

(i) Suppose by way of contradiction that $W_e$ is infinite and $W_e \cap A = \emptyset$. Then here is an algorithm for testing $m \in B$, that is, find $s$ and $n > \max\{m, 3e\}$ such that $n \in W_{e,s}$; then by Step 1, $f(t) \geq n > m$ for all $t > s$, so that $m \in B \iff m \in B_s$.

(ii) We claim that $B_s \lceil n = B \lceil n$ implies $A_s \lceil n = A_s \lceil n$. To see this, let $m < n$ and $m \in A_{t+1} \setminus A_t$. Then in Step 1, we have $m > f(t+1)$, and in Step 2, we have $m = a^s_{3f(t+1)+1} > f(t+1)$, so that in either case $f(t+1) < n$ and $f(t+1) \in B_{t+1} \setminus B_t$. Thus to test $m \in A$, just compute from $B$ a stage $s$ such that $B_s \lceil m+1 = B \lceil m+1$ and then $m \in A \iff m \in A_s$.

(iii) Note that $|A \cap [0, 3e]| \leq 2e$, since at most $e$ elements $\leq 3e$ are put into $A$ under Step 1 (one from each $W_i$, $i < e$) and at most $e$ elements under Step 2 (one for each $i \in B$, $i < e$, since $a_{3f(s+1)+1} = a_{3i+1} > 3i$ for $i = f(s+1) \in B$). Thus $\mathbb{N} \setminus A$ is majorized by the function $3x$ and $A$ is not h-simple.

(iv) To test $m \in B$, use $A$ to compute $s$ such that $a^s_{3m+1} = \lim_t a^t_{3m+1}$; then $m \in B \iff m \in B_s$. Thus $B \leq_T A$.

Now let $C$ be a c. e. set and let $D$ be any infinite set such that $D \leq_T C$ and $A \cap D = \emptyset$. Let $D = \{d_0 < d_1 < \dots\}$. By the Modulus Lemma, there exists a uniformly computable double sequence $\{d^s_i\}_{i,s\in\mathbb{N}}$ such that $\lim_s d^s_i = d_i$ for all $i$ with a modulus of convergence computable in $C$.

Let $s(e) = f^{-1}(e)$ and use the Recursion Theorem to define a computable function $h(e)$ such that $W_{h(e)} = \emptyset$ if $e \notin B$ and otherwise

$$W_{h(e)} = \{d_0^{s(e)}, d_1^{s(e)}, \dots, d_{3(h(e))}^{s(e)}\}.$$

We may assume without loss of generality that $h(e) > e$ for all $e$. Use $C$ to compute a function $r(e)$ such that $d_i^{r(e)} = d_i$ for all $i \leq g(h(e))$. Let

$$\hat{B} = \{e \in B : e \notin B_{r(e)}\}.$$

There are two cases.

*Case* 1. $\hat{B}$ is finite. Then clearly $B \leq_T C$.

*Case* 2. $\hat{B}$ is infinite. Here is the procedure to test whether $b \in B$ using the function $r$. First find $e \in \hat{B}$ such that $3(h(e)) > b$. Since $e \in \hat{B}$, $r(e) < s(e)$ and therefore $d_i^{s(e)} = d_i$ for all $i \leq g(h(e))$, so that $W_{h(e)} \subseteq D \subseteq \mathbb{N} \setminus A$. It follows that $W_{h(e)}$ contains an element $u > 3h(e)$; let $s_b$ be a stage such that $u \in W_{h(e),s_k}$. It follows from Step 1 that $f(s) \geq u > b$ for all $s \geq s_k$, so that

$$b \in B \iff b \in B_{s_b}.$$

Since $s_b$ can be computed from $C$ uniformly in $b$, it follows that $B \leq_T C$. □

We will obtain a lattice-theoretic characterization due to Lachlan [111] of hhsimple using the following two lemmas. For any c. e. set $C$, let $\mathcal{L}(C)$ denote the lattice (under inclusion) of c. e. supersets of $C$.

**Lemma 1.11.9.** *(Lachlan) For any c. e. set $C$, if $\mathcal{L}(C)$ is a Boolean algebra, then $C$ is hh-simple.*

*Proof.* Suppose that $C$ is not hhsimple as witnessed by the disjoint weak array $\{W_{f(n)}\}_{n\in\mathbb{N}}$ and let $A = C \cup \bigcup_n (W_n \cap W_{f(n)})$. Suppose by way of contradiction that $W_n$ is the complement of $A$ in $\mathcal{L}(C)$ and choose $m \in W_{f(n)} \setminus C$. Since $m \notin C$, it follows that $m \in A \iff m \in W_n$, a contradiction. □

**Theorem 1.11.10** (Owings Splitting Theorem)**.** *Let $C \subseteq B$ be c. e. sets such that $B \setminus C$ is not co-c. e. Then there exists disjoint c. e. sets $A_0$ and $A_1$ (whose indices may be obtained uniformly from those of $B$ and $C$) such that*

*(i)* $B = A_0 \cup A_1$;

*(ii)* $A_i \setminus C$ *is not co-c.e. for* $i = 0, 1$;

*(iii) For any c. e. set $W$, if $C \cup (W \setminus B)$ is not c. e., then $C \cup (W \setminus A_i)$ is not c. e. for $i = 0, 1$.*

*Proof.* Let $f$ be a 1:1 computable function with range $B$ and let $\{C_s\}_{s\in\mathbb{N}}$ be any computable enumeration of $C$. We try to meet the requirements

$$P_s : f(s) \in A_{0,s} \cup A_{1,s}, s \in \mathbb{N}$$

and

$$R_{\langle e,i\rangle} : A_i \setminus C \neq \mathbb{N} \setminus W_e, \text{ for } i = 0, 1.$$

Requirement $R_{\langle e,i\rangle}$ requires attention at stage $s+1$ if $f(s+1) \in W_{e,s}$ and $f(s+1) \leq g(e,i,s)$.

**Stage** $s = 0$: Put $f(0) \in A_0$ and set $g(e,i,0 = 0$ for all $e, i$.

**Stage** $s+1$:

**Step 1**: If there exists $x \leq g(e,i,s)$ such that $x \in W_{e,s} \cap (A_{i,s} \setminus C_s)$, set $g(e,i,s+1) = g(e,i,s)$. Otherwise $g(e,i,s+1) = s+1$.

**Step 2**: Let $y = f(s+1)$ and choose the least $\langle e,i\rangle$ such that $R_{\langle e,i\rangle}$ requires attention at stage $s+1$. Then put $y \in A_{i,s+1}$. If no such $e, i$ exist, then put $y \in A_{0,s+1}$.

Let $A_i = \cup_s A_{i,s}$. Clearly $B = A_0 \cup A_1$.

To prove (ii), assume that $A_i \setminus C = \mathbb{N} \setminus W_e$. We must show that $B \setminus C$ is co-c. e. to obtain a contradiction. For $\langle e', i'\rangle < \langle e, i\rangle$, there are two possibilities. Either $\lim_s g(e', i', s) = z_{e',i'} < \infty$ so that after some stage $s$, we never put any $y \in A_i$ for the sake of $R_{\langle a,i\rangle}$, or $\lim_s g(e,i,s) = \infty$. Let $z$ be the maximum of the $z_{e',i'}$ and choose $s_0$ large enough so that for all $\langle e,i\rangle$ of the first type, $g(e', i', s)$ has already converged to $z_{e',i'}$ and such that $f(s) > z$ for all $s \geq s_0$. Define the c. e. set

$$V_e = \{m : (\exists s \geq s_0)[m \in W_{e,s} \setminus B_s \ \&\ m \leq g(e,i,s)\}.$$

Now $V_e \setminus B = W_e \setminus B$ since $\lim_s g(e,i,s) = \infty$, so that in fact $\mathbb{N} \setminus B \subseteq V_e$, that is, if $m \notin B$, then also $m \notin A_i$, so that by our assumption, $m \in W_e$ and thus $m \in V_e \setminus B$.

Also $V_e \cap (B \setminus C) = \emptyset$ (and hence $V_e \subseteq C \cup (\mathbb{N} \setminus B)$) by the following. Let $m \in V_e \cap B$ and take $s \geq s_0$ such that $m \leq g(e,i,s)$ and $m \in W_{e,s} \setminus B_s$. Then $m \in B \setminus B_s$ so that $m = f(t)$ for some $t > s_0$. Now at stage $t+1$, there exists $\langle e', i'\rangle \leq \langle e, i\rangle$ such that $\lim_r g(e', i', r) = \infty$ and $m$ is put into $A_{i'}$ at stage $t+1$. But $m$ cannot be a permanent witness for $\langle e', i'\rangle$, and therefore $m \in C$.

It follows that

$$\mathbb{N} \setminus (B \setminus C) = C \cup (\mathbb{N} \setminus B) = C \cup V_e.$$

□

**Theorem 1.11.11.** *(Lachlan) For any c. e. set $C$, $C$ is hhsimple if and only if $\mathcal{L}(C)$ is a Boolean algebra.*

*Proof.* The direction ($\longleftarrow$) follows from Lemma 1.11.9. For the other direction, suppose that $B$ is not complemented in $\mathcal{L}(C)$, that is $B \setminus C$ is not co-c. e. Apply Theorem 1.11.10 to obtain $A_0$ and $A_1$. Let $W_{g(0)} = A_0$ and apply Theorem 1.11.10 to $A_1$ and $C \cap A_1$ to obtain $A_0^1$ and $A_1^1$. Set $W_{g(1)} = A_0^1$ and continue in this fashion to obtain a pairwise disjoint sequence of c. e. sets $W_{g(n)}$ such that $W_{g(n)} \setminus C$ is not co-c. e. and hence is non-empty for all $n$. Finally, it is easy to uniformly define finite subsets $W_{f(n)} \subseteq W_{g(n)}$ such that $W_{f(n)} \setminus C \neq \emptyset$ for each $n$. That is, given a c. e. set $W = W_{g(n)}$ such that $W \setminus C \neq \emptyset$, let $W_{f(n),s}$ contain $i$ if $i \in W_s$ and $(\forall j < i)[j \in W_s \to j \in C_s]$. (Thus if $j \in W_s \setminus C$, then no $i > j$ can enter $W_{f(n)}$ after stage $s$, so that $W_{f(n)}$ is finite, as desired. If $j$ is the least element of $W \setminus C$, then $j$ will be put into $W_{f(n)}$ as soon as all elements $i < j$ of $W$ have come into $C$.) □

**Definition 1.11.12.** *An infinite set $C$ is* cohesive *if there is no c. e. set $W$ such that $W \cap C$ and $C \setminus W$ are both infinite. A c. e. set $A$ is* maximal *if for any c. e. set $B \supseteq A$, either $B$ is cofinite or $B \setminus A$ is finite.*

Thus $A$ is maximal if and only if its complement is cohesive.

From the lattice viewpoint, $A$ is maximal if it is as large as possible in $\mathcal{E}^*$ without being trivial. Friedberg first constructed a maximal c. e. set in [64]. The proof is based on the following notion.

**Definition 1.11.13.** *The* $e$-state *of a number $m$ is $\{i \leq e : m \in W_i\}$ and the $e$-state at stage $s$ is $\{i \leq e : m \in W_{i,s}\}$.*

The $e$-states are ordered lexicographically so that $m$ has a higher $e$-state than $n$ if there is some $j < e$ such that $m \in W_j$ but $n \notin W_j$ and for all $i < j$, $m \in W_i \iff n \in W_i$. Note that, for each $e$, there are exactly $2^{e+1}$ possible $e$-states.

**Theorem 1.11.14** (Friedberg)**.** *There exists a maximal c. e. set $A$.*

*Proof.* Let $\sigma(e, m, s)$ denote the $e$-stage of $m$ at stage $s$. We define the cohesive set $C$ in stages $C^s$ so that

$$C_s = \{c_0^s < c_1^s < \cdots <\}.$$

Then $c_i = \lim_s c_i^s$ and $C = \{c_i : i \in \mathbb{N}\}$.

Initially $c_i^0 = i$ for all $i$. The construction proceeds in stages with the goal of making $\sigma(e, c_i) \geq \sigma(e, c_j)$ for all $e < i < j$.

At stage $s + 1$, choose the least $e$ such that for some $i$ with $e < i \leq s$, $\sigma(e, c_i^s, s+1) > \sigma(e, c_e^s, s+1)$ For this $e$, choose the least such $i$ and let $c_e^{s+1} = c_i^s$. For $j < e$, $c_j^{s+1} = c_j^s$ and for all $j$, $c_{e+j}^{s+1} = c_{i+j}^s$. If no such $e$ exists, then $c_i^{s+1} = c_i^s$ for all $i$.

**Claim 1**: For every $e$, $\lim_s c_e^s = c_e$ exists.

Proof of Claim 1: The proof is by induction on $e$, so we may suppose that it holds for all $i < e$ and take $s_0$ so that $c_i^s = c_i$ for all $i < e$ and all $s \geq s_0$. Then for any $s > s_0$, if $c_e^{s+1} > c_e^s$, it follows that $\sigma(e, c_e^{s+1}, s + 1) > \sigma(e, a_e^s, s)$.

But there are only $2^{e+1}$ different $e$-states, so this can happen at most $2^{e+1}-1$ times after stage $s_0$, after which the $e$-state has converged and $c_e^s$ cannot change again.

It follows from Claim 1 that $C$ is coinfinite. It is clear from the construction that $c \in C \iff (\forall s) c \in C_s$, so that $C$ is a co-c. e. set.

**Claim 2**: For all $e \leq i$, $\sigma(e, c_e) \leq \sigma(e, c_i)$.

Proof of Claim 2: Assume by induction that the Claim holds for all $d < e$. Fix $e < i$ and let $s$ be large enough so that $c_j^s = c_j$ for all $j \leq i$. Suppose by way of contradiction that $\sigma(e, c_e) < \sigma(ec_, i)$ and choose $t > s$ such that $\sigma(e, c_e, t) = \sigma(e, c_e)$ and $\sigma(e, c_i, t) = \sigma(e, c_i)$. Then at stage $t+1$, the construction will force $c_e^{t+1} \neq c_e^t$, a contradiction.

**Claim 3**: For each $e$, there is some $k_e$ such that for all $i, j \geq k_e$, $c_i \in W_e$ if and only if $c_j \in W_e$.

Proof of Claim 3: Note that since the $e$-state of any $c$ is an initial segment of the $i$-state for any $i \geq e$ it follows that for $e \leq i < j$, $\sigma(e, c_i) \geq \sigma(e, c_j)$. Since there are only finitely many $e$-states, there must be some $k_e$ such that $\sigma(e, c_i) = \sigma(e, c_j)$ for all $i, j \geq k_e$. Claim 3 now follows.

**Claim 4**: For all $e$, either $C \cap W_e$ is finite are $C \setminus W_e$ is finite.

Proof of Claim 4: Fix $e$ and suppose that $W_e \cap C$ is infinite. Then there must be some $i \geq k_e$ such that $c_i \in W_e$. It follows from Claim 3 that $c_j \in W_e$ for all $j \geq k_e$, so that $C \setminus W_e \subseteq \{c_0, \ldots, c_{j-1}\}$ is finite. □

Martin proved the following result connecting high degrees and maximal sets in [130].

**Theorem 1.11.15.** *(Martin) A degree* **d** *is high if and only if there exists a maximal c. e. set $A$ of degree* **d**. □

**Exercises**

1.11.1. Prove Theorem 1.11.7.

1.11.2. Show that any hypersimple set is simple.

1.11.3. Prove that any maximal set must have high degree. Hint: Show that the principal function $p_a$ is *dominant*, that is, for any computable function $f$, $f(m) \leq p_a(m)$ for almost all $m$.

## 1.12 Computable ordinals and the analytical hierarchy

**Definition 1.12.1.** *(i) A relation $P$ is said to be $\Pi^1_0$ (and also $\Sigma^1_0$) if $P$ is arithmetical.*

*(ii) A relation $P$ is said to be $\Pi^1_{n+1}$ if there is a $\Sigma^1_n$ relation $R$ such that*

$$P(\overrightarrow{m}, \overrightarrow{x}) \iff (\exists y) R(\overrightarrow{m}, \overrightarrow{x}, y).$$

*(iii) A relation $S$ is $\Sigma^1_n$ if the complement if $\Pi^1_n$.*

*(iv) $S$ is $\Delta^1_n$ if it is both $\Sigma^1_n$ and $\Pi^1_n$.*

The relativized notions of $\Sigma^1_n[z]$ and $\Pi^1_n[z]$ are similarly defined. A relation is said to be *analytic* (resp. coanalytic, Borel) if it is $\Sigma^1_1[z]$ (resp. $\Pi^1_1[z]$, $\Delta^1_1[z]$) for some $z$.

In this section, we will focus on $\Sigma^1_1$ and $\Pi^1_1$ sets and relations.

We first need to consider normal forms for $\Sigma^0_1$, $\Pi^0_1$, $\Sigma^1_1$ and $\Pi^1_1$ relations and their connection with trees.

Recall that a relation $S \subseteq \mathbb{N}^k \times \mathbb{N}^{\mathbb{N}\ell}$ is $\Sigma^0_1$ provided that there is a computable relation $R$ such that $S(\overrightarrow{m}, \overrightarrow{x}) \iff (\exists n)R(n, \overrightarrow{m}, \overrightarrow{x})$.

**Proposition 1.12.2.** *A relation $S \subseteq \mathbb{N}^k \times \mathbb{N}^{\mathbb{N}\ell}$ is $\Sigma^0_1$ provided that there is a computable relation $A$ such that $S(\overrightarrow{m}, \overrightarrow{x}) \iff (\exists n)A(\overrightarrow{m}, \overrightarrow{x}\lceil n)$.*

*Proof.* Given the computable relation $R$ from the definition, we know from the Use Principle (Theorem 1.6.6) that if $R(n, \overrightarrow{m}, \overrightarrow{x})$, then there is some maximum use $u$ of each $\overrightarrow{x}$ so that $R(n, \overrightarrow{m}, \overrightarrow{x}\lceil u)$ and then we can let $A(\overrightarrow{m}, \overrightarrow{x}\lceil u)$ if $(\exists n < u)R(\overrightarrow{m}, \overrightarrow{x}\lceil u)$. □

For simplicity, we will now consider $\Pi^0_1$, and more generally, closed sets of reals. A $\Pi^0_1$ subset of $\mathbb{N}^{\mathbb{N}}$ is also called an effectively closed set.

xxx – put this in an earlier section – xxx

A subset $T$ of $\mathbb{N}^*$ is said to be a *tree* if it is closed under initial segments.

**Proposition 1.12.3.** *$P \subseteq \mathbb{N}^{\mathbb{N}}$ is closed if and only if there is a tree $T \subseteq \mathbb{N}^*$ such that $P = [T]$.*

zzzz

xxx

**Lemma 1.12.4.** *For any coanalytic relation $S$, there is a relation $R$ such that*

$$S(\overrightarrow{m}, x) \iff (\exists y)(\forall n)R(y\lceil n, \overrightarrow{m}, c\lceil n).$$

*If $S$ is $\Sigma^1_1$, t hen $R$ may be taken to be computable.*

*Proof.* It is clear that the family of relations expressible in this form includes the computable relations and it will suffice to show that this family is closed under number quantification and under existential function quantification. Given $S$ in this form,

$$(\exists i)S(i, \overrightarrow{m}, \overrightarrow{x}) \iff (\exists z)(\forall n)R(z(0), \langle z(1), \ldots, z(n)\rangle, \overrightarrow{m}, \overrightarrow{x}).$$

Also

$$(\forall i)S(i, \overrightarrow{m}, \overrightarrow{x}) \iff (\exists z)(\forall n)R(i, \langle z(2^i), \ldots, z(2^i(2n+1))\rangle, \overrightarrow{m}, \overrightarrow{x}).$$

Finally,

$$(\exists u)S(\overrightarrow{m}, \overrightarrow{x}, u) \iff (\exists z)(\forall n)R(\langle z(1), \ldots, z(2n-1)\rangle, \overrightarrow{m}, \overrightarrow{x}, (z(0), z(2), \ldots)).$$

□

There are two classic examples here.

**Example 1.12.5.** *A set $A \subseteq \mathbb{N}$ may code a partial ordering $\leq_A$, where $m \leq_A n \iff \langle m,n \rangle \in A$. Also, we write $m <_A n \iff m \leq_A n \ \& \ \neg n \leq_A m$.*

$$WO = \{A : \leq_A \ \text{is a well-ordering}\}.$$

*Let $LO$ be the set of linear orderings. It is easy to see that $LO$ is $\Pi^0_1$. Note for example that $\leq_A$ is transitive if and only if*

$$(\forall i)(\forall j)(\forall k)[(i \leq_A j \ \& \ j \leq_A k) \rightarrow i \leq_A k)],$$

*which is a $\Pi^0_1$ condition. Now a linear ordering is a well-ordering if and only if it is well-founded, that is, has no infinite descending chain. Thus $WO$ is a $\Pi^1_1$ class, since $\leq_A$ is well-founded if and only if*

$$(\forall x)[(\forall m)(x(m+1) \leq_A x(m)) \rightarrow (\exists m)(x(m) \leq_A x(m+1))].$$

*We similarly define the $\Pi^0_1$ class*

$$PWO = \{A : \leq_A \ \text{is a pre-well-ordering}\}.$$

*We may also define the following $\Sigma^1_1$ relation $A \precsim (\prec) B$ to mean that $\leq_B$ is a (pre)-linear ordering and $\leq_A$ is isomorphic to a (proper) subordering of $\leq_B$. Here the subordering property can be expressed as*

$$(\exists x)(\forall p)(\forall q)[p \leq_A q \iff x(p) \leq_B x(q)].$$

*For the proper subordering add the following clause*

$$(\exists r)(\forall p)(x(p) <_B r).$$

*Observe that if $\leq_A$ and $\leq_B$ are linear orderings and $B \in WO$ and $A \precsim B$, then $A \in WO$. A similar result holds for pre-orderings.*

*The order type $\|R\|$ of a well-ordering in $WO$ is the unique ordinal $\rho$ such that $(Fld(R), \leq_R)$ is isomorphic to the standard ordering $(\rho, \in)$, where $Fld(R) = dom(R) \cup ran(R)$. For a pre-well-ordering, the* norm *$\|R\|$ of $R$ is the unique ordinal $\alpha$ such that there is an order-preserving map from $Fld(R)$ onto $\alpha$.*

In a certain sense, the ordering relation $\precsim$ on $WO$ is $\Delta^1_1$. Let $\|A\| = \aleph_1$ if $A \notin WO$.

**Lemma 1.12.6.** *For any linear orderings $A$ and $B$, if $A \notin WO$ and $B \in WO$, then $\neg A \precsim B$.*

*Proof.* Suppose that $f$ were an embedding of $A$ into $B$ and let $a_0 >_A a_1 > \cdots$ be a descending chain in $A$. Then $f(a_0) >_B f(a_1) >_B \cdots$, so that $A \notin WO$. $\square$

The following are left as exercises.

**Lemma 1.12.7.** *For $A, B \in WO$:*

*(i) Either* $A \precsim B \vee B \prec A$*;*

*(ii)* $A \precsim B$ *if and only if* $\|A\| \leq \|B$*;*

1. *(iii)* $A \prec B$ *if and only if* $\|A\| < \|B$*;*

**Theorem 1.12.8** (Prewellordering Theorem)**.** *For all pre-linear orderings* $A$ *and* $B$*, if either* $A$ *or* $B$ *is in* $WO$*, then*

*(i)* $A \precsim B \vee B \notin WO \iff [A \in WO \ \& \ \|A\| \leq \|B\|] \iff \neg B \precsim A \ \& \ B \in WO$*;*

*(ii)* $A \prec B \vee B \notin W \iff [A \in WO \ \& \ \|A\| < \|B\|] \iff \neg B \prec A \ \& \ A \in WO$*.*

A related example is the following.

**Example 1.12.9.** $T \subseteq \mathbb{N}$ *is said to be a* tree *if* $\{\sigma \in \mathbb{N}^* : \langle\sigma\rangle \in T\}$ *is closed under initial segments.*

Then the set $WF$ of well-founded trees is $\Pi^1_1$ since

$$T \in WF \iff (\forall x)(\exists n) x\lceil n \notin T.$$

An ordinal may be associated with a well-founded tree by means of the Brouwer-Kleene linear ordering $\leq_{KB}$ on $\mathbb{N}^*$, where

$$\sigma \leq_{KB} \tau \iff (\tau \preceq \sigma) \vee (\exists j)[\sigma(j) < \tau(j) \ \& \ (\forall i < j)\sigma(i) = \tau(i)].$$

Now given a well-founded tree $T$, let

$$F(T) = \{\langle\langle\sigma\rangle, \langle\tau\rangle\rangle : \sigma \in T \ \& \ \tau \in T \ \& \ \sigma \leq_{KB} \tau\}.$$

**Lemma 1.12.10.** $T$ *is a well-founded tree if and only if* $F(T)$ *is a well-ordering.*

*Proof.* It is easy to see that $\leq_{KB}$ is a linear ordering on $\mathbb{N}^*$ (see the exercises). If $T$ is not well-founded, then there is an infinite path $y$ through $T$ and $\{y\lceil n : n \in \mathbb{N}\}$ provides an infinite descending $\leq_{KB}$ chain in $F(T)$. On the other hand, suppose that $F(T)$ is not well-founded and let $\{\sigma_i\}_{i\in\omega}$ be a descending $\leq_{KB}$ chain in $F(T)$. Then we can define by recursion an infinite path through $T$. For all $i > 0$, $|\sigma| > 0$ and $\sigma_{i+1}(0) \leq \sigma_i(0)$ since $\sigma_{i+1} \leq_{KB} \sigma_i$. Thus $y(0) = \lim_i \sigma_i(0)$ exists. Now if $\sigma_j(0) = y(0)$, then we have $\sigma_{j+1}(1) \leq \sigma_j(1)$, so that $y(1) = \lim_i \sigma_i(1)$ also exists. Proceeding by recursion we can define a sequence $y(n) = \lim_i \sigma_i(n)$. Now for each $n$, there is some $j$ such that $y(i) = \sigma_j(i)$ for all $i < n$, so that $y\lceil n \preceq \sigma_j$ and therefore $y\lceil n \in T$ for all $n$. □

For a computable well-founded tree, $F(T)$ is of course a computable ordinal, since $\leq_{KB}$ is a computable relation.

The well-ordering $F(T)$ of a well-founded tree $T$ is closely related to the *rank* $rk(T)$, defined inductively as follows.

**Definition 1.12.11.** *For any non-empty well-founded tree $T \subseteq \mathbb{N}^*$ and any $\sigma \in T$, the rank $rk_T(\sigma)$ of $\sigma \in T$ is given by*

$$rk_T(\sigma) = sup\{rk_T(\sigma^\frown i) + 1 : \sigma^\frown i \in T\}.$$

*Then the rank $rkt(T) = rk_T(\emptyset)$. If $T$ is not well-founded, then $ht(T) = \infty$.*

**Lemma 1.12.12.** *For any non-empty well-founded tree $T$, $rk(T) \leq \|F(T)\| \leq \omega^{rk(T)} + 1$.*

*Proof.* These inequalities are proved by induction on $rk(T)$. For the base case, $rk(T) = 0$ if and only if $T = \{\emptyset\}$, which is if and only if $\|F(T)\| = 1$. Now for $rk(T) > 0$, the ordering $F(T)$ consists of $\omega$ blocs $T((0)), T((1)), \ldots$ followed by the largest element $\emptyset$. By induction $rk(T((i))) \leq \|F(T((i))\| \leq \omega^{rk(T((i)))} + 1$ for each $i$. The first inequality is immediate. For the second, we have

$$\|F(T)\| \leq (\omega^{rk(T((0)))} + 1 + \omega^{rk(T((1)))} + 1 + \ldots) + 1.$$

There are two cases.

(Case 1): There is a fixed $m$ such that $rk((T((i))) \leq rk(T((m)))$ for each $i$, so that $rk(T) = rk(T((m))) + 1 = \alpha + 1$. Then $\|F(t((m)))\| \leq \omega^\alpha + 1$ for each $m$ and hence

$$\|F(T)\| \leq \omega \cdot \omega^\alpha + 1 \leq \omega^{\alpha+1} + 1.$$

(Case 2): There is no maximum $rk(T((m)))$. Let $rk(T) = \alpha$ and, for each $m$, let $rk(T((m))) = \alpha_m$, so that $\alpha = \sup\{\alpha_n : n \in \mathbb{N}\}$. For each $m$, there exists $n > m$ such that $\alpha_i < \alpha_n$ for all $i < n$ and thus

$$\|F(T((0)))\| + \|F(T((1)))\| + \cdots + \|F(T((n)))\| \leq \omega^{\alpha_0} + 1 + \cdots + \omega^{\alpha_n} = \omega^{\alpha_n} < \omega^\alpha,$$

so that

$$\|F(T)\| \leq \omega^\alpha + 1 = \omega^{rk(T)} + 1.$$

□

The next result follows from the Enumeration Theorem 1.6.5.

**Theorem 1.12.13.** *For each $n$, there is universal $\Pi^1_{n+1}$ and a universal $\Sigma^1_{n+1}$ set of numbers.* □

The sets $WO$ and $WF$ are both many-one complete for $\Pi^1_1$ sets. To see this, we first need a normal form for $\Sigma^1_1$ and $\Pi^1_1$ relations.

**Definition 1.12.14.** *An ordinal $\alpha$ is* computable *if there is a computable well-ordering $A$ such that $\|A\| = \alpha$. $W = \{e : \phi_e = \chi_A \quad \text{for some } A \in WO\}$ and $PW = \{e : \phi_e = \chi_A \quad \text{for some } A \in PWO\}$. The least noncomputable ordinal is denoted by $\omega_1^{CK}$ or just $\omega_1$.*

Later on, we will need the concept of a *system of notations* for a computable ordinal.

**Definition 1.12.15.** *A system of notations for a computable ordinal $\alpha$ is a map $o$ from $\omega \setminus \{0\}$ to $\kappa + 1$ such that each of the following relations is recursive:*

*(i)* $o(a)$ *is a limit ordinal;*

*(ii)* $o(b) = o(a) + 1$*;*

*(iii)* $o(a) < o(b)$*.*

**Lemma 1.12.16.** *Any computable ordinal $\alpha$ possesses a system of notations.*

*Proof.* Note that a computable well-ordering induces a mapping from $\omega \to \kappa$ with property (iii) but not with the other two properties. First observe that the computable ordinals form an initial segment of the ordinals (See the exercises.)

Now let $\alpha$ be an infinite countable ordinal and let $\lambda$ be the largest limit ordinal $< \alpha$. Let $\lambda = \omega \cdot \gamma$ and $\alpha = \lambda + n$ for some ordinal $\gamma$ and some finite $n$. Let $\leq_G$ denote a computable well-ordering of type $\gamma$ with domain $G \subseteq \omega$ and let $a \in G$ such that $|a|_G = \gamma$. Let

$$C = \{\langle g, i\rangle : g \in G \ \& \ i \in \mathbb{N} \ \& \ (a = \lambda \ \rightarrow \ i \leq n)\}.$$

$C$ is an infinite computable set, so computably isomorphic to $\omega$. Thus it suffices to define the desired map $o$ from $C$ to $\alpha$ by

$$o(\langle g, i\rangle) = \omega \cdot |g|_G + i.$$

To verify that this defines a system of notations, observe that for $a = \langle g, i\rangle$ and $b = \langle h, j\rangle$ in $C$,

**(i)** $o(a)$ is a limit ordinal if and only if $i = 0$.

$o(b) = o(a) + 1$ if and only if $g = h$ and $j = i + 1$.

$o(a) < o(b)$ if and only if either $g = h$ and $i < j$ or if $g <_G h$.

□

**Theorem 1.12.17.** *For all relations $P$, $P$ is $\Pi^1_1$ if and only if $P \leq_m WO$, which is if and only if $P \leq_m WF$. Furthermore, if $P \subseteq \mathbb{N}$, then $P$ is $\Pi^1_1$ if and only if $P \leq_m W$.*

*Proof.* It is clear that the $\Pi^1_1$ relations are closed under $\leq_m$, which gives one direction. Now let $R$ be $\Pi^1_1$ and let $P$ be a computable relation so that

$$P(\overrightarrow{m}, \overrightarrow{x}) \iff (\forall y)(\exists n)R(y\lceil n, \overrightarrow{m}, \overrightarrow{x}).$$

We may assume without loss of generality that

$$R(\tau, \overrightarrow{m}, \overrightarrow{x}) \ \& \ \sigma \preceq \tau \rightarrow R(\sigma, \overrightarrow{m}, \overrightarrow{x}).$$

by using bounded quantification if necessary. Then we may define a computable functional $F$ such that

$$F(\overrightarrow{m}, \overrightarrow{x}) = \{\sigma : \neg R(\sigma, \overrightarrow{m}, \overrightarrow{x})\},$$

and we claim that

$$P(\overrightarrow{m}, \overrightarrow{x}) \iff F(\overrightarrow{m}, \overrightarrow{x}) \in WF.$$

It is clear that a witness $y$ to the fact that $\neg P(\overrightarrow{m}, \overrightarrow{x})$ will also be a witness to the fact that $F(\overrightarrow{m}, \overrightarrow{x})$ is not well-founded, and vice versa. Thus $P \leq_m WF$.

It remains to check that $WF \leq_m WO$. It follows from Lemma 1.12.10 that $T$ is well-founded if and only if $F(T)$ is a well-ordering. Now for $P \subseteq \mathbb{N}$, let $F$ be a computable function such that $m \in P \iff R_m \in WO$, where $R_m(p) \iff F(p, m) = 1$ and otherwise $F(p, m) = 0$. Let $f$ be a computable function such that $\phi_{f(m)}(p) = F(p, m)$. Then $m \in P \iff f(m) \in W$. □

The proof of this theorem also shows that $PWO$ is also $m$-complete for $\Pi^1_1$ sets.

**Corollary 1.12.18.** *None of the sets $W$, $PW$, $WO$, $PWO$ and $WF$ are $\Sigma^1_1$.*

*Proof.* Since these sets are $m$-complete, if they were $\Sigma^1_1$, then all $\Pi^1_1$ sets would be $\Sigma^1_1$. (See exercise 4 below. □

**Theorem 1.12.19** (Selection Theorem). *Any $\Pi^1_1$ relation $R$ has a partial selection function $Sel_R$ with a $\Pi^1_1$ graph such that*
$(\exists n)R(n, \overrightarrow{m}, \overrightarrow{x}) \iff R(Sel_R(\overrightarrow{m}, \overrightarrow{x}), \overrightarrow{m}, \overrightarrow{x}) \iff Sel_R(\overrightarrow{m}, \overrightarrow{x}) \downarrow$.

*Proof.* Let $R$ be reducible to $W$ by the function $F$ and let

$$Sel_R(\overrightarrow{m}, \overrightarrow{x}) = b \iff R(b, \overrightarrow{m}, \overrightarrow{x}) \ \& \ (\forall a)[F(a, \overrightarrow{m}, \overrightarrow{x}) \preceq F(b, \overrightarrow{m}, \overrightarrow{x}) \to b \leq a].$$

□

**Theorem 1.12.20** (Boundedness Theorem). *(Spector, [184])*

*(i) If $S$ is a $\Sigma^1_1$ subset of $PWO$, then $sup\{\|A\| : A \in S\} < \omega_1$;*

*(ii) If $S$ is a $\Sigma^1_1$ subset of $PW$, then $sup\{\|\phi_e\| : e \in S\} < \omega_1$.*

*Proof.* Suppose that $S$ were a counterexample to (i). Then

$$e \in W \iff (\exists c)[c \in S \ \& \ \phi_e \preceq \phi_c],$$

contradicting Corollary 1.12.18.

If $S$ were a counterexample to (ii), then $\{\phi_c : c \in S\}$ would be a counterexample to (i). □

Let $WO_\alpha = \{R \in WO : \|R\| < \alpha\}$ and $W_\alpha = \{c : \phi_c \in WO_\alpha\}$. Similarly, $PWO_\alpha = \{R \in PWO : \|R\| < \alpha\}$ and $PW_\alpha = \{c : \phi_c \in PWO_\alpha\}$

**Theorem 1.12.21.** *For all $R \subseteq \mathbb{N}^k \times \mathbb{N}^{\mathbb{N}^l}$,*

*(i) $R$ is $\Delta^1_1$ if and only if $R \leq_m WO_\alpha$ for some $\alpha < \omega_1$.*

*(ii) $R$ is $\Delta^1_1$ if and only if $R \leq_m W_\alpha$ for some $\alpha < \omega_1$.*

*Proof.* Let $B \in WO$ such that $\|B\| = \alpha$. Then

$$A \in WO_\alpha \iff A \precsim B \iff (A \in WO \ \& \ \neg(B \precsim A).$$

Thus $WO_\alpha$ is $\Delta^1_1$.

Now let $R \subseteq \mathbb{N}$ be $\Delta^1_1$, let $F$ be a computable functional such that $R(x) \iff F(x) \in WO$ and define the $\Sigma^1_1$ set $Q$ by

$$Q = \{F(x) : R(x)\}.$$

Then $Q \subseteq WO$, so by the Boundedness Theorem, there exists $\alpha < \omega_1$ such that $Q \subseteq WO_\alpha$.

The proof of (ii) is similar. □

Here is a surprising corollary to the Boundedness Principle.

**Theorem 1.12.22.** *(i) For any $\Sigma^1_1$ pre-well-ordering relation R, $\|R\| < \omega_1$;*

*(ii) There is a $\Pi^1_1$ well-ordering of order type $\omega_1$.*

*Proof.* (i) Let $\|R\| = \alpha$. Then $A \in W_{\alpha+1} \iff A \precsim R$ and it is easy to see that this is a $\Sigma^1_1$ definition of $W_{\alpha+1}$, implying that $\alpha < \omega_1$ by the Boundedness Principle.

(ii) Define the $\Pi^1_1$ set $W^*$ to contain a unique index $c$ with $\|\phi_c\} = \alpha$ for each $\alpha < \omega_1$. That is,

$$c \in W^* \iff c \in W \ \& \ (\forall d < c)[\neg\phi_c \precsim \phi_d \vee \neg\phi_d \precsim \phi_c].$$

Then let

$$R(c, d) \iff c \in W^* \ \& \ d \in W^* \ \& \neg\phi_d \precsim \phi_c.$$

□

**Exercises**

1.12.1. Show that the computable ordinals form an initial segment of the countable ordinals.

1.12.2. Show that the property of coding a linear ordering is in fact $\Pi^0_1$.

1.12.3. Show that the Brouwer-Kleene ordering is a linear ordering.

1.12.4. Prove the Enumeration Theorem 1.12.13 for $\Pi^1_1$ sets and show that the universal $\Pi^1_1$ set cannot be $\Sigma^1_1$.

1.12.5. Prove Theorem 1.12.8.

1.12.6. The definition of a computable ordinal may be relativized to computability from a fixed oracle $A$. Give appropriate definitions for $W^A$ and $\omega_1^A$ and prove relativized versions of Theorems 1.12.20, 1.12.21 and 1.12.22.

## 1.13 Inductive Definability

Inductive definitions play a fundamental role in many areas of mathematics. We have already seen that the set of computable functions is defined inductively and of course the set of terms and formulas of a given language are also defined inductively. The formal notion of inductive definability was first given by Spector [186] and is fully developed by Moschovakis in his book [141].

An *operator* $\Gamma$ over a set $X$ is a a function from $\mathcal{P}(X)$ to $\mathcal{P}(X)$. $\Gamma$ is said to be *inclusive* if $Y \subseteq \Gamma(Y)$ for all $Y \subseteq X$. $\Gamma$ is said to be *monotone* if $\Gamma(Y) \subseteq \Gamma(Z)$ for all $Y \subseteq Z \subseteq X$. $\Gamma$ is said to be *inductive* if it is either inclusive or monotone. The operator $\Gamma$ inductively defines a subset $Cl(\Gamma)$ as follows. A sequence $\Gamma^\alpha$ of subsets of $X$ is defined recursively by $\Gamma^0 = \emptyset$, $\Gamma^{\alpha+1} = \Gamma(\Gamma^\alpha)$ and $\Gamma^\lambda = \bigcup_{\beta<\lambda} \Gamma^\beta$ for limit ordinals $\lambda$. The closure of $\Gamma$ is $Cl(\Gamma) = \bigcup_\alpha \Gamma^\alpha$. A set $Y$ is said to be a *fixed point* of $\Gamma$ if $\Gamma(Y) = Y$.

**Lemma 1.13.1.** *For any inductive operator $\Gamma$,*

*(i) For any ordinal $\alpha$, if $\Gamma^{\alpha+1} = \Gamma^\alpha$, then $\Gamma^\beta = \Gamma^\alpha = Cl(\Gamma)$ for all $\beta \geq \alpha$.*

*(ii) There exists $\alpha$ such that $Card(\alpha) \leq Card(X)$ such that $\Gamma^{\alpha+1} = \Gamma^\alpha$.*

*Proof.* (i) is easily proved by induction on $\beta$.

For (ii), let $\kappa$ be a cardinal and suppose that $\Gamma^{\alpha+1} \setminus \Gamma^\alpha$ is non-empty for all $\alpha < \kappa$. Then clearly $Card(\kappa) \leq Card(\Gamma^\kappa) \leq Card(X)$. $\square$

Now we can define the *closure ordinal* $|\Gamma|$ of $\Gamma$ to be the least ordinal $\alpha$ such that $\Gamma^{\alpha+1} = \Gamma^\alpha$. The following is immediate.

**Corollary 1.13.2.** *For any inductive operator $\Gamma$ over $X$, $Card(|\Gamma|) \leq Card(X)$ and $Cl(\Gamma) = \Gamma^{|\Gamma|}$.*

In particular, for $X = \mathbb{N}$, $|\Gamma| < \aleph_1$.

**Example 1.13.3.** *The $\Pi^1_1$ set $W$ can be given by a $\Pi^0_1$ monotone inductive definition. First define the computable function $\nu$ so that $\phi_{\nu(c,n)}$ defines a restriction of $\phi_c$ to elements* below $n$ *in the following sense. Recall that $i \leq_c j$ means that $\phi_c(i,j) = 1$ and $i <_c j$ means that $i \leq_c j$ but not $j \leq_c i$. Let*

$$\phi_{\nu(c,n)}(i,j) = \begin{cases} \phi_c(i,j), & \text{if } j <_c n \\ 0, & \text{otherwise.} \end{cases}$$

*Now let*

$$c \in \Gamma(A) \iff W_c = \emptyset \ \vee \ (\forall n)\nu(c,n) \in A$$

*It is clear that $c \in \Gamma^1$ if and only if $W_c = \emptyset$, which is if and only if $c \in W$ and $\|c\| = 0$. It follows by induction that $c \in \Gamma^{m+1} \iff \|c\| \leq m$ and hence $c \in \Gamma^\omega \iff \|c\| < \omega$ and in general, $c \in \Gamma^\alpha \iff (c \in W \ \& \ \|c\| < \alpha)$.*

*Thus we see that $|\Gamma| = \omega_1$ and $Cl(\Gamma) = W$.*

Note that $Cl(\Gamma)$ is a *fixed point* of $\Gamma$.

**Theorem 1.13.4.** *For any monotone operator $\Gamma$ over a set $X$, $Cl(\Gamma)$ is the least fixed point of $\Gamma$, that is,*

$$x \in Cl(\Gamma) \iff (\forall Z)[\Gamma(Z) = Z \to x \in Z].$$

*Proof.* Let $U = \{x : (\forall Z)[\Gamma(Z) = Z \to x \in Z]\}$. Then $x \in U$ implies $x \in Cl(\Gamma)$, since $Z = \Gamma$ satisfies $\Gamma(Z) = Z$. For the other direction, let $Z$ be any set such that $\Gamma(Z) = Z$. It can be seen by induction that $\Gamma^\alpha \subseteq Z$ for all $\alpha$. That is, certainly $\Gamma^0 = \emptyset \subseteq Z$. Then supposing $\Gamma^\alpha \subseteq Z$, we have $\Gamma^{\alpha+1} = \Gamma(\Gamma^\alpha \subseteq \Gamma(Z) = Z$. Finally, if $\Gamma^\beta \subseteq Z$ for all $\beta < \lambda$, then $\Gamma^\lambda \subseteq Z$. It follows that $\Gamma^\alpha \subseteq U$ for all $\alpha$ and hence $Cl(\Gamma) \subseteq U$. □

The monotone operator $\Gamma$ is said to be *finitary* if for all $x$ and $Y$, if $x \in \Gamma(Y)$, then there is a finite $Z \subseteq Y$ such that $x \in \Gamma(Z)$. For example, the operator which defines the set of formulas of propositional logic is finitary, since each new formula is generated by either one or two previously generated formulas.

**Lemma 1.13.5.** *If $\Gamma$ is a finitary monotone operator on $X$, then $|\Gamma| \leq \omega$.*

*Proof.* Suppose $x \in \Gamma(\Gamma^\omega)$ and let $Z \subseteq \Gamma^\omega$ be a finite set such that $x \in \Gamma(Z)$. Then there exists $n < \omega$ such that $Z \subseteq \Gamma^n$ and hence $x \in \Gamma^{n+1}$. Thus $\Gamma^{\omega+1} = \Gamma^\omega$. □

The complexity of an operator $\Gamma$ on $\mathbb{N}$ is given by the complexity of the relation $\{\langle m, A\rangle : m \in \Gamma(A)\}$. We will also consider operators on $\mathbb{N}$ with real *parameters.* That is, for example, a family $\{\Gamma_x : x \in \mathbb{N}^\mathbb{N}\}$ of operators over $\mathbb{N}$ will be computable if $\{\langle m, x, A\} : m \in \Gamma_x(A)\}$ is a computable relation.

**Lemma 1.13.6.** *Any $\Sigma^0_1$ operator is finitary.*

*Proof.* Let $\Gamma$ be a $\Sigma^0_1$ operator and let $R$ be a computable relation so that

$$m \in \Gamma(A) \iff (\exists k)R(m, A\lceil k).$$

Suppose $m \in \Gamma(A)$ and let $k$ be given as above so that $R(m, A\lceil k)$. Now let $Z = \{i : i < k \;\&\; i \in A\}$. Then $R(m, Z\lceil k)$ so that $m \in \Gamma(Z)$. □

**Theorem 1.13.7.** *If $\Gamma$ is a $\Sigma^0_1$ monotone operator over $\mathbb{N}$, then $|\Gamma| \leq \omega$ and $Cl(\Gamma)$ is a c. e. set.*

*Proof.* Let $\Gamma$ be a $\Sigma^0_1$ operator and let $R$ be a computable relation so that

$$m \in \Gamma(A) \iff (\exists k)R(m, A\lceil k).$$

It follows from lemmas 1.13.5 and 1.13.6 that $|\Gamma| \leq \omega$.

For the closure, we have the following

**CLAIM**: $m \in \Gamma(W_e) \iff (\exists k, s)R(m, W_{e,s}\lceil k)$.

Proof of Claim: If $m \in \Gamma(W_e)$, then $(\exists k)R(m, W_e\lceil k)$ and thus $R(m, W_{e,s}\lceil k)$ where $s$ is large enough so that $W_{e,s}\lceil k = W_e\lceil k$. If $R(m, W_{e,s}\lceil k)$, then $m \in \Gamma(W_{e,s})$ and since $\Gamma$ is monotone, $m \in \Gamma(W_e)$.

It follows that there is a computable function $\phi$ such that $\Gamma(W_e) = W_{\phi(e)}$. Now we can recursively define a function $\psi$ such that $\Gamma^n = W_{\psi(n)}$ by letting $\psi(0) = 0$ and then $\psi(n+1) = \phi(\psi(n))$. Finally, we have $m \in Cl(\Gamma) \iff (\exists n) m \in W_{\psi(n)}$. □

**Theorem 1.13.8.** *For any $n > 0$ and any $\Pi^1_n$ monotone operator $\Gamma$ over $\mathbb{N}$, $Cl(\Gamma)$ is $\Pi^1_n$.*

*Proof.* By the improved version of Theorem 1.13.4 (see Exercise 1 below), we have

$$\begin{aligned} m \in Cl(\Gamma) &\iff (\forall Z)[\Gamma(Z) \subseteq Z \to m \in Z] \\ &\iff (\forall Z)[(\forall m)(m \in \Gamma(Z) \to m \in Z) \to m \in Z], \end{aligned}$$

which is a $\Pi^1_1$ definition. □

In particular, the closure of any $\Pi^0_1$ monotone operator is $\Pi^1_1$. This can be reversed up to many-one reduction.

**Theorem 1.13.9.** *For any $\Pi^1_1$ $P \subseteq \mathbb{N} \times \mathbb{N}^{\mathbb{N}}$, there is a uniformly $\Pi^0_1$ monotone operator $\Gamma_x$ and a computable function $f$ such that*

$$P(m, x) \iff f(m) \in Cl(\Gamma_x).$$

*Proof.* Let $R$ be a computable relation such that

$$P(m, x) \iff (\forall y)(\exists k) R(m, y\lceil k, x).$$

Let

$$\langle m, \sigma\rangle \in \Gamma_x(A) \iff R(m, \sigma, x) \ \vee \ (\forall n)\langle m, \sigma\lceil n\rangle \in A.$$

Then it is easy to check that $P(m, x) \iff \langle m, \emptyset\rangle \in Cl(\Gamma_x)$. □

**Theorem 1.13.10.** *For any $\Pi^0_1$ inductive operator $\Gamma$, $|\Gamma| \leq \omega_1$ and $Cl(\Gamma)$ is $\Pi^1_1$.*

*Proof.* Let $R$ be a computable relation so that, for all $m$ and $A$,

$$m \in \Gamma(A) \iff (\forall k) R(m, A\lceil k).$$

We want to uniformly define the levels $\Gamma^\alpha$ of the inductive definition $\Gamma$ as follows. Let $i <_c j$ denote $\phi_c(i, j) = 1$ & $\phi_c(j, i) = 0$, so that if $c \in W$ and $\phi_c$ is the characteristic function of the set $C$, then $i <_c j$ means that $i <_C j$; we also let $\|i\|_c = \|i\|_C$. Now define

$$S(c, P) \iff (\forall m, i)(P(m, i) \iff (\exists j)[j <_c i \ \& \ m \in \Gamma(\{n : P(n, j)\})]).$$

It follows by induction that if $c \in W$ and $S(c, P)$, then, for all $i \in Fld(R)$,

$$P(m, i) \iff m \in \Gamma^{\|i\|_c}.$$

Now we have

$$m \in \Gamma^{\omega_1} \iff (\exists i, c)[c \in W \ \& \ (\forall P)(S(c,P) \ \to P(m,i))].$$

Thus $\Gamma^{\omega_1}$ is a $\Pi^1_1$ set, which we denote as $Q$. For each $c \in W$, let $Q_c$ denote $\Gamma^{\|c\|}$, so that as above, we have a $\Pi^1_1$ definition

$$m \in Q_c \iff (\exists i)(\forall P)[S(c,P) \ \to \ P(m,i)],$$

and we also have a $\Sigma^1_1$ definition

$$m \in Q_c \iff (\exists i, P)[S(c,P) \ \& \ P(m,i)].$$

We note that this part of the argument applies to any $\Delta^1_1$ operator.

It remains to show that $\Gamma(Q) \subseteq Q$. Suppose therefore that $m \in \Gamma(Q)$, so that

$$(\forall k)R(m, Q\lceil k).$$

and define the $\Sigma^1_1$ set $V$ by

$$a \in V \iff (\exists k)(\forall c)[(c \in W \ \& \ R(m, Q_c)) \to \phi_a \precsim \phi_c].$$

It follows that $V \subseteq W$ and hence $sup\{\|\phi_a\| : a \in V\} = \alpha < \omega_1$. Then $(\forall k)R(m, \Gamma^\alpha\lceil k)$, so that $m \in \Gamma^{\alpha+1}$ and thus $m \in Q$. □

A natural problem in connection with Theorem 1.13.4 is the nature of the family of fixed points of an inductive operator. Of course we have $\Gamma(\mathbb{N}) = \mathbb{N}$ by the assumption that $X \subseteq \Gamma(X)$, so that we will not have a unique fixed point. However, we can refine the previous result to show that

The Boundedness Principle for inductive definability is needed for the analysis of the Cantor-Bendixson derivative in Chapter 4. The following prewellordering theorem for inductively definable sets is due to Kunen; see [141], p. 27.

The prewellordering associated with any inductive operator $\Gamma$ is induced by the following norm:

$$|x|_\Gamma = \begin{cases} (\text{least } \alpha)x \in \Gamma^{\alpha+1}, & \text{if } \ x \in Cl(\Gamma), \\ \infty, & \text{otherwise.} \end{cases}$$

The following Stage Comparison Theorem is due independtly to P. Aczel and K. Kunen; see [141] for a more general result.

**Theorem 1.13.11.** *Let $\Delta$ be a $\Delta^1_1$ monotone inductive operator. Then the following relations are both $\Pi^1_1$.*

$$R(m,n) \iff |m|_\Delta \leq |n|_\Delta \ \& \ m \in Cl(\Delta);$$
$$S(m,n) \iff |m|_\Delta < |n|_\Delta \ \& \ m \in Cl(\Delta).$$

*Proof.* Define a $\Delta_1^1$ monotone operator $\Lambda$ by

$$(0,m,n) \in \Lambda(A) \iff m \in \Delta(\{i : 1,i,n) \in A\});$$
$$(1,m,n) \in \Lambda(A) \iff n \notin \Delta(\{j : (0,m,j) \notin A\}).$$

It can be checked that $R(m,n) \iff (0,m,n) \in Cl(\Lambda)$ and $S(m,n) \iff (1,m,n) \in Cl(\Lambda)$. □

**Theorem 1.13.12.** *For any $\Delta_1^1$ monotone inductive operator $\Delta$ and any $\Sigma_1^1$ set $A \subseteq Cl(\Delta)$, there is a computable ordinal $\alpha$ such that $A \subseteq \Delta^\alpha$.*

*Proof.* Let $A$ be a $\Sigma_1^1$ subset of $Cl(\Delta)$ as described. Then by Theorem 1.13.11, the prewellordering $R_A$ defined by

$$R_A(m,n) \iff |m|_\Delta \leq |n|_\Delta \;\&\; n \in A$$

is a $\Sigma_1^1$ prewellordering. It now follows from Theorem 1.12.22(a) that $|R_A| < \omega_1^{CK}$. □

**Corollary 1.13.13.** *For any $\Delta_1^1$ monotone inductive operator $\Delta$, $|\Delta| \leq \omega_1^{C-K}$.*

**Exercises**

1.13.1. Improve Theorem 1.13.4 by showing that $Cl(\Gamma) = \bigcap\{Z \subseteq X : \Gamma(Z) \subseteq Z\}$.

1.13.2. Show that a non-monotone $\Sigma_1^0$ operator need not have a $\Sigma_1^0$ closure and in particular can have a $\Pi_n^0$ complete closure for any $n$.

1.13.3. Show that a non-monotone $\Pi_2^0$ operator may have a closure which is not $\Pi_1^1$.

1.13.4. Show that there is a $\Sigma_1^0$ inductive operator $\Gamma$ such that the set of true sentences of arithmetic is many-one reducible to $Cl(\Gamma)$.

1.13.5. Show that if $\Gamma$ is a $\Delta_1^1$ inductive operator and $A \subseteq \Gamma^{\omega_1}$ is $\Sigma_1^1$, then for some $\alpha < \omega_1$, $A \subseteq \Gamma^\alpha$. Then show that if $\Gamma$ is $\Pi_1^0$, then $Cl(\Gamma)$ is $\Delta_1^1$ if and only if $|\Gamma| < \omega_1$.

## 1.14 The hyperarithmetical hierarchy

In our study of the Cantor-Bendixson derivative $D(P)$ of a $\Pi_1^0$ class $P$, we will see that the iterated derivative $D^\alpha(P)$ for an infinite, computable ordinal $\alpha$ is a hyperarithmetical, or effectively Borel set. Since index sets for $\Pi_1^0$ classes and for hyperarithmetical sets will be important in this work, we will present a definition of the hyperarithmetical sets based on indices. This approach is based on that given by Hinman ([80], p. 163).

Informally, a set is $\Sigma_\omega^0$ if it is the union of an effective sequence $A_n$ of arithmetical sets and more generally a set is $\Sigma_\lambda^0$ for a computable ordinal $\lambda$ if it is the union of an effective sequence of sets, each of which is $\Sigma^0_\alpha$ for some $\alpha < \lambda$. As for the arithmetical hierarchy, a set is $\Pi^0_\alpha$ if its complement is $\Sigma^0_\alpha$ and is $\Sigma^0_{\alpha+1}$ if it is the union of an effective sequence of $\Pi^0_\alpha$ sets.

The following is an inductive definition of the hyperarithmetic sets $H_e$, taken essentially from Hinman ([80], p. 163).

First we define a set of ordinal notations.

**Definition 1.14.1.** *$H$ is the smallest subset of $\mathbb{N}$ such that for all $a$,*

*(i)* $\langle 7, a\rangle \in H$*;*

*(ii) if* $\phi_a(n) \in H$ *for all* $n$*, then* $a \in H$*.*

We observe that $H$ is the closure of a $\Pi^0_1$ monotone inductive operator $\Gamma_H$ and thus each $a \in H$ is assigned an ordinal $\alpha = \|a\|_H$ so that $a \in H^{\alpha+1} - H^\alpha$. It follows from Theorem 1.13.8 that $H$ is $\Pi^1_1$ and that each ordinal $\|a\|_H$ is computable.

Then each $a \in H$ is assigned a hyperarithmetic set by the following. (For $a \notin H$, we let $H_a = \emptyset$.)

**Definition 1.14.2.** *Let* $a \in H$*. Then*

*(i) If* $a = \langle 7, b\rangle$*, then* $H_a = W_b$*;*

*(ii) if* $\phi_a$ *is total, then* $H_a = \cup_n \mathbb{N} \setminus H_{\phi_a(n)}$*.*

For example, let $B$ be a $\Sigma^0_2$ set and let $R$ be a computable relation such that $i \in B \iff (\exists n)(\forall m)R(i, m, n)$. Let $A_n = \{(i, m) : \neg(\forall m)R(i, m, n)$ so that $B = \cup_n \mathbb{N} \setminus A_n$. Let $\psi(i, m) =$ (least $n)\neg R(i, m, n)$ and define $\phi$ by the s-m-n Theorem so that $\phi_{\phi(m)}(i) = \psi(i, m)$. Then $A_n = W_{\phi(m)}$ for each $n$, so that if $\phi_a(m) = \langle 7, \phi(m)\rangle$, then $H_a = B$.

A subset of $\mathbb{N}$ is said to be *hyperarithmetical* if it equals $H_a$ for some index $a \in H$. The hyperarithmetical hierarchy is defined as follows.

**Definition 1.14.3.** *For all* $\alpha$ *and all* $A \subseteq \mathbb{N}$*,*

*(i)* $A$ *is* $\Sigma^0_\alpha$ *if* $A = H_a$ *for some* $a \in H^\alpha$*;*

*(ii)* $A$ *is* $\Pi^0_\alpha$ *if* $\mathbb{N} \setminus A$ *is* $\Sigma^0_\alpha$*;*

*(iii)* $\Delta^0_\alpha = \Sigma^0_\alpha \cap \Pi^0_\alpha$.

It follows that for limit ordinals $\lambda$, $A$ is $\Sigma^0_\lambda$ if and only if $A$ is $\Sigma^0_\alpha$ for some $\alpha < \lambda$. It is easy to see that this definition agrees with the arithmetic hierarchy for $\alpha < \omega$. Note that for infinite ordinals $\alpha$, some authors (e.g. Hinman [80]) denote this hierarchy as the $\Sigma^0_{(\alpha)}$ sets and let the $\Sigma^0_\lambda$ sets be effective unions of $\Sigma^0_{(\lambda)}$ sets. The definition we use here is for uniformity of results concerning $\Pi^0_1$ classes.

There is a natural set $P_H = \{\langle a, m\rangle : a \in H \;\&\; m \in H_a\}$ which is complete for the family of hyperarithmetical sets. However, this set is $\Pi^1_1$ and not hyperarithmetical.

**Lemma 1.14.4.** *Every hyperarithmetical set is* $\Delta^1_1$.

*Proof.* We give a monotone $\Pi^0_2$ inductive definition of the following set:

$$V = \{\langle 0, a, m\rangle : a \in H \ \&\ m \in H_a\} \cup \{\langle 1, a, m\rangle : a \in H \ \&\ m \notin H_a\}.$$

That is, let

**(0)** $\langle 0, a, m\rangle \in \Gamma(X) \iff [(a = \langle 7, b\rangle \wedge m \in W_b) \vee (\exists n)(\phi_a(n)\downarrow \wedge \langle 1, \phi_a(n), m\rangle \in X)]$.

**(1)** $\langle 1, a, m\rangle \in \Gamma(X) \iff [(a = \langle 7, b\rangle \wedge m \notin W_b) \vee (\forall n)(\phi_a(n)\downarrow \wedge \langle 0, \phi_a(n), m\rangle \in X)]$.

We then show by induction on $\alpha$ that $\Gamma^\alpha = V^\alpha$. For $\alpha = 0$, both sets are empty and for $\alpha = 1$,

$$\Gamma^1 = V^1 = \{\langle\langle 7, b\rangle, 0, m\rangle : m \in W_b\} \cup \{\langle\langle 7, b\rangle, 1, m\rangle : m \notin W_b\}.$$

Now suppose that $\Gamma^\alpha = V^\alpha$ for all $\alpha < \beta$. If $\beta$ is a limit ordinal, then clearly

$$\Gamma^\beta = \cup_{\alpha<\beta}\Gamma^\alpha = \cup_{\alpha<\beta}V^\alpha = V^\beta.$$

Next suppose that $\beta = \alpha + 1$ for some $\alpha$.

If $\langle a, i, m\rangle \in \Gamma^{\alpha+1}$, then, for all $p$, $(\exists j \le 1)\langle \phi_a(p), j, \rangle \in \Gamma^\alpha$, so that by induction $\phi_a(p) \in H^\alpha$ for all $p$ and hence $a \in H^{\alpha+1}$. For $i = 0$, there exists $p$ such that $\langle \phi_a(p), 1, m\rangle \in \Gamma^\alpha$ and hence by induction $m \notin H_{\phi_a(p)}$. It follows that $m \in H_a$ and therefore $\langle a, 0, m\rangle \in V^{\alpha+1}$. For $i = 1$, it follows similarly that $m \notin H_a$.

Now suppose that $\langle a, i, m\rangle \in V^{\alpha+1}$. Then $a \in H^{\alpha+1}$, so that for all $p$, $\phi_a(p) \in H^\alpha$. For $i = 0$, $m \in H_a$ and therefore there exists $p$ such that $m \notin H_{\phi_a(p)}$ and hence by induction, $\langle \phi_a(p), 1, m\rangle \in \Gamma^\alpha$. It follows that $\langle a, 0, m\rangle \in \Gamma^{\alpha+1}$. The argument for $i = 1$ is similar.

Since $\Gamma$ is an arithmetical monotone inductive operator, it follows from Theorem 1.13.8 that $V$ is a $\Pi^1_1$ set and hence the hyperarithmetical set $H_a$ is $\Delta^1_1$ for each $a$. □

It follows from the proof of Lemma 1.14.4 that the set $P_H$ is $\Pi^1_1$. We can now prove the Spector-Gandy Theorem, due independently to Spector [185] and Gandy [71]. We say that a relation $R \subseteq \mathbb{N}^k \times \mathbb{N}^{\mathbb{N}^\ell}$ is ${\Sigma^1_1}^{HYP}$ if and only if there is an arithmetical relation $P$ such that

$$R(\overrightarrow{m}, \overrightarrow{x}) \iff (\exists y \in \Delta^1_1[\overrightarrow{x}])P(\overrightarrow{m}, \overrightarrow{x}, y).$$

**Theorem 1.14.5** (Spector-Gandy Theorem). *A relation* $R \subseteq \mathbb{N}^k \times \mathbb{N}^{\mathbb{N}^\ell}$ *is* ${\Sigma^1_1}^{HYP}$ *if and only if it is* $\Pi^1_1$.

*Proof.* We give the proof without the real parameters $\overrightarrow{x}$ and with just one number variable $m$. We may assume that the $\exists y$ quantifier ranges over $\{0,1\}^{\mathbb{N}}$. It follows from the proof of Lemma 1.14.4 that we can define a $\Sigma_1^1$ relation $S$ such that for any $a \in H$, $S(a,y) \iff y = H_a$. That is,

$$y = H_a \iff (\forall n)(\forall i < 2)[(a,i,n) \in V \rightarrow y(n) = i].$$

Now we have

$$(\exists y \in \Delta_1^1)(P(m,y)) \iff (\exists a)[a \in H \;\&\; (\forall y)(y = H_a \rightarrow P(m,y))].$$

This demonstrates that any ${\Sigma_1^1}^{HYP}$ relation $R$ is in fact $\Pi_1^1$.

For the reverse direction, it clearly suffices to show that the $\Pi_1^1$ complete relation $W$ is ${\Sigma_1^1}^{HYP}$.

It follows immediately from the first part of our proof that the set $\{0,1\}^{\mathbb{N}} \cap \Delta_1^1$ is itself $\Pi_1^1$. Then by Theorem 1.12.17, there is a computable function $F$ such that, for all $z$,

$$z \in \Delta_1^1 \iff F(z) \in W.$$

Now $\Delta_1^1 \cap \{0,1\}^{\mathbb{N}}$ cannot be a $\Delta_1^1$ set, by the following argument. Choose $a_0 \in W$ and let

$$Q(a,y) \iff a \in W \;\&\; [(y \in \Delta_1^1 \;\&\; F(y) = a] \;\vee\; (y \notin \Delta_1^1 \;\&\; a = a_0)].$$

Then $Q$ is $\Pi_1^1$ and therefore has a selector $Sel_Q$ with $\Pi_1^1$ graph by Theorem 1.12.19. Since $Sel_Q$ is total, the graph is actually $\Delta_1^1$. Now if $\Delta_1^1$ were itself $\Delta_1^1$, then the image of $\Delta_1^1$ would be a $\Sigma_1^1$ subset of $W$ and hence bounded by Theorem 1.12.20.

It follows from Theorem 1.12.21 that the range of $F$ is unbounded in $W$ and therefore

$$a \in W \iff (\exists z \in \Delta_1^1)\phi_a \preceq F(z).$$

□

Before establishing the reverse implication of lemma 1.14.4, we will need several technical lemmas from [80].

**Lemma 1.14.6.** *For all $\alpha$, $\Sigma_\alpha^0$ and $\Pi_\alpha^0$ are effectively closed under many-one reduction, that is, there is a primitive recursive function $g$ such that for all $m, e$:*

$$m \in H_{g(a,e)} \iff \phi_e(m) \in H_a.$$

*Proof.* Let $h$ be a primitive recursive function such that

$$g(a,e) = \langle 7, b\rangle, \text{ where } \phi_b(m) = \phi_{(a)_1}(\phi_e(m)), \text{ if } a = \langle 7, b\rangle;$$

and otherwise define $g$ by the Recursion Theorem so that

$$\phi_{g(a,e)}(p) = g(\phi_a(p), e).$$

□

**Lemma 1.14.7.** *For all $\alpha > 0$, the family of $\Sigma^0_\alpha$ relations is effectively closed under computably enumerable union and finite intersection; that is, there exists primitive recursive functions $f$ and $g$ such that*

*(i) If $\phi_a(p) \in H^\alpha$ for all $p$, then $f(a) \in H^\alpha$ and*

$$m \in H_{f(a)} \iff (\exists p) m \in H_{\phi_a(p)}.$$

*(ii) if $a, b \in H_\alpha$, then $g(a,b) \in H_\alpha$ and*

$$H_{g(a,b)} = H_a \cap H_b.$$

*Proof.* (i) We will define $\phi_{f(a)}(r)$ in two cases and then use the s-m-n theorem to define $f(a)$. In either case,

$$m \in H_{f(a)} \iff (\exists r) m \notin H_{f(a)}(r).$$

Let $(r)_0 = p$ and $(r)_1 = q$.

Case I: If $\phi_a(p) \notin H^1$, then

$$\phi_{f(a)}(r) = \phi_{\phi_a(p)}(q).$$

If there exists such an $r$ with $m \notin H_{f(a)}(r)$, then

$$(\exists p)[(\exists q) m \notin \phi_{\phi_a(p)}(q)],$$

so that as desired

$$(\exists p) m \in H_{\phi_a(p)}.$$

This argument is clearly reversible for $\phi_a(p) \notin H^1$.

Case II: If $\phi_a(p) = \langle 7, b\rangle$, then define $h(a,r)$ so that

$$W_{h(a,r)} = \mathbb{N} \setminus W_{b,q}$$

and let $\phi_{f(a)}(r) = \langle 7, h(a,r)\rangle$.

If $m \notin H_{\phi_{f(a)}(r)}$, then $m \notin W_{h(a,r)}$, so that $m \in W_b = H_{\phi_a(p)}$ and hence $(\exists p) m \in H_{\phi_a(p)}$ as desired. Again the reverse direction is clear.

(ii) There are four cases.

(1) If $a = \langle 7, d\rangle$ and $b = \langle 7, e\rangle$, then $g(a,b) = \langle 7, c\rangle$, where

$$\phi_c(n) = \phi_d(n) + \phi_e(n).$$

(ii) If $a = \langle 7, d\rangle$ and $b \notin H^1$, then $g(a,b) = c$ may be defined by part (i) so that

$$H_{\phi_c(p)} = H_{\phi_a(p)} \cup (\mathbb{N} \setminus W_d).$$

(iii) The case when $a \notin H^1$ and $b = \langle 7, e\rangle$ is similar to (ii).

(iv) If $a, b \notin H^1$, then $g(a,b)$ is defined by (i) so that

$$H_{\phi_{g(a,b)}(p)} = H_{\phi_a(p)} \cup H_{\phi_b(p)}.$$

□

**Lemma 1.14.8.** *For all $\alpha > 0$, the family of $\Sigma^0_\alpha$ relations is effectively closed under existential number quantification (and thus the family of $\Pi^0_\alpha$ relations is effectively closed under universal number quantification); that is, there is a primitive recursive function $h$ such that for all $a \in H_\alpha$, $h(a) \in H_\alpha$ and $m \in H_{h(a)} \iff (\exists p)\langle p, m\rangle \in H_a$.*

*Proof.* Let $h$ be a computable function such that $\phi_{h(p)}(m) = \langle p, m\rangle$. Then

$$(\exists p)\langle p, m\rangle \in H_a \iff (\exists p)\phi_{h(p)}(m) \in H_a.$$

Then taking $g$ from Lemma 1.14.6, we have

$$(\exists p)\langle p, m\rangle \in H_a \iff (\exists p) m \in H_{g(a,h(p))}.$$

Now letting $\phi_{\pi(a)}(p) = g(a, h(p))$ and taking $f$ from Lemma 1.14.7, we have

$$(\exists p)\langle p, m\rangle \in H_a \iff m \in H_{f(\pi(a))},$$

so we let $h(a) = f(\pi(a))$. □

**Theorem 1.14.9.** *Let $\Gamma$ be a $\Pi^0_k$ (resp. $\Sigma^0_k$) inductive operator, let $\lambda$ be a limit ordinal and let $n < \omega$. Then $\Gamma^n$ is $\Pi^0_{kn}$ (resp. $\Sigma^0_{kn}$), $\Gamma^\lambda$ is $\Sigma^0_{\lambda+1}$ and $\Gamma^{\lambda+n}$ is $\Pi^0_{\lambda+kn+1}$ (resp. $\Sigma^0_{\lambda+kn+1}$).*

*Proof.* We give the proof for a $\Pi^0_n$ operator and leave the other case to the reader. First we show that there is a primitive recursive function $g$ such that $\mathbb{N} \setminus \Gamma(H_e) = \mathbb{N} \setminus H_{g(e)}$ and furthermore if $e \in H^\alpha$, then $g(e) \in H^{\alpha+n}$. We give the proof for $k = 1$ and leave the general result as an exercise. Let $R$ be a computable relation such that

$$i \in \Gamma(A) \iff (\forall j) R(i, A\lceil j).$$

Then

$$i \notin \Gamma(H_e) \iff (\exists j)\neg R(i, (\mathbb{N} \setminus H_e)\lceil j).$$

Now

$$R(i, (\mathbb{N}\setminus H_e)\lceil j) \iff (\exists \sigma \in \{0,1\}^j)[R(i,\sigma) \;\&\; (\forall t < j)(\sigma(t) = 0 \iff t \in H_e)].$$

Since the last clause makes both positive and negative reference to $H_e$, it follows that we can define primitive recursive functions $f_p$ and $f_n$ such that

$$R(i, (\mathbb{N} \setminus H_e)\lceil j) \iff (i \in H_{f_p(e,j)} \;\wedge i \notin H_{f_n(e,j)}).$$

Then

$$i \notin \Gamma(H_e) \iff (\exists j)(i \notin H_{\phi_{f_p(e)}(j)}) \vee (\exists j)(i \in H_{\phi_{f_n(e)}(j)}).$$

It follows that

$$\mathbb{N} \setminus \Gamma(H_e) = H_{f_p(e)} \cup H_{f(f_n(e))},$$

where $f$ is the function from Lemma 1.14.7. Now the set $H_{g(e)} = H_{f_p(e)} \cup H_{f(f_n(e))}$ is $\Sigma^0_{\alpha+1}$ by Lemma 1.14.7 and thus $\Gamma(H_e)$ is $\Pi^0_{\alpha+1}$ as desired.

Now fix $c \in W$ with $\|c\| = \alpha$ and use the Recursion Theorem to define a primitive recursive function $h$ such that

$$\Gamma^{\|i\|_c} = \mathbb{N} \setminus H_{h(i)}.$$

(1) If $\|i\|_c = 0$, then $H_{h(i)} = \mathbb{N}$;

(2) If $\|j\|_c = \|i\|_c + 1$, then $H_{h(j)} = H_{g(h(i))}$;

(3) If $\|j\|_c$ is a limit, then $H_{h(j)} = \bigcap\{H_{h(i)} : i <_c j\}$.

It follows by induction that if $n$ is finite and $\lambda$ is a limit, then $\Gamma^n$ is $\Pi^0_n$, that $\Gamma^\lambda$ is $\Sigma^0_{\lambda+1}$ and that $\Gamma^{\lambda+n}$ is $\Pi^0_{\lambda+n+1}$. □

The next result gives a uniform inductive definition of the hyperarithmetic sets and can be used to define the transfinite jumps $\mathbf{0}^\alpha$.

**Theorem 1.14.10.** *There is a $\Pi^0_1$ inductive definition $\Gamma$ such that, for all $a$ and $m$, $m \notin H_a \iff \langle 4, a, m\rangle \in Cl(\Gamma)$ and furthermore, if $a \in H^\alpha$, then $m \notin H_a \iff \langle 4, a, m\rangle \in \Gamma^\alpha$.*

*Proof.* There are several clauses in the definition. We assume that $\phi_0$ is the empty function and omit the inclusive part of each clause (that $i$ must be in $\Gamma(A)$ if it is in $A$.)

**(1)** $\langle 1, a, m\rangle \in \Gamma(A) \iff m \notin W_a$.

**(2)** $\langle 2, a\rangle \in \Gamma(A) \iff \langle 1, 0, 0\rangle \in A \;\&\; (\forall m)\langle 1, a, m\rangle \notin A$.

The result of these two clauses is that $\phi_a$ is total if and only if $\langle 2, a\rangle \in Cl(\Gamma)$, which is if and only if $\langle 2, a\rangle \in \Gamma^2$.

**(3a)** $\langle 3, \langle 7, a\rangle\rangle \in \Gamma(A)$

**(3b)** $\langle 3, b \in \Gamma(A) \iff \langle 2, b\rangle \in \Gamma(A) \;\&\; (\forall n)\langle 3, \phi_a(n)\rangle \in A$.

These two clauses ensure that, for all $a$ and for all ordinals $\alpha$,

$\langle 3, a\rangle \in \Gamma^\alpha \iff a \in H^\alpha$.

**(4a)** $\langle 4, \langle 7, a\rangle, m\rangle \in \Gamma(A) \iff m \notin W_a$.

**(4b)** $\langle 4, b, m\rangle \in \Gamma(A) \iff \langle 3, b\rangle \in \Gamma(A) \;\&\; (\forall n)\langle 4, \phi_b(n), m\rangle \notin A$.

These final two clauses complete the definition. The theorem follows by induction on $\alpha$ as follows.

For $\alpha = 1$, if $b = \langle 7, a\rangle \in H^1$, then $\langle 3, b\rangle \in \Gamma^1$ by clause (3a) and, by clause (4a):

$$m \notin H_b \iff m \notin W_a \iff \langle 4, b, m\rangle \in \Gamma^1.$$

For $\alpha \geq 1$ and $b \in H^{\alpha+1} - H^\alpha$, $\phi_b$ must be total, so we have $\langle 2, b\rangle \in \Gamma^{\alpha+1}$ and then

$$\langle m \notin H_b \iff (\forall n) m \in H_{\phi_b(n)} \iff (\forall n)\langle 4, \phi_b(n), m\rangle \in \Gamma^\alpha \iff \langle 4, b, m\rangle \in \Gamma^{\alpha+1}.$$

□

This theorem has two important corollaries. Note that we have already defined $\mathbf{0}^{(n)}$ for finite $n$.

**Definition 1.14.11.** *For any computable ordinal $\alpha \geq \omega$, let*

$$\mathbf{0}^{(\alpha)} = \{\langle a, m\rangle : a \in H^\alpha \ \& \ m \in H_a\}.$$

**Theorem 1.14.12.** *For each computable ordinal $\alpha \geq \omega$,*

1. *$\mathbf{0}^{(\alpha+1)}$ is $\Sigma^0_{\alpha+1}$ complete, and*

2. *any set $A$ is $\Sigma^0_{\alpha+1}$ if and onlyl if it is $\Sigma^0_1$ in $\mathbf{0}^{(\alpha)}$.*

*Proof.* (1) It follows from Theorems 1.14.9 and 1.14.10 that $\mathbf{0}^{(\alpha+1)}$ is $\Sigma^0_{\alpha+1}$. The completeness is immediate from Theorem 1.14.10.

(2) is left as an exercise. □

The $\Sigma^0_\alpha$ and $\Pi^0_\alpha$ sets may be characterized in terms of inductive definability. The next result follows directly from Theorems 1.14.9 and 1.14.10.

**Theorem 1.14.13.** *For any $A \subseteq \mathbb{N}$ and any computable ordinal $\alpha$, $A$ is $\Pi^0_\alpha$ if and only if $A$ is m-reducible to $\Gamma^\alpha$ for some $\Pi^0_1$ inductive definition $\Gamma$.* □

Finally, we can characterize the $\Sigma^0_{\alpha+1}$ sets as relative c. e. over the jumps.

**Theorem 1.14.14.** *For any computable ordinal $\alpha$ and any $A \subseteq \mathbb{N}$, $A$ is $\Sigma^0_{\alpha+1}$ if and only if $A$ is c. e. in $\mathbf{0}^{(\alpha)}$ and $A$ is $\Delta^0_{\alpha+1}$ if and only if $A$ is computable in $\mathbf{0}^{(\alpha)}$.*

*Proof.* Let $B = \mathbf{0}^{(\alpha)} = H_b$ for some $b \in H^\alpha$. Suppose first that $A$ is $\Sigma^0_{\alpha+1}$. Then $A = H_a$ for some $a \in H^{\alpha+1}$. Thus we have

$$m \in A \iff (\exists n) m \in H_{\phi_a(n)} \iff (\exists n)\langle \phi_a(n), m\rangle \in B.$$

For the other direction, suppose that $A$ is c. e. in $B$. Then for some $e$, we have

$$m \in A \iff \phi_e(m, B)\downarrow \iff (\exists t)\phi_e(m, B\lceil t)\downarrow \iff (\exists \sigma)[\sigma \prec B \ \& \ \phi_e(m,\sigma)\downarrow].$$

By Lemma 1.14.7, it suffice to show that $\sigma \prec B$ is $\Sigma^0_{\alpha+1}$. But we have

$$\sigma \prec B \iff (\forall i < |\sigma|)(\sigma(i) = 0 \to i \in B) \ \& \ (\forall i < |\sigma|)(\sigma(i) = 1 \to i \notin B).$$

Now "$i \in B$" is $\Sigma^0_\alpha$ and therefore $\Sigma^0_{\alpha+1}$ and $i \notin B$ is clearly $\Sigma^0_{\alpha+1}$, so the result follows, again by Lemma 1.14.7. □

Monotone inductive definitions are frequently used and there is a finer result for the complexity of the levels.

**Theorem 1.14.15.** *For any ordinal $\alpha$, any $n \in \mathbb{N}$ and any monotone $\Pi^0_1$ inductive operator $\Gamma$, $\Gamma^{\omega\cdot\alpha}$ is $\Sigma^0_{2\alpha}$ and $\Gamma^{\omega\cdot\alpha+n+1}$ is $\Pi^0_{2\alpha+1}$.*

*Proof.* The proof is similar to that of Theorem 1.14.9, with the additional idea that if $A$ is $\Pi^0_\beta$, then $\Gamma(A)$ is also $\Pi^0_\beta$, since by monotonicity,

$$i \in \Gamma(A) \iff (\forall j)R(i, A\lceil j) \iff (\forall j)(\forall C \subseteq n)[A \subseteq C \to R(i, C)].$$

It follows that $\Gamma^n$ is $\Pi^0_1$ for all $n$ and that if $\Gamma^\lambda$ is $\Sigma^0_\beta$, then $\Gamma^{\lambda+n}$ is $\Pi^0_{\beta+1}$. Making use of Lemmas 1.14.6, 1.14.7, 1.14.8 and the Recursion Theorem, the proof follows as above. Details are left to the reader. □

**Theorem 1.14.16.** *(Souslin-Kleene) A subset of* $\mathbb{N}^k \times (\mathbb{N}^\mathbb{N})^l$ *is* $\Delta^1_1$ *if and only if it is hyperarithmetical.*

*Proof.* The direction $(\longleftarrow)$ is a routine generalization of Lemma 1.14.4. For the other direction, it suffices by Lemma 1.14.6 and Theorem 1.12.21 to show that $W_\alpha$ is hyperarithmetical for any computable ordinal $\alpha$. Recall the $\Pi^0_1$ monotone inductive definition $\Gamma$ of $W$ given in Example 1.13.3 such that such that $W_\alpha = \Gamma^\alpha$, It now follows from Theorem 1.14.9 each $W_\alpha$ is hyperarithmetic. □

The next application of inductive definability is part of a theorem of Chen [44].

**Theorem 1.14.17.** *Let* $\alpha > 1$ *be a computable ordinal and let* $n \geq 1$ *be a natural number. Then* $PW_{\omega\cdot\alpha}$ *is* $\Sigma^0_{2\alpha}$ *complete and* $PW_{\omega\cdot\alpha+n}$ *is* $\Pi^0_{2\alpha+1}$ *complete.*

*Proof.* We will just demonstrate the upper bound on the complexity. A partial computable function $\phi_c$ represents the pre-ordering $R_c$ if it is the characteristic function, that is

$$R_c(i, j) \iff \phi_c(\langle i, j\rangle) = 1$$

and

$$\neg R_c(i, j) \iff \phi_c(\langle i, j\rangle) = 0.$$

The restriction of $R_c$ to elements below $n$ may be given by

$$\phi_{h(c,n)}(i, j) = \phi_c(i, j) \cdot (1 - \phi_c(n, j)).$$

Note that if $\phi_c$ is total, then $\phi_{h(c,n)}$ is total for all $n$.

There is a natural $\Pi^0_1$ monotone inductive definition of $PW$ given by

$$c \in \Gamma(A) \iff (\forall n)h(c, n) \in A.$$

Then it is easy to see that for a total function $\phi_c$ which represents a pre-linear-ordering,

$$c \in PW_\beta \iff c \in Tot \;\&\; c \in \Gamma^\beta.$$

The condition that $\phi_c$ is total is $\Pi^0_2$ and the condition that $\phi_c$ is the characteristic function of a pre-linear-ordering is $\Pi^0_1$. Thus the upper bound on the complexity follows from Theorem 1.14.15.

The proof of the other direction is omitted. □

**Theorem 1.14.18.** *Let $T$ be a computable tree and define a prewellordering $R$ on $T$ so that $\|\sigma\|_R = ht_T(\sigma)$ (where $ht(\sigma) = \infty$ if $\sigma \notin Ext(T)$.) Then for any ordinal $\alpha$, and any $n \in \mathbb{N}$, $\{\sigma \in T : ht_T(\sigma) < \omega \cdot \alpha\}$ is $\Sigma^0_{2\alpha}$ and $\Gamma^{\omega\cdot\alpha+n+1}$ is $\Pi^0_{2\alpha+1}$.*

*Proof.* The proof is left as an exercise. □

Computable trees with a unique infinite branch were studied by Clote [48]. It is well-known that for every hyperarithmetic set $A$, there exists a computable tree with a unique infinite branch $x$ such that $A$ is Turing reducible to $x$. We will prove this below in Chapter 4.

**Theorem 1.14.19.** *([48]) Let $T$ be a computable tree $T$ with a unique infinite branch $x$. Then for any $\sigma \in T - Ext(T)$, $ht_T(\sigma) < \omega_1$. If $ht_T(\sigma) < \omega \cdot \alpha$ for all $\sigma \notin Ext(T)$, then $x$ is Turing reducible to a $\Sigma^0_{2\alpha}$ set.*

*Proof.* The first part follows from Lemma 1.12.12. For the second part, note that the set $A = \{\sigma \in T : ht_T(\sigma) < \omega \cdot \alpha\}$ is $\Sigma^0_{2\alpha}$ by Lemma 1.14.18. Then $x$ may be computed recursively from $A$ by

$$x(n+1) = (\text{least } i)[(x(0), \ldots, x(n), i) \notin A].$$

□

**Exercises**

1.14.1. Show that for finite $n$, the definition of the hyperarithmetical sets agrees with the previous definition of the arithmetical sets.

1.14.2. Give an alternate proof of Lemma 1.14.4 using the Prewellordering Theorem 1.13.11.

1.14.3. Give the following improvement of Theorem 1.14.9. Suppose that $\Gamma$ is a $\Pi^0_k$ (respectively, $\Sigma^0_k$) inductive operator and that $\Gamma^1$ is $\Delta^0_k$ for some $m < k$. Show that for each $n$, $\Gamma^{n+1}$ is $\Pi^0_{kn+m}$ (resp. $\Sigma^0_{kn+m}$).

1.14.4. For computable ordinals $\alpha < \beta$, $\Sigma^0_\alpha \cup \Pi^0_\alpha \subset \Delta^0_\beta$.

1.14.5. Show that a set $A$ is $\Sigma^0_{\alpha+1}$ if and onlyl if it is $\Sigma^0_1$ in $\mathbf{0}^{(\alpha)}$.

1.14.6. Give the details in the proof of Theorem 1.14.15.

1.14.7. Let $T$ be a computable tree and define a prewellordering $R$ on $T$ so that $\|\sigma\|_R = ht_T(\sigma)$ (where $ht(\sigma) = \infty$ if $\sigma \notin Ext(T)$.) Show that for any ordinal $\alpha$, and any $n \in \mathbb{N}$, $\{\sigma \in T : ht_T(\sigma) < \omega \cdot \alpha\}$ is $\Sigma^0_{2\alpha}$ and $\Gamma^{\omega\cdot\alpha+n+1}$ is $\Pi^0_{2\alpha+1}$.

1.14.8. Show that $P_H$ is $\Pi^1_1$.

1.14.9. Show that there can be no *universal* hyperarithmetical set ( so in particular $P_H$ is not hyperarithmetical.)

1.14.10. For each computable ordinal $\alpha$, both the $\Sigma^0_\alpha$ and the $\Pi^0_\alpha$ relations are closed under $\leq_m$ reducibility.

1.14.11. The definition of the hyperarithmetical hierarchy is easily extended to subsets of $\mathbb{N}^\mathbb{N}$ and in general to relations $R \subseteq \mathbb{N}^k \times (\mathbb{N}^\mathbb{N})^\ell$. Give the details.

# Chapter 2

# Fundamentals of $\Pi^0_1$ Classes

This chapter contains the formal definition of a $\Pi^0_1$ class as well as the set of infinite paths through a computable tree, well as some equivalent formulations. In particular, the notation "$\Pi^0_1$" indicates that a $\Pi^0_1$ class may be represented in arithmetic by a formula having one universal quantifier, ranging over natural numbers. We will explain the connection between these notions. Some notions of boundedness for trees are examined, including highly computable and finite-branching trees. This leads to computably bounded and bounded $\Pi^0_1$ classes. Decidable $\Pi^0_1$ classes are defined, corresponding to computable trees with no dead ends. Products and disjoint unions of trees and classes are studied. The notion of compactness is examined together with König's Lemma. The family of $\Pi^0_1$ classes is shown to have the dual reduction and separation properties. Strong $\Pi^0_n$ classes are defined. Computable and continuous functions on $\mathbb{N}^{\mathbb{N}}$ and $\{0,1\}^{\mathbb{N}}$ are studied in connection with $\Pi^0_1$ classes. Classes of separating sets for a pair of disjoint c.e. sets are studied and in particular diagonally non-computable sets. The connection between retraceable and hyperimmune sets and the $\Pi^0_1$ class of initial subsets of a co-c.e. set is given. Several notions of reducibility between various classes are examined. For example, every c. b. class is computably homeomorphic to a subclass of $\{0,1\}^{\mathbb{N}}$, every bounded $\Pi^0_1$ class is computably homeomorphic to a strong $\Pi^0_2$ class of sets and every $\Pi^0_2$ class can be put in one-to-one degree-preserving correspondence with a $\Pi^0_1$ class.

We also introduce the applications of $\Pi^0_1$ classes by considering the representation of logical theories.

## 2.1 Computable trees and notions of boundedness

Recall that a tree $T$ is a subset of $\mathbb{N}^*$ which is closed under initial segments. Such a tree is said to be *$\omega$-branching*, since each node has potentially a countably infinite number of immediate successors. We identify an element $\sigma$ of $\mathbb{N}^*$ with its code $\langle\sigma\rangle \in \mathbb{N}$ and say that $T$ is computable if the set of codes $\langle\sigma\rangle$ such that $\sigma \in T$ is a computable set.

**Definition 2.1.1.** *(i) For a given function $g : \mathbb{N}^* \to \mathbb{N}$, a tree $T \subseteq \mathbb{N}^*$ is said to be g-bounded* if for every $\sigma \in T$ and every $i \in \omega$, if $\sigma^\frown i \in T$, then $i < g(\sigma)$.

(ii) $T$ is *finite branching* if each node $\sigma$ of a tree $T$ has finitely many immediate successors $\sigma^\frown i$.

There are other equivalent formulations.

**Lemma 2.1.2.** *For any tree $T \subseteq \mathbb{N}^*$, the following are equivalent:*

1. *$T$ is finite branching;*
2. *$T$ is $g$-bounded for some $g$.*
3. *There is a function $f : \mathbb{N}^* \to \mathbb{N}^*$ such that, for each $\sigma \in T$, $f(\sigma) = (i_1, \dots, i_k)$, where $i_1 < \cdots < i_k$ enumerates $\{i : \sigma^\frown i \in T\}$;*
4. *There is a function $f' : \mathbb{N}^* \to \mathbb{N}$ such that, for each $\sigma \in T$, $\sigma$ has at most $f'(\sigma)$ immediate successors;*
5. *There is a function $h$ such that $\sigma(i) < h(i)$ for all $\sigma \in T$ and all $i < |\sigma|$.* □

The proof of this lemma is left as an exercise.

**Definition 2.1.3.** *(i) A tree $T$ is* computably bounded (c. b.) *if it is $g$-bounded for some computable function $g$.*

*(ii) A computable tree $T$ is said to be* highly computable *if it is also computably bounded.*

*(iii) $T$ is* highly computable in $z$ *if it is computable in $z$ and also $g$-bounded by some function $g$ computable in $z$.*

**Lemma 2.1.4.** *For any computable tree $T \subseteq \mathbb{N}^*$, the following are equivalent:*

*(a) $T$ is highly computable;*

*(b) There is a computable function $h$ such that $\sigma(i) < h(i)$ for all $\sigma \in T$ and all $i < |\sigma|$.*

*(c) There is a computable function $f : \mathbb{N}^* \to \mathbb{N}^*$ such that, for each $\sigma \in T$, $f(\sigma) = (i_1, \dots, i_k)$, where $i_1 < \cdots < i_k$ enumerates $\{i : \sigma^\frown i \in T\}$.*

*Proof.* (a) → (b): Given the function $g$, let $h(0) = g(\emptyset)$ and recursively compute $h$ by $h(n+1) = max\{g(\sigma) : \sigma \in \{0, 1, \dots, h(n)\}^n \cap T\}$.

(b) → (c): Given the function $h$, the sequence $i_1 < \cdots < i_k$ can be computed from $\sigma$ by testing in turn whether $\sigma^\frown i \in T$ for $i < h(|\sigma| + 1)$.

(c) → (a): For a given $\sigma$, use the function $f$ to compute $i_1 < \cdots < i_k$ as indicated and then $g(\sigma) = i_k + 1$ will be an upper bound for $\{i : \sigma^\frown i \in T\}$. □

Note that the argument that (c) implies (a) does not use the computability of $T$ but the other two arguments do. We leave it as an exercise to define (noncomputable) trees which possess computable bounding functions of type (a) but do not possess bounding functions of type (b) and similarly for (b) and (c).

Observe that we have omitted here the condition from Lemma 2.1.2 that there is a computable upper bound on the number of successors of $\sigma$. It is an exercise to show that any highly computable tree possesses such a function, but there exists a computable tree which is not highly computable but possesses such a bounding function.

**Example 2.1.5.** *Here is an example of a computable tree which is bounded but not computably bounded. Put $0^e{}^\frown s+1 \in T$ if and only if $\phi_e(e)$ converges at stage $s$. Then each node $\sigma \in T$ has at most two extensions, so that $T$ is finite branching. Now suppose by way of contradiction that $T$ were computably bounded by the function $g$. Then $\phi_e(e)\downarrow \iff \phi_{e,g(0^e)}(e)\downarrow$, which would make $\{e : \phi_e(e)\downarrow\}$ a computable set.*

There is a further notion of boundedness.

**Definition 2.1.6.**

1. *$T$ is* almost bounded *by $g$ if there is some $k \in \mathbb{N}$ such that for all $\sigma$ with $|\sigma| > k$ and for all $i$, if $\sigma{}^\frown i \in T$, then $i < g(\sigma)$.*
2. *$T$ is* almost bounded (a. b.) *if it is almost bounded by some $g$.*
3. *$T$ is* almost computably bounded (a. c. b.) *if it is almost bounded by some computable function $g$.*

Note that these notions are not equivalent to the existence of a (computable) function $h$ and a finite $k$ such that $\sigma(i) < h(i)$ for all $\sigma \in T$ with $|\sigma| > k$ and all $i < |\sigma|$.

**Example 2.1.7.** *Let $T = \{n^k : n, k \in \omega\}$. Then $T$ is a. c. b. but has, for each $k$ and $n$, strings $\sigma = n^{k+1} \in T$ with $\sigma(k) = n$. It is clear that $T$ is not finite branching.*

Two trees may be joined together in various ways.

**Definition 2.1.8.** *For two trees $S, T \subset \mathbb{N}^*$*

1. *$S \oplus T = \{0{}^\frown\sigma : \sigma \in S\} \cup \{1{}^\frown\tau : \tau \in T\}$.*
2. *For strings $\sigma$ and $\tau$ with $|\sigma| = |\tau| = n$, $\sigma \oplus \tau = (\sigma(0), \tau(0), \ldots, \sigma(n-1), \tau(n-1))$; if $|\sigma| = n+1$ and $|\tau| = n$, then $\sigma \oplus \tau = (\sigma(0), \tau(0), \ldots, \sigma(n-1), \tau(n-1), \sigma(n))$.*
3. *$S \otimes T = \{\sigma \oplus \tau : \sigma \in S \;\&\; \tau \in T\}$.*

Clearly $S \oplus T$ is bounded if and only if both $S$ and $T$ are bounded and similarly for the other notions of boundedness.

**Exercises**

2.1.1. Prove Lemma 2.1.2.

2.1.2. Find a (non-computable) tree $T$ with a computable function $h$ such that $\sigma(i) < h(i)$ for all $\sigma \in T$ and all $i < |\sigma|$, such that there is no computable function $f$ which enumerates the immediate successors of $\sigma \in T$.

2.1.3. Given a highly computable tree $T$, find a computable function $f$ such that, for each $\sigma \in T$, $\sigma$ has $\leq f(\sigma)$ immediate successors in $T$.

2.1.4. Show that $S \oplus T$ and $S \otimes T$ are trees.

## 2.2 Definition and basic properties of $\Pi^0_1$ Classes

In this section, we examine $\Pi^0_1$ classes with various boundedness conditions. We begin with some general facts.

**Lemma 2.2.1.** *A subset $K$ of $\mathbb{N}^{\mathbb{N}}$ is closed if and only if $K = [T]$ for some tree $T$.*

*Proof.* Suppose first that $K$ is closed and let $T = \{\sigma \in \mathbb{N}^* : K \cap I[\sigma] \neq \emptyset\}$. We will verify that $K = [T]$. If $x \in K$, then for any $n$, $x \in K \cap I[x \upharpoonright n]$, so that $x \upharpoonright n \in T$. It follows that $x \in [T]$. Conversely, suppose that $x \notin K$. Since $K$ is closed, there must be some basic interval $I[x \upharpoonright n]$ such that $K \cap I[x \upharpoonright n] = \emptyset$. But then $x \upharpoonright n \notin T$ and hence $x \notin [T]$. $\square$

**Definition 2.2.2.** *1. For any closed set $P \subseteq \mathbb{N}^{\mathbb{N}}$, let $T_P$ denote the tree $\{\sigma \in \mathbb{N}^* : P \cap I[\sigma] \neq \emptyset\}$.*

*2. For any tree $T$, an* infinite path *through $T$ is a sequence $(x(0), x(1), \dots)$ such that $x \upharpoonright n \in T$ for all $n$. Let $[T]$ be the set of infinite paths through $T$.*

*3. A subset $P$ of $\mathbb{N}^{\mathbb{N}}$ is a $\Pi^0_1$* class *($P$ is* effectively closed *if $P = [T]$ for some computable tree $T$.*

*4. A subset $P$ of $\mathbb{N}^{\mathbb{N}}$ is a* decidable *$\Pi^0_1$ class if $T_P$ is a computable set.*

It is important to note that for a $\Pi^0_1$ class $P$, the set $T_P$ need not be computable.

**Example 2.2.3.** *Let $A$ be an arbitrary c.e. set and let $P = \{0^n 1^\omega : n \notin A\}$. Then $P$ is a $\Pi^0_1$ class but $T_P$ is not computable. (Details left as an exercise.)*

The notions of boundedness for trees from the previous section carry over to notions of boundedness for closed sets.

**Definition 2.2.4.** *A subset $K$ of $\mathbb{N}^{\mathbb{N}}$ is* (topologically) bounded *if there is a function $h$ such that, for all $x \in K$ and all $n$, $x(n) \leq h(n)$.*

In the study of $\Pi^0_1$ classes, the term "bounded" has the effective version given below, so that we use the modifier "topologically" to distinguish the two notions.

**Definition 2.2.5.** *1. A $\Pi^0_1$ class $P$ is* bounded *if there is a computable tree $T$ such that $P = [T]$ and a function $h$ (not necessarily computable) such that $\sigma(n) \leq h(n)$ for all $\sigma \in T$.*

*2. A $\Pi^0_1$ class $P$ is* computably bounded (c. b.) *if $P = [T]$ for some highly computable tree $T$.*

*3. A $\Pi^0_1$ class $P$ is* almost bounded *if $P = [T]$ for some a.b. tree $T$.*

*4. A $\Pi^0_1$ class $P$ is* almost computably bounded (a. c. b.) *if $P = [T]$ for some a. c. b. tree $T$.*

In the next chapter, we will show that for any hyperarithmetical real $x \in \mathbb{N}^{\mathbb{N}}$, there exists $y \equiv_T x$ such that $\{y\}$ is a $\Pi^0_1$ class. Here we give the special case when $x$ is a $\Sigma^0_2$ set.

**Example 2.2.6.** *Here is an example of a $\Pi^0_1$ class $P$ which is topologically bounded, but not bounded. That is, $P = \{g\}$ for a particular function $g$ and hence $P$ is bounded by $g$ itself. At the same time, $P = [T]$ for some computable tree $T$. However, the tree $T$ is not itself bounded by $g$, or even finite branching.*

*Let $A$ be a $\Sigma^0_2$ set which is not $\Pi^0_2$. Let $R$ be a computable relation so that*

$$e \in A \iff (\exists m)(\forall n)R(e, m, n).$$

*Define the (non-computable) function $f$ in two cases as follows.*

*(Case I): If $e \in A$, then $f(e, 0) = 1$, $f(e, 1)$ is the least $m$ such that $(\forall n)R(e, m, n)$ and $f(e, 2) = \langle n_0, \ldots, n_{f(e,1)-1}\rangle$ where for each $i$, $n_i$ is the least $n$ such that $\neg R(e, i, n)$.*

*(Case II): If $e \notin A$, then $f(e, 0) = 0$ and $f(e, m + 1)$ is the least $n$ such that $\neg R(e, m, n)$.*

*Then define $g$ by $g(2^e(2m + 1)) = f(e, m)$. Observe that for each $e$, the values of $g(2^e(2m + 1)$ tell us whether $e \in A$ and also verify the answer.*

*We claim that $\{g\}$ is a $\Pi^0_1$ class. That is, $\{g\} = [T]$ for the computable tree $T$ defined as follows.*

*$\sigma \in T$ if and only if, for all $e$ with $2^e < |\sigma|$, one of the following.*

*(0) $\sigma(2^e) = 0$ and, for all $i$ with $2^e(2i + 1) < |\sigma|$, $\neg R(e, i, \sigma(2^e(2i + 1))$, and for all $j < \sigma(2^e(2i + 1))$, $R(e, i, j)$.*

*(1) $\sigma(2^e) = 1$ and, for all $n < |\sigma|$, $R(e, \sigma(3 \cdot 2^e), n)$. For all $i > 2$, $\sigma(2^e(2i + 1)) = 0$. Finally, $\sigma(5 \cdot 2^e = \langle n_0, n_1, \ldots, n_{\sigma(3\cdot 2^e)-1}\rangle$ where for all $i < \sigma(3 \cdot 2^e)$, $n_i$ is the least $n$ such that $\neg R(e, i, n)$.*

*Clearly $g \in [T]$. Now suppose there were some other function $h \in [T]$. Fix $e$ and consider two cases as above.*

*(Case I): $h(2^e) = 0$. Then for each $i$, $\neg R(e, i, h(2^e(2i+1))$ and hence $e \notin A$. It follows that, for each $m$, $h(2^e(2m+1))$ is the least $n$ such that $\neg R(e, m, n)$ and hence $h(2^e(2m+1)) = f(2^e(2m+1)$ for all $m$.*

*(Case II): $h(2^e) = 1$. Then for all $n$, $R(e, h(3 \cdot 2^e), n)$ and hence $e \in A$. Furthermore, for each $i < h(3 \cdot 2^e)$, $h(5 \cdot 2^e)$ witnesses that $\neg(\forall n)R(e, i, n)$, so that $h(3 \cdot 2^e)$ is the least $m$ such that $(\forall n)R(e, m, n)$; that is, $h(3 \cdot 2^e) = f(3 \cdot 2^e)$. It follows that $h(2^e(2i+1) = h(3 \cdot 2^e) = f(3 \cdot 2^e) = f(2^e(2i+1)$ for all $i > 2$. Finally, $h(5 \cdot 2^3)$ must code the sequence of least witnesses $n$ such that $\neg R(e, i, n)$ for each $i < h(3 \cdot 2^e)$ and hence $h(5 \cdot 2^e) = f(5 \cdot 2^e)$ as well.*

*It follows that $h = f$ and hence $[T] = \{f\}$ as desired.*

*Now we claim also that this class does not have a $\Delta^0_2$ bounding function $h$. Suppose by way of contradiction that there were such a function $h$. Then*

$$e \notin A \iff (\forall m < h(2^e 3))(\exists n)\neg R(e, m, n).$$

*This would give a $\Pi^0_2$ definition of $A$. To see this, let*

$$\phi(e) = (least\ n)(\forall m < h(2^e 3))(\exists n' < n)\neg R(e, m, n').$$

*Then $\phi$ is computable in $\mathbf{0}'$ and $\mathbb{N} \setminus A = Dom(\phi)$, so that $A$ is a $\Pi^0_2$ set.*

*Now the $\Pi^0_1$ class $\{g\}$ is certainly topologically bounded, but it follows from Lemma 2.2.7 that it is not bounded.*

**Lemma 2.2.7.** *(a) A closed subset $K$ of $\mathbb{N}^{\mathbb{N}}$ is (topologically) bounded if and only if there is a finite-branching tree $T$ such that $K = [T]$;*

*(b) A $\Pi^0_1$ class $P$ is bounded if and only if there is a finite-branching computable tree $T$ such that $P = [T]$. Furthermore, the bounding function may always be taken to be computable in $\mathbf{0}'$.*

*(c) A decidable $\Pi^0_1$ class $P$ is effectively bounded if and only if it is bounded.*

*Proof.* We leave the proof of part (a) to the reader. Let $T$ be a computable tree such that $K = [T]$. Suppose first that $K$ is bounded and let $h$ be a function such that $\sigma(i) \le h(i)$ for all $\sigma \in T$ and all $i < |\sigma|$. It is immediate that $T$ is finite-branching, since for any $\sigma \in T$, if $\sigma^\frown j \in T$, then $j \le h(|\sigma|)$. On the other hand, if $T$ is finite-branching, then it is easy to see by induction that $T \cap \mathbb{N}^k$ is finite for all $k$ and hence we may define the bounding function $h$ by

$$h(k) = max\{i : (\exists \sigma \in T \cap \mathbb{N}^k)\sigma^\frown i \in T\}.$$

It is clear that $h$ is computable in $\mathbf{0}'$.

(c) Let $P = [T]$ where $T$ is computable and has no dead ends. If $T$ is finite-branching, then $P$ is bounded by (a). Now suppose that $P$ is bounded by a function $h$. Since $T$ has no dead ends, it follows that $T$ is also bounded by $h$. Thus $T$ must be finite-branching. $\square$

**Proposition 2.2.8.** *A $\Pi^0_1$ class $P$ is computably bounded if and only if there is a computable function $h$ such that $x(n) \leq h(n)$ for all $x \in P$.*

*Proof.* Let $P = [T]$ for the computable tree $T$. If $P$ is computably bounded, then there is a computable function $h$ such that $\sigma(n) \leq h(n)$ for all $\sigma \in T$ and all $n < |\sigma|$ and it follows that $x(n) \leq h(n)$ for all $x \in P$. For the other direction, let $h$ be given as described. Then we can define a tree $S \subseteq T$ by having

$$\sigma \in S \iff \sigma \in T \;\&\; (\forall n < |\sigma|)\sigma(n) \leq h(n).$$

It is clear that $[S] = [T] = P$ so that $P$ is computably bounded. □

**Definition 2.2.9.** *(i) $\sigma$ is an* extendible *node of $T$ if $I(\sigma) \cap [T] \neq \emptyset$, that is, if $\sigma$ has an infinite extension which belongs to $[T]$; $Ext(T)$ is the set of extendible nodes of $T$.*

*(ii) A node $\sigma$ such that $\sigma \notin T$, but all $\tau \prec \sigma$ are in $T$, is a* dead end *of $T$.*

Observe that $T_P$ has no dead ends and is the unique tree without dead ends such that $P = [T]$. The following lemma gives an alternate definition of the notion of a decidable $\Pi^0_1$ class. The proof is left as an exercise.

**Lemma 2.2.10.** *For any $\Pi^0_1$ class $P$, the following are equivalent:*

*(a) $P$ is decidable;*

*(b) There is a computable tree $T$ with $P = [T]$ such that $Ext(T)$ is computable.*

*(c) There is a computable tree $T$ with no dead ends such that $P = [T]$.* □

It is a fundamental property of the real line that a subset is compact if and only if it is closed and bounded.

**Theorem 2.2.11.** *(a) A subset $K$ of $\mathbb{N}^{\mathbb{N}}$ is compact if and only if it is closed and (topologically) bounded.*

*(b) A $\Pi^0_1$ class $P$ is bounded if and only if there exists a computable tree $T$ with $P = [T]$ and a function $h$ such that $\sigma(n) \leq h(n)$ for all $\sigma \in T$ and all $n < |\sigma|$. Furthermore, $h$ may be taken to be computable in $\mathbf{0}'$.*

*Proof.* (a) Suppose first that $K$ is compact. Then $K$ is certainly closed. For each $n$, $K \subset \bigcup_i \{x : x(n) = i\}$ and it follows from compactness that there exists some $i_n$ such that $K \subseteq \{x : x(n) \leq i_n\}$. Then the function $h(n) = i_n$ satisfies the condition above. Suppose next that $K$ is closed and bounded and let $h$ be a bounding function for $K$. Then $K \subseteq \prod_{n\in\omega}\{0, 1, \ldots, h(n)\}$ and is therefore compact since it is a closed subset of a compact space.

(b) If $P = \emptyset$, then this is obvious. Thus we let $P$ be a nonempty $\Pi^0_1$ class and let $T$ be a computable tree such that $P = [T]$. Suppose first that $T$ is finitely branching. Then we may define the bounding function $h$ by letting $h(n)$ be the maximum of $\{\sigma(m) : \sigma \in T,\ m\||\sigma| \leq n\}$. It is clear that $h$ is computable in $\mathbf{0}'$. Conversely, suppose that $P = [T]$ and that $h$ is any bounding function such that $\sigma(n) \leq h(n)$ for all $\sigma \in T$. It is immediate that $T$ must be finitely branching. □

A crucial result for bounded classes is König's Lemma, which follows from the compactness of bounded classes.

**Lemma 2.2.12** (König's Lemma)**.** *Any infinite, finite-branching tree has an infinite path.*

*Proof.* Let $T$ be an infinite, finite-branching tree and let $P = [T]$. An infinite path $x$ through $T$ is defined as follows. Let $x(0)$ be the least $i$ such that $T$ contains infinitely many extensions of $(i)$ and for each $n$, similarly let $x(n+1)$ be the least $i$ such that $T$ contains infinitely many extensions of $(x(0), x(1), \ldots, x(n), i)$. Since $T$ is finite-branching, it follows by induction that such an $i$ always exists. $\square$

Notice that the path defined in the proof of König's Lemma is not necessarily computable, despite the "recursive" definition. The complexity of this path will be considered further below in Chapter 3.

We can use König's Lemma to determine the complexity of $Ext(T)$ for a computable tree $T$.

**Theorem 2.2.13.** *Let $T$ be a computable tree in $\mathbb{N}^*$.*

*(a) $T$, $Ext(T)$ is $\Sigma^1_1$.*

*(b) For a finite-branching tree $T$, $Ext(T)$ is $\Pi^0_2$.*

*(c) For a highly computable tree $T$, $Ext(T)$ is $\Pi^0_1$.*

*Proof.* In general, we have

$$\sigma \in Ext(T) \iff (\exists x)[\sigma \prec x \ \& \ (\forall n) x \restriction n \in T].$$

For a finite branching tree, König's Lemma implies that

$$\sigma \in Ext(T) \iff (\forall n)(\exists \tau \in \mathbb{N}^*)[|\tau| = n \ \& \ \sigma^\frown\tau \in T].$$

Part (c) is left as an exercise. $\square$

We now present the fundamental basis result is due to Kleene.

**Theorem 2.2.14** (Kleene)**.** *For any tree $T$ such that the $\Pi^0_1$ class $P = [T]$ is nonempty, $P$ contains a member which is computable in $Ext(T)$.*

*Proof.* The infinite path $x$ through $T$ can be defined recursively by letting $x(0)$ be the least $n$ such that $(n) \in Ext(T)$ and, for each $k$, letting $x(k+1)$ be the least $n$ such that $(x(0), \ldots, x(k), n) \in Ext(T)$. $\square$

Combining this with Theorem 2.2.13, we get the following:

**Theorem 2.2.15.** *For any nonempty $\Pi^0_1$ class $P \subseteq \mathbb{N}^{\mathbb{N}}$:*

*(a) $P$ has a member computable in some $\Sigma^1_1$ set;*

(b) *if $P$ is bounded, then $P$ has a member of $\Sigma^0_2$ degree (hence computable in $\mathbf{0}''$);*

(c) *if $P$ is c. b., then $P$ has a member of c.e. degree (hence computable in $\mathbf{0}'$);*

(d) *if $P$ is decidable, then $P$ has a computable member.* □

*Proof.* Part (a) and the parenthetical consequences in parts (b) and (c) are immediate from Theorems 2.2.13 and 4.2.3. The existence of a member of c.e. $\Sigma^0_2$ degree in (b) and of a member of c.e. degree in (c) are left as an exercise. □

The following corollary is very useful.

**Corollary 2.2.16.** *Any isolated element of a computably bounded $\Pi^0_1$ class is computable.*

*Proof.* Without loss of generality, Let $x \in \mathbb{N}^{\mathbb{N}}$ and let $P = \{x\}$ be a c.b. $\Pi^0_1$ class. Let $P = [T]$ where $T$ is computable and computably bounded and let $f$ be given so that $\sigma(m) < f(n)$ for all $\sigma \in \mathbb{N}^n$ and all $m < n$. Then $Ext(T)$ is $\Pi^0_1$ by Theorem 2.2.13. But for any $\sigma \in Ext(T)$, $\sigma = x\lceil|\sigma|$ and is the unique member of $Ext(T)$ with length $|\sigma|$. Thus we also have

$$\sigma \in Ext(T) \iff (\forall \tau \in \{0, 1, \ldots, f(n)\}^{|}\sigma|)[\tau \in Ext(T) \implies \tau = \sigma]/$$

This shows that $Ext(T)$ is also c. e. and is therefore computable, so that $x$ is computable by Theorem 2.2.15. □

Part (a) is the Kleene basis theorem [97] and part (c) is the Kreisel basis theorem [104].

For two $\Pi^0_1$ classes $P = [S]$ and $Q = [T]$, define the *amalgamation* of $P$ and $Q$, $P \otimes Q$, by $P \otimes Q = \{x \oplus y : x \in P \ \& \ y \in Q\}$. Then it is clear that $P \otimes Q = [S \otimes T]$. More generally, define the infinite amalgamation $\otimes_i S_i$ to be those strings $\sigma$ such that for each $i$, $(\sigma([i, 0]), \sigma([i, 1]), \ldots, \sigma([i, j])) \in S_i$, where $j$ is the maximum such that $[i, j] < |\sigma|$. Then $[\otimes_i S_i]$ is isomorphic to the direct product $\Pi_i[S_i]$.

We also wish to consider the disjoint union. For two $\Pi^0_1$ classes $P = [S]$ and $Q = [T]$, $P \oplus Q = \{0^\frown x : x \in P\} \cup \{1^\frown y : y \in Q\}$. It is easy to see that $P \oplus Q = [S \oplus T]$.

Another consequence of compactness is the dual reduction property of $\Pi^0_1$ classes in $\{0, 1\}^{\mathbb{N}}$. Some definitions are needed.

**Definition 2.2.17.** *A family $\Gamma$ of subsets of some set $X$ has the* Reduction Property *if, for any $A$ and $B$ in $\Gamma$, there exist $A_1$ and $B_1$ in $\Gamma$ such that*

(i) *$A_1 \subseteq A$ and $B_1 \subseteq B$;*

(ii) *$A_1 \cup B_1 = A \cup B$;*

(iii) *$A_1 \cap B_1 = \emptyset$.*

A standard result from classical computability theory is the following.

**Proposition 2.2.18.** *The family of c.e. subsets of* $\mathbb{N}$ *satisfy the reduction property.*

*Proof.* Given c.e. sets $A$ and $B$, let $f$ and $g$ be computable functions which enumerate $A$ and $B$, respectively. Then define

$$A_1 = \{f(n) : (\forall m < n) f(n) \neq g(m)\}$$

and

$$B_1 = \{g(m) : (\forall n \leq m) g(m) \neq f(n)\}.$$

That is, $i = f(n)$ is enumerated into $A_1$ as long as it has not already appeared in $B$ and similarly $j = g(m)$ is enumerated into $B_1$ as long as it does not come into $A$ by stage $m$. $\square$

We can modify this to show that the $\Sigma^0_1$ classes in $\mathbb{N}^{\mathbb{N}}$ also have the reduction property.

**Proposition 2.2.19.** *The family of* $\Sigma^0_1$ *classes in* $\mathbb{N}^{\mathbb{N}}$ *has the reduction property.*

*Proof.* Since $A$ and $B$ are the complements of $\Pi^0_1$ classes, it follows from Theorem 2.3.2 that there exist computable functions $f$ and $g$ such that $A = \bigcup_n I(\sigma_{f(n)})$ and $B = \bigcup_m I(\sigma_{g(m)})$ where $\sigma_{f(n_1)}$ and $\sigma_{f(n_2)}$ are incompatible for $n_1 \neq n_2$ and similarly for $g$. Now define computable sequences $U_n$ and $V_n$ of clopen sets by

$$U_n = I(\sigma_{f(n)}) \setminus \bigcup_{m<n} I(\sigma_{g(m)})$$

and

$$V_m = I(\sigma_{g(m)}) \setminus \bigcup_{n \leq m} I(\sigma_{f(n)}).$$

Then let $A_1 = \bigcup_n U_n$ and $B_1 = \cup_m V_m$. $\square$

**Definition 2.2.20.** *A family* $\Gamma$ *of subsets of some set* $X$ *has the* Dual Reduction Property *if, for any* $P$ *and* $Q$ *in* $\Gamma$*, there exist* $P_1$ *and* $Q_1$ *in* $\Gamma$ *such that*

*(i)* $P \subseteq P_1$ *and* $Q \subseteq Q_1$*;*

*(ii)* $P_1 \cap Q_1 = P \cap Q$*;*

*(iii)* $P_1 \cup Q_1 = X$*.*

This is the dual of the usual reduction property and was first studied by Herrmann. It will be important in the study of the lattice $\mathcal{E}_\Pi$ in Chapter 16. It is easy to see that a family $\Gamma$ will satisfy the reduction property if and only if the family of complements of $\Gamma$ satisfies the dual reduction property. (See the exercises.)

**Corollary 2.2.21.** *The family of* $\Pi^0_1$ *classes in* $\{0,1\}^{\mathbb{N}}$ *has the dual reduction property.* $\square$

**Definition 2.2.22.** *A class $\Gamma$ of subsets of some set $X$ has the* Separation Property *if for any $P$ and $Q$ in $\Gamma$, if $P \cap Q = \emptyset$, then there exists $R$ such that both $R$ and $X \setminus R$ are in $\Gamma$ and $P \subset R \subset X \setminus Q$. $R$ is said to* separate *$P$ and $Q$.*

If $\Gamma$ is the family of $\Pi^0_1$ classes in $\{0,1\}^{\mathbb{N}}$ (or in general, the family of closed subsets of $\{0,1\}^{\mathbb{N}}$), then $R$ and $\{0,1\}^{\mathbb{N}} \setminus R$ are both closed if and only if $R$ is clopen.

The following result is left as an exercise.

**Proposition 2.2.23.** *If $\Gamma$ has the dual reduction property, then it also has the separation property.* □

**Corollary 2.2.24.** *[Separation Property] The family of $\Pi^0_1$ classes has the separation property.* □

It is well-known that the c.e. sets do not satisfy the separation property. (See section 2.5 below.) This also carries over to the $\Sigma^0_1$ classes.

**Proposition 2.2.25.** *The family of $\Sigma^0_1$ classes does not have the separation property.*

*Proof.* Let $A$ and $B$ be disjoint, computably inseparable c.e. sets and let $U = \cup_{n \in A} I(0^n 1)$ and $V = \cup_{n \in B} I(0^n 1)$. Suppose by way of contradiction that $G$ is a clopen set such that $U \subset G$ and $V \cap G = \emptyset$. Since $G$ is closed, $0^\omega \in G$. Since $G$ is open, there must be some finite $m$ such that $I(0^m) \subset G$. But then there must be some $n \in B$ with $n > m$, so that $0^n 1^\omega \in V \cap G$. □

**Exercises**

2.2.1. Prove Lemma 2.2.7(a).

2.2.2. Show that $T_P$ is the least tree $T$ such that $P = [T]$, that is, the intersection of all such trees.

2.2.3. Explain why the computable tree $T$ in Example 2.2.6 cannot be finite-branching. Where does the infinite branching occur?

2.2.4. Show that $P$ as defined in Example 2.2.3 is a $\Pi^0_1$ class and that $T_P$ is not computable.

2.2.5. Let $P = \{0,1\}^{\mathbb{N}}$. Show that there are infinitely many computable trees in $\{0,1,2\}^{\mathbb{N}}$ such that $P = [T]$ and continuum many non-computable trees $T$ in $\{0,1,2,3\}^{\mathbb{N}}$ with $P = [T]$.

2.2.6. Prove Lemma 2.2.10.

2.2.7. Show that $Ext(T)$ is a $\Pi^0_1$ set for a highly computable tree $T$.

2.2.8. Complete the proof of Theorem 2.2.15. Hint: If $T$ is a computably bounded tree, then the leftmost infinite path of $T$ is a c.e. real, as defined in Section 1.8.

2.2.9. Verify that $[S \oplus T] = [S] \oplus [T]$ and that $[S \otimes T] = [S] \otimes [T]$.

2.2.10. Show that $\Gamma$ has the reduction property if and only if $\tilde{\Gamma} = \{X \setminus S : S \in \Gamma\}$ has the dual reduction property.

2.2.11. Prove Proposition 2.2.23

2.2.12. Show that the family of all open sets in $\mathbb{N}^{\mathbb{N}}$ has the reduction property and hence the class of closed sets has both the dual reduction property and the separation property.

2.2.13. Improve Proposition 2.2.25 by showing that the $\Sigma^0_1$ classes defined in the proof cannot be separated by any *decidable* $\Pi^0_1$ class.

## 2.3 Effectively Closed Sets in the Arithmetic Hierarchy

The following lemma makes the connection between trees and quantified relations precise.

**Proposition 2.3.1.** *For any class $P \subset \mathbb{N}^{\mathbb{N}}$, the following are equivalent:*

*(a) $P = [T]$ for some computable tree $T \subset \mathbb{N}^*$;*

*(b) $P = [T]$ for some primitive recursive tree $T$;*

*(c) $P = \{x : (\forall n)R(n, x)\}$, for some computable relation $R$;*

*(d) $P = [T]$ for some $\Pi^0_1$ tree $T \subset \omega^{<\omega}$;*

*Proof.* : [(a) $\to$ (b)]: Suppose that $P = [T]$, where $T$ is a computable tree and let $\phi_e$ be a total $\{0, 1\}$-valued computable function such that $\sigma \in T$ if and only if $\phi_e(\sigma) = 1$. Define the primitive recursive tree $S$ by $\tau \in S \iff (\forall n < |\tau|)\neg\phi_{e,|\tau|}(\tau\lceil n) = 0$. Clearly $T \subset S$, so that $[T] \subset [S]$. Suppose now that $x \notin [T]$. Then for some $n$, $x\lceil n \notin T$. Thus we have some $m$ such that $\phi_{e,m}(x\lceil n) = 0$. Then for any $k > max\{m, n\}$, we clearly have $x\lceil k \notin S$. It follows that $x \notin [S]$.

[(b) $\to$ (c)]: Suppose that $P = [T]$, where $T$ is a primitive recursive tree. Define the relation $R$ by $R(n, x) \iff x\lceil n \in T$. then we have $x \in [T] \iff (\forall n)x\lceil n \in T \iff (\forall n)R(n, x)$.

[(c) $\to$ (d)]: Suppose that $x \in P \iff (\forall n)R(n, x)$ where $R$ is a computable relation, that is, there is a computable functional $\Phi = \Phi_e$ such that $R(n, x) \iff \Phi(n, x) = 1$ and $\neg R(n, x) \iff \Phi(n, x) = 0$. By the Master Enumeration Theorem II.1.6.5, we have $\Phi(n, x) = i$ if and only if $\Phi(n, x\lceil m) = i$ for some $m$. Define the tree $T$ by

$$\sigma \in T \iff (\forall n < |\sigma|)\Phi(n, \sigma) \downarrow \to \Phi(n, \sigma) = 1.$$

It is clear that $P = [T]$.

[(d) $\to$ (a)]: Suppose that the tree $T$ is a $\Pi^0_1$ subset of $\omega^{<\omega}$, so that there is a computable relation $R$ such that $\sigma \in T \iff (\forall n)R(n,\sigma)$. Define the computable tree $S \supset T$ by $\sigma \in S \iff (\forall m, n \leq |\sigma|)R(m, \sigma\lceil n)$. It is easily verified that $[T] = [S]$. $\square$

Standard topology tells us that a closed set may be defined as the complement of an open set, and that an open set in $\mathbb{N}^{\mathbb{N}}$ is a countable union of intervals $I(\sigma)$. Since $\mathbb{N}^{\mathbb{N}}$ is completely disconnected, this countable union can be made disjoint. The effective version of this fact is the following. Let $\sigma_0, \sigma_1, \dots$ be an effective enumeration of $\mathbb{N}^*$. (This can be done in order first by the sum $\sigma(0) + \sigma(1) + \dots + \sigma(|\sigma|)$ and then lexicographically.)

**Theorem 2.3.2.** *(a) If $P \subseteq \mathbb{N}^{\mathbb{N}}$ is a $\Pi^0_1$ class, then there is a computable set $W$ such that $\mathbb{N}^{\mathbb{N}} \setminus P = \bigcup_{n\in W} I(\sigma_n)$. Furthermore, the set $W$ may be chosen so that $I(\sigma_m)$ and $I(\sigma_n)$ are disjoint for $m \neq n$.*

*(b) For any c.e. set $W \subseteq \mathbb{N}^*$, $\mathbb{N}^{\mathbb{N}} \setminus \bigcup_{n\in W} I(\sigma)$ is a $\Pi^0_1$ class.*

*Proof.* (a): Let $P$ be a $\Pi^0_1$ class and let $T$ be a computable tree such that $P = [T]$. Let $W = \{n : \sigma_n \text{ is a dead end of } T\}$. Then in fact $W$ is a computable set and clearly $\mathbb{N}^{\mathbb{N}} \setminus P = \bigcup_{n\in W} I(\sigma_n)$. That is, if $x \in I(\sigma_n)$ for some $n \in W$ and $k = |\sigma_n|$, then $x \restriction k = \sigma_n \notin T$, so that $x \notin P$. On the other hand, if $x \notin P$, then there is a least $k$ such that $x \restriction k \notin T$. Take $n$ so that $x \restriction k = \sigma_n$. Then $n \in W$ and $x \in I(\sigma_n)$. To check the disjointness condition, suppose that $\sigma_m$ and $\sigma_n$ are both dead ends of $T$. If they were comparable, then without loss of generality, $\sigma_m \prec \sigma_n$, which contradicts the assumption that both are dead ends.

(b) Given the c.e. set $W \subseteq \mathbb{N}^*$, let $P = \mathbb{N}^{\mathbb{N}} \setminus \bigcup_{n\in W} I(\sigma) = [T]$ and define the $\Pi^0_1$ tree $T$ by

$$\tau \in T \iff (\forall \sigma \preceq \tau)\sigma \notin W.$$

Then $\mathbb{N}^{\mathbb{N}} \setminus \bigcup_{n\in W} I(\sigma) = [T]$. It follows from Proposition 2.3.1 that $P$ is a $\Pi^0_1$ class. $\square$

Part (d) of Proposition 2.3.1 may be used to define an enumeration of the $\Pi^0_1$ classes. That is, let we could let $P_e = \mathbb{N}^{\mathbb{N}} \setminus \bigcup_{n\in W_e} I(\sigma_n)$, where $\sigma_0, \sigma_1, \dots$ is the standard enumeration of $\{0,1\}^*$ in length-lexicographic order. By applying Proposition 2.3.1 uniformly, we obtain the following.

**Theorem 2.3.3.** *There is a uniformly primitive recursive sequence $\langle T_e \rangle_{e\in\omega}$ of trees such that $\langle [T_e] \rangle_{e\in\omega}$ enumerates the $\Pi^0_1$ classes.*

*Proof.* Simply let

$$\sigma \in T_e \iff (\forall n < |\sigma|)[\sigma_n \preceq \sigma \to n \notin W_{e,|\sigma|}.$$

$\square$

The official enumeration for the $\Pi^0_1$ classes will be defined below in Chapter 5 and is based directly on primitive recursive trees.

There is another notion equivalent to being a computably bounded $\Pi^0_1$ class.

**Proposition 2.3.4.** *Let $P \subset \mathbb{N}^{\mathbb{N}}$ and $h$ a computable function such that $x(n) < h(n)$ for all $x \in P$. Then $P$ is a $\Pi^0_1$ class if and only if $T_P$ is a $\Pi^0_1$ set.*

*Proof.* If $T_P$ is a $\Pi^0_1$ set, then $P$ is a $\Pi^0_1$ class by Proposition 2.3.1 since $P = [T_P]$. Now suppose that $h$ is a computable function and $x(n) < h(n)$ for all $x \in P$ and all $n$. Then by König's Lemma,

$$\sigma \in T_P \iff (\forall k > |\sigma|)(\exists \tau \in h(0) \times h(1) \times \ldots h(k))[\tau \in T \;\&\; \sigma \prec \tau].$$

□

We will also consider in general the families of $\Pi^0_n$ classes and $\Sigma^0_n$ classes. Analogous to the definition of $\Pi^0_n$ sets, we have the following.

**Definition 2.3.5.** *Let $R$ be a relation on $\mathbb{N}^k \times \mathbb{N}^{\mathbb{N}}$ and let $n > 0$ be a natural number.*

1. *$R$ is $\Pi^0_0$ if it is computable.*

2. *$R$ is $\Sigma^0_{n+1}$ if there is a $\Pi^0_n$ relation $Q \subset \mathbb{N}^{k+1} \times \mathbb{N}$ such that*

$$R(a_1, \ldots, a_k, x) \iff (\exists i) Q(i, a_1, \ldots, a_k, x).$$

3. *$R$ is $\Pi^0_{n+1}$ if $\mathbb{N}^k \times \mathbb{N}^{\mathbb{N}} \setminus R$ is $\Sigma^0_{n+1}$.*

4. *$R$ is $\Delta^0_{n+1}$ if $R$ is both $\Sigma^0_{n+1}$ and $\Pi^0_{n+1}$.*

Note that a computable class in $\mathbb{N}^{\mathbb{N}}$ is both open and closed. These definitions can also be relativized to a set oracle $C$, but the results are not quite analogous to those for sets.

**Definition 2.3.6.** *(i) A subset of $\mathbb{N}^{\mathbb{N}}$ is a* **strong** *$\Pi^0_{n+1}$ class if $P = [T]$ for some tree $T$ computable in $\emptyset^{(n)}$.*

*(ii) A strong $\Pi^0_{n+1}$ class $P$ is* highly bounded *if $P = [T]$ for some tree $T$ computable in $\emptyset^{(n)}$ and a bounding function $f$ also computable in $\emptyset^{(n)}$ such that $\sigma(n) \leq f(n)$ for all $\sigma \in T$.*

**Proposition 2.3.7.** *For any class $P \subset \mathbb{N}^{\mathbb{N}}$, the following are equivalent:*

*(a) $P = [T]$ for some tree $T$ computable in $\mathbf{0}^{(n)}$.*

*(b) $P = [T]$ for some $\Pi^0_{n+1}$ tree $T$.*

*(c) $P = [T]$ for some $\Sigma^0_n$ tree $T$.*

*Furthermore, if $P \subset \{0,1\}^{\mathbb{N}}$, then $T \subset \{0,1\}^*$.*

*Proof.* The equivalence of (a) and (b) follows from a relativization of Proposition 2.3.1. Clearly (c) implies (b). It remains to be shown that (b) implies (c). Let $T$ be a $\Pi^0_{n+1}$ tree such that $P = [T]$. Then there is a $\Sigma^0_n$ relation $R$ such that

$$\sigma \in T \iff (\forall i)R(i, \sigma),$$

so that

$$x \in P \iff (\forall m)(\forall i)R(i, x \upharpoonright m).$$

Now define the $\Sigma^0_n$ tree $S$ by

$$\sigma \in S \iff (\forall m \leq |\sigma|)(\forall i \leq |\sigma|)R(i, \sigma \upharpoonright m).$$

It is easy to check that $S$ is a tree and that $P = [S]$. $\square$

**Exercises**

2.3.1. Complete the proof of Proposition 2.3.7 by showing that $S$ is a tree and that $P = [S]$.

## 2.4 Graphs of Computable Functions

In classical computability theory, computable functions and computably enumerable sets are the two primary objects of study. In this context, the two are naturally related. For any (partial) computable function, the domain, the range and the graph are all c.e. sets. Furthermore, every c.e. set is the domain of a computable function and every nonempty c.e. set is the range of a total computable function. In addition, any partial function with a c.e. graph is necessarily computable. For our purposes, computably continuous functions and $\Pi^0_1$ classes are the primary objects of study. In this section, we explore possible analogues of the classical results.

Recall from the Master Enumeration Theorem II.1.6.5 that a (partial) computable function $\Phi : \mathbb{N}^\mathbb{N} \to \mathbb{N}^\mathbb{N}$ may be approximated by maps on sequences. Let $\Phi(\sigma, m) = n$ if $\Phi$ computes output $n$ on input $m$ using only oracle information from $\sigma$ and in $|\sigma|$ or fewer steps; we may assume that $n < |\sigma|$. Let $\Phi(\sigma) = \tau$ denote the partial function on strings as before.

Before giving a characterization of a computably continuous function, we first consider arbitrary continuous functions.

**Lemma 2.4.1.** *A function $F : \mathbb{N}^\mathbb{N} \to \mathbb{N}^\mathbb{N}$ (respectively, $F : \{0,1\}^\mathbb{N} \to \{0,1\}^\mathbb{N}$) is continuous if and only if there is a function $f : \mathbb{N}^* \to \mathbb{N}^*$ (resp. $f : \{0,1\}^* \to \{0,1\}^*$) such that*

1. *for all $\sigma \prec \tau$, $f(\sigma) \preceq f(\tau)$;*
2. *for all $x \in \mathbb{N}^\mathbb{N}$ ($\{0,1\}^\mathbb{N}$), $\lim_{n\to\infty} |f(x \upharpoonright n)| = \infty$;*
3. *for all $x \in \mathbb{N}^\mathbb{N}$ ($\{0,1\}^\mathbb{N}$), $\cup_n f(x \upharpoonright n) = F(x)$.*

*Proof.* Since $F$ is continuous, it follows that $F^{-1}(I(\tau))$ is open for each $\tau \in \mathbb{N}^*$. Define $f(\sigma)$ to be the unique longest $\tau$ such that $I(\sigma) \subseteq F^{-1}(I(\tau))$ [equivalently, $F(I(\sigma)) \subseteq I(\tau)$].

Then $f$ is certainly monotonic.

Fix $x \in \mathbb{N}^{\mathbb{N}}$ and let $Y = F(X)$. For each $n$, $X$ belongs to the open set $F^{-1}(I(Y \upharpoonright n))$ and hence there is some basic open set $I(\sigma)$ such that $X \in I(\sigma) \subseteq F^{-1}(I(Y \upharpoonright n))$. It follows that $Y \upharpoonright n \preceq f(\sigma)$ and of course $\sigma = x \upharpoonright m$ for some $m$. Then for all $t > m$, $|f(x \upharpoonright t)| \geq n$. Thus $lim_{t\to\infty}|f(x \upharpoonright t)| = \infty$, as desired.

It now follows that $\cup_n f(x \upharpoonright n) = F(x)$. □

The fundamental notion of computable analysis is the computable version of Lemma 2.4.1.

**Lemma 2.4.2.** *A function $F : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ (respectively, $F : \{0,1\}^{\mathbb{N}} \to \{0,1\}^{\mathbb{N}}$) is computably continuous if and only if there is a computable function $f : \mathbb{N}^* \to \mathbb{N}^*$ (resp. $f : \{0,1\}^* \to \{0,1\}^*$) such that*

1. *for all $\sigma \prec \tau$, $f(\sigma) \preceq f(\tau)$;*
2. *for all $x \in \mathbb{N}^{\mathbb{N}}$ ($\{0,1\}^{\mathbb{N}}$), $\lim_{n\to\infty} |f(x \upharpoonright n)| = \infty$;*
3. *for all $x \in \mathbb{N}^{\mathbb{N}}$ ($\{0,1\}^{\mathbb{N}}$), $\cup_n f(x \upharpoonright n) = F(x)$.*

*Proof.* Given such a representation $f$ for $F$, compute $y(n)$ for $y = F(x)$ from $x$ by computing $f(x \upharpoonright k)$ for sufficiently large $k$.

Given a computable function $F$, define the representation $f$ as follows. On input $\sigma$ of length $n$, compute the values of $\tau = f(\sigma)$ for each $i < n$ by applying the algorithm for $F$ for $n$ steps, using oracle $\sigma$. The length of $\tau$ will be the least $k < n$ such that $\tau(k)$ does not converge in $n$ steps. □

*Remark*: The *modulus of convergence* function $\mu$ of $F : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ as defined from the approximation map $f : \mathbb{N}^* \to \mathbb{N}^*$ may be defined by $\mu(x, n) = (least\ s)|f(x \upharpoonright n)| > s$. For a total computable function $F$, this modulus function is also computable. The following lemma will be useful. The proof is left as an exercise.

**Lemma 2.4.3.** *For any computable function $F : \{0,1\}^{\mathbb{N}} \to \{0,1\}^{\mathbb{N}}$, there is a computable uniform modulus function $\mu : \mathbb{N} \to \mathbb{N}$ such that for all $x \in \{0,1\}^{\mathbb{N}}$, $|f(x\lceil\mu(n)| > n$.* □

**Theorem 2.4.4.** *Let $\Phi : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ be a (partial) computable function. Then the graph of $\Phi$ is a $\Pi^0_2$ class. Furthermore, if $\Phi$ is total, then the graph is a decidable $\Pi^0_1$ class.*

*Proof.* Let $\phi$ be a representing function for $\Phi$. In general, we have,

$$\Phi(x) = y \iff (\forall m)(\exists k)[\phi(x \upharpoonright k)(m) = y(m)].$$

For a total function, define the computable tree $T$ with $graph(\Phi) = [T]$ by putting $\sigma \oplus \tau \in T$ if and only if $\tau$ is consistent with $\Phi(\sigma)$, that is, for any $m$, if $\Phi(\sigma, m) = n$, then $\tau(m) = n$. Then $Ext(T)$ is $\Sigma^0_1$ and hence computable, since

$$\sigma \oplus \tau \in Ext(T) \iff (\exists \rho)[\sigma \prec \rho \ \& \ \tau \prec \Phi(\rho)].$$

To see this, note that if $\tau \preceq \Phi(\rho)$ and $\sigma \preceq \rho$, then for *any* $x \in I(\sigma)$, $\tau \prec F(x)$. $\square$

**Example 2.4.5.** *The graph of a partial computable function need not be closed. Define the partial function $\Phi : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ by $\Phi(x)(m) = 0 \cdot [(least\ n)x(n) = 1]$. For each $n$, let $x_n = 0^n{}^\frown 1^\frown 0^\infty$, so that $lim_n x_n = 0^\infty$. Then, for each $n$, $F(x_n) = 0^\infty$, whereas $F(0^\infty)$ is undefined.*

For total functions on $\{0,1\}^{\mathbb{N}}$, there is a converse.

**Theorem 2.4.6.** *A function $F : \{0,1\}^{\mathbb{N}} \to \{0,1\}^{\mathbb{N}}$ is computably continuous if and only if the graph is a $\Pi^0_1$ class.*

*Proof.* One direction follows from Theorem 2.4.4. Next suppose that $F : \{0,1\}^{\mathbb{N}} \to \{0,1\}^{\mathbb{N}}$ and let $T$ be a computable tree such that $graph(F) = [T]$. Define a computable function $f$ on strings by letting $f(\sigma)$ be the common part of $\{\tau : \sigma \oplus \tau \in T\}$. $\square$

**Theorem 2.4.7.** *A subset $D$ of $\mathbb{N}^{\mathbb{N}}$ is a $\Pi^0_2$ class if and only if $D$ is the domain of some partial computable function $\Phi : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$.*

*Proof.* Suppose first that $D$ is the domain of $\Phi$. Then

$$x \in Dom(\Phi) \iff (\forall n)(\exists k)[|\Phi(x \upharpoonright k)| > k].$$

Next suppose that $D$ is a $\Pi^0_2$ class and let $R$ be a computable relation such that

$$x \in D \iff (\forall n)(\exists k)R(n, x\lceil k).$$

Then $D$ is the domain of the partial computable function $\Phi$ defined by

$$\Phi(x)(n) = (leastk)R(n, x\lceil k).$$

$\square$

We next examine the complexity of the image of a $\Pi^0_1$ class under a computably continuous function. The classical results is that the image of any compact set under a continous function is compact and that the image of a closed set is an analytic set.

**Theorem 2.4.8.** *Let $F$ be a computably continuous function on a $\Pi^0_1$ subclass $P$ of $\mathbb{N}^{\mathbb{N}}$ and let $F[P] = \{F(x) : x \in P\}$. Then*

1. *$F[P]$ is a $\Sigma^1_1$ class;*

2. *if $P$ is bounded, then $F[P]$ is a strong $\Pi^0_2$ class;*

3. *if $P$ is computably bounded, then $F[P]$ is a computably bounded $\Pi^0_1$ class and, furthermore, if $P$ is decidable, then $F[P]$ is decidable.*

*Proof.* (1) We have $y \in F[P] \iff (\exists x)(x \in P \;\&\; x \oplus y \in graph(F))$.

(2) Suppose that $T$ is a finite branching, computable tree and let $S$ be a computable tree such that $graph(F) = [S]$. Then it follows from König's Lemma that $F[P] = [R]$, for the finite branching $\Sigma^0_1$ tree $R$ defined by

$$\tau \in R \iff (\exists \sigma)[\sigma \in T \textit{ and } \sigma \oplus \tau \in S].$$

(3) Now suppose that $T$ is computably bounded and let $F$ be represented by the computable function $f : \mathbb{N}^* \to \mathbb{N}^*$. Then the definition of $R$ above in (2) becomes computable since the $(\exists\sigma)$ quantifier becomes bounded.

To find a bound for the possible value of $\tau(n)$ for $\tau \in R$, compute the least $m$ such that $|f(\sigma)| > n$ for all $\sigma \in T$ of length $m$. Then we compute the maximum value $h(r)$ of $f(\sigma(n))$ for all $\sigma \in T$ of length $n$. Thus $R$ is seen to be highly computable. □

This result has a converse for $\{0,1\}^{\mathbb{N}}$.

**Theorem 2.4.9.** *A $\Pi^0_1$ subclass $P$ of $\{0,1\}^{\mathbb{N}}$ is the computably continuous image of $\{0,1\}^{\mathbb{N}}$ if and only if it is decidable.*

*Proof.* Let $P = T$, where $T$ is a computable tree with no dead ends. We will define the computable map $f : \{0,1\}^* \to \{0,1\}^*$ which represents a map $F : \{0,1\}^{\mathbb{N}} \to \{0,1\}^{\mathbb{N}}$ such that $y = F(x)$ is some element of $P$ which is nearest to $x$. Let $f(\sigma) = \sigma$ if $\sigma \in T$ and, if $\sigma \notin T$, let $\rho$ be the longest initial segment of $\sigma$ which is in $T$ and let $f(\sigma)$ be the lexicograpical least extension of $\rho$ which is in $T$ and has length $|\sigma|$. □

**Exercises**

2.4.1. Prove Lemma 2.4.3.

2.4.2. A mapping $f$ from $\mathbb{N}^*$ (or $\{0,1\}^*$) into $\mathbb{N}^*$ is a *tree homomorphism* if $\sigma \prec \tau$ implies $f(\sigma) \prec f(\tau)$ for all $\sigma$ and $\tau$. Show that for any tree homomorphism $f$, $T = \{\tau : (\exists\sigma) : f(\sigma) \prec \tau\}$ is a tree and that if $f$ is one-to-one and computable, then $T$ is a computable tree and $[T]$ is a perfect $\Pi^0_1$ class.

2.4.3. Show that for $Rx \in \{0,1\}^{\mathbb{N}}$, $x$ is computable if and only if $\{x\}$ is a $\Pi^0_1$ class.

2.4.4. Show that for any computable $x \in \{0,1\}^{\mathbb{N}}$ and any computable function $F : \{0,1\}^{\mathbb{N}} \to \{0,1\}^{\mathbb{N}}$, $F(x)$ is computable.

## 2.5 Computably enumerable sets and $\Pi_1^0$ Classes

There are numerous connections between computable functions, $\Pi_1^0$ and c.e. subsets of $\mathbb{N}$, and $\Pi_1^0$ classes. We will consider in particular the class $S(A, B)$ of separating sets for a pair of c.e. sets $A, B$, and the class $I(C)$ of initial subsets of a $\Pi_1^0$ set $C$. Then we will take a first look at the notion of *forbidden words* and *sets* and their connection with $\Pi_1^0$ classes and with *subshifts*. In this section, the notion of retraceability

### 2.5.1 Separating classes

The most basic example here is that, for any $\Pi_1^0$ set $C$, the power set $\mathcal{P}(C)$ is a $\Pi_1^0$ class. That is, we have

$$x \subset C \iff (\forall n)[x(n) = 1 \rightarrow n \in C].$$

More generally, consider the notion of *separating sets*.

**Definition 2.5.1.** *Let $A$ and $B$ be infinite disjoint c.e. sets and let $C \subset \mathbb{N}$.*

1. *$C$ is a separating set for $A$ and $B$ if $A \subset C$ and $B \cap C = \emptyset$.*
2. *$A$ and $B$ are said to be computably (or recursively) inseparable if there is no computable separating set $C$ for $A$ and $B$.*
3. *The class of separating sets for $A$ and $B$ is denoted by $S(A, B)$.*
4. *$P$ is a separating class if $P = S(A, B)$ for some c.e. sets $A$ and $B$.*

Of course, $C \in S(A, B)$ if and only if $C \in \mathcal{P}(\mathbb{N} \setminus B)$ and $\mathbb{N} \setminus C \in \mathcal{P}(\mathbb{N} \setminus A)$. The notion of computably inseparable sets was introduced by Kleene in [98]. Shoenfield showed in [174] that every non-computable c. e. degree contains a pair of computably inseparable sets. Shoenfield observed in [174] that the class $S(A, B)$ of separating sets for $A$ and $B$ is a c. b. $\Pi_1^0$ class. We note that $S(A, B)$ is finite if and only if $A \cup B$ is cofinite, in which case $A$ and $B$ are both computable and every separating set is also computable. Otherwise, $S(A, B)$ is a perfect set and thus has the cardinality of the continuum. In either case, both the c. e. set $A$ and the co-c. e. set $\mathbb{N} \setminus B$ are of course separating sets for $A$ and $B$.

The classic example of a separating class comes from the notion of *diagonally non-computable* sets. Here a function $f \in \{0, 1\}^{\mathbb{N}}$ is diagonally non-computable if $f(e) \neq \phi_e(e)$ whenever $\phi_e(e)$ converges. Let $K_i = \{e : \phi_e(e) = i\}$. Then in particular, $K_0$ and $K_1$ are c.e. non-computable sets and any separating set for $K_0$ and $K_1$ has a diagonally non-computable characteristic function. (See the exercises.) Separating classes are important for the study of reverse mathematics and so will be examined further in Chapter 6. The complexity of the members of separating classes will be studied in Chapter 3.

For $\sigma, \tau \in \{0, 1\}^*$, let $\sigma \subseteq \tau$ mean that, for all $i \leq min\{|\sigma|, |\tau|\}$, $\sigma(i) = 1$ implies $\tau(i) = 1$. Define $\sigma \cup \tau$ to be the sequence $\rho$ of length $max\{|\sigma|, |\tau|\}$ such that $\rho(i) = max\{\sigma(i), \tau(i)\}$ and similarly define $\sigma \cap \tau$ to be the sequence $\rho$ of length $max\{|\sigma|, |\tau|\}$ such that $\rho(i) = min\{\sigma(i), \tau(i)\}$

Recall that for any closed subset $P$ of $\mathbb{N}^{\mathbb{N}}$, $T_P = \{\sigma \in \{0,1\}^* : I(\sigma) \cap P \neq \emptyset\}$.

**Lemma 2.5.2.** *Suppose that $P$ is a closed subset of $\{0,1\}^{\mathbb{N}}$.*

1. *$P$ is closed under subsets if and only if for every $\sigma \subset \tau$, if $\tau \in T_P$, then $\sigma \in T_P$.*
2. *$P$ is closed under supersets if and only if for every $\sigma \subset \tau$, if $\sigma \in T_P$, then $\tau \in T_P$.*
3. *$P$ is closed under union if and only if, for every $\sigma$ and $\tau$ in $T_P$, $\sigma \cup \tau \in T_P$.*
4. *$P$ is closed under intersection if and only if, for every $\sigma$ and $\tau$ in $T_P$, $\sigma \cap \tau \in T_P$.*

*Proof.* We prove (1) and (3) and leave (2) and (4) to the reader.

(1) Suppose $P$ is closed under subsets. Let $\sigma \subseteq \tau$ and $\tau \in T_P$. Then there exists $B \in P$ such that $\tau \preceq B$. Let $C = \{i \in B : \sigma(i) = 1\}$. Then $C \subseteq B$ so $C \in P$ by assumption and clearly $\sigma \preceq C$. On the other hand, suppose that $T_P$ is closed under $\subseteq$. Let $B \in P$ and $C \subseteq B$. Then for any $n$, $C\lceil n \subseteq B\lceil n$ and $B\lceil n \in T_P$, so that $C \in P$.

(3) Suppose $P$ is closed under union and let $\sigma$ and $\tau$ be in $T_P$. Then there exist $A$ and $B$ in $P$ such that $\sigma \prec A$ and $\tau \prec B$. It follows that $\sigma \cup \tau \prec A \cup B$ and $A \cup B \in P$ by assumption, so that $\sigma \cup \tau \in T_P$. Suppose next that $T_P$ is closed under union and let $A, B \in P$. Fix $n$ and let $\sigma = A \upharpoonright n$ and $\tau = B \upharpoonright n$. Then $(A \cup B) \upharpoonright n = \sigma \cup \tau \in T_P$ since both $\sigma, \tau \in T_P$. Hence $A \cup B \in P$. $\square$

**Lemma 2.5.3.** *Let $P$ be a closed subset of $\{0,1\}^{\mathbb{N}}$. Then $P = \mathcal{P}(A)$ for some $A$ if and only if $P$ is closed under subsets and $P$ is closed under union.*

*Proof.* If $P = \mathcal{P}(A)$, then $P$ is certainly closed under subsets and union. Suppose that $P$ is closed under subsets and union and let $A = \{i : (\exists \sigma \in T_P)\sigma(i) = 1\}$. Suppose first that $B \in P$. If $i \in B$, then $\sigma(i) = 1$ for $\sigma = B\lceil(i+1) \in T_P$, so that $i \in A$. Hence $B \subseteq A$. Next suppose that $B \subseteq A$. Then for each $i \in B$, there exists $\sigma_i \in T_P$ such that $\sigma_i(i) = 1$. Fix $n$ and define $\sigma \in \{0,1\}^n$ by $\sigma = \cup\{\sigma_i : i < n \;\&\; i \in B\}$. Then $\sigma \in T_P$ by Lemma 2.5.2 since $P$ is closed under union. Also, $B\lceil n \subseteq \sigma$, so that $B\lceil n \in T_P$ again by Lemma 2.5.2, since $P$ is closed under subsets. It follows that $B \in P$. $\square$

Observe that if $P$ is actually a $\Pi^0_1$ class, then $T_P$ is a $\Pi^0_1$ set and the set $A$ defined in the proof of Lemma 2.5.3 is in fact a $\Pi^0_1$ set. Thus we have the following.

**Proposition 2.5.4.** *For any nonempty $\Pi^0_1$ class $P$ of sets, the following are equivalent:*

1. *$P$ is the class of subsets of a $\Pi^0_1$ set $A$;*

2. *$P$ is the class of subsets of some set $A$;*

3. *$P$ is closed under subsets and under union.* □

There is a similar result for supersets, which is left to the exercises.

Let us say that a class $P$ of sets is closed under *between-ness* if, for any sets $X, Y, Z$, if $X \subset Y \subset Z$ and $X, Z \in P$, then $Y \in P$. It is clear that any separating class is closed under between-ness.

**Proposition 2.5.5.** *For any $\Pi^0_1$ class $P$, the following are equivalent.*

1. *$P$ is the class of separating sets of some pair $A, B$ of r. e. sets.*

2. *$P$ is the class of separating sets of some pair $A, B$*

3. *$P$ is closed under union, intersection and between-ness.*

*Proof.* It is immediate that (1) implies (2) and (2) implies (3). Suppose therefore that $P$ is closed under union, intersection and between-ness. Define the $\Pi^0_1$ class $Q$ to be the family of subsets of sets in $P$. That is, for $\sigma \in \{0,1\}^n$,

$$\sigma \in T_Q \iff (\exists \tau \in \{0,1\}^n)\sigma \subseteq \tau \ \&\ \tau \in T_P.$$

It is clear that $Q$ is closed under subsets and under union, so it follows from Proposition 2.5.4 that $Q = \mathcal{P}(C)$ for some $\Pi^0_1$ set $C$. Let $B = \mathbb{N} \setminus C$.

Similarly define the $\Pi^0_1$ class $R$ to be the family of supersets of sets in $P$. That is, for $\sigma \in \{0,1\}^n$,

$$\tau \in T_R \iff (\exists \sigma \in \{0,1\}^n)\sigma \subseteq \tau \ \&\ \tau \in T_P.$$

It follows that $R$ is the class of supersets of some c.e. set $A$.

We claim that $P = Q \cap R = S[A, B]$. Suppose first that $X \in P$. Then certainly $X \in Q \cap R$ and therefore $A \subseteq X$ and $X \cap B = \emptyset$. Next suppose that $X \in S[A, B]$. Then $A \subseteq X$, so that $X \in R$ and therefore $Y \subseteq X$ for some $Y \in P$. Also, $X \cap B = \emptyset$, so that $X \subseteq C$ and $X \in Q$, which means that $X \subseteq Z$ for some $Z \in P$. It now follows by the between-ness property that $X \in P$. □

The proof of the following corollary is left as an exercise 6.

**Corollary 2.5.6.** *For any subset $A$ of $\mathbb{N}$, if $\{A\}$ is a $\Pi^0_1$ class, then $A$ is a computable set.* □

### 2.5.2 Subsimilar classes

The notions of a *subsimilar set* (or *subshift*) and of *forbidden words* provides another link between c. e. sets and $\Pi^0_1$ classes and is also closely related to symbolic dynamics.

**Definition 2.5.7.** 1. *A finite string $\sigma \in \mathbb{N}^*$ is a* factor *of another string $\tau$ if $\tau = \tau_0{}^\frown\sigma{}^\frown\tau_1$ for some strings $\tau_0, \tau_1$.*

2. *Similarly* $\sigma \in \mathbb{N}^*$ *is a* factor *of* $x \in \mathbb{N}^{\mathbb{N}}$ *if* $x = \tau^\frown\sigma^\frown Y$ *for some strings* $\tau$ *and some* $Y \in \mathbb{N}^{\mathbb{N}}$.

3. $x \in \mathbb{N}^{\mathbb{N}}$ avoids $\sigma$ *if* $\sigma$ *is not a factor of* $x$ *and* $x$ *avoids a set* $S$ *of strings if* $x$ *avoids each* $\sigma \in S$.

4. $AV(S) \subseteq \mathbb{N}^{\mathbb{N}}$ *is the set of reals which avoid* $S$.

Similar definitions apply for strings in $\{0,1\}^*$ and infinite words in $\{0,1\}^{\mathbb{N}}$ and also for finite and infinite sequences any other alphabet $\Sigma$. In symbolic dynamics, strings are referred to as *words* and infinite sequences as infinite words. The set $S$ may be thought of a set of *forbidden words* for $AV(S)$.

The *shift* function $Shift$ is defined as follows.

**Definition 2.5.8.** 1. *For a finite string* $\tau$, $Shift(\tau) = (\tau(1), \tau(2), \ldots, \tau(|\tau|-1))$.

2. *For an infinite sequence* $x$, $Shift(x) = (x(1), x(2), \ldots)$.

3. *A set* $Q \subseteq \mathbb{N}^{\mathbb{N}}$ *is a* subshift *(or* subsimilar*) if* $Shift(x) \in Q$ *for all* $x \in Q$*; that is,* $Q$ *is closed under the subshift function.*

Informally, the shift function simply removes the first symbol of a finite or infinite sequence.

**Lemma 2.5.9.** *For any set* $S$, $AV(S)$ *is closed; if* $S$ *is c. e. , then* $AV(S)$ *is a* $\Pi^0_1$ *class.*

The proof is left as an exercise.

**Theorem 2.5.10** (Dashti [20])**.** 1. *A closed set* $Q$ *is a subshift if and only if* $Q = AV(S)$ *for some set* $S$.

2. *A* $\Pi^0_1$ *class* $Q$ *is a subshift if and only if* $Q = AV(S)$ *for some set c. e. set* $S$.

*Proof.* For the first part, it is clear that $AV(S)$ is a subshift, so we will just sketch the reverse implication. Suppose that $P \subseteq \{0,1\}^{\mathbb{N}}$ is subsimilar and closed, and let $S = \{0,1\}^* - T_P$. If $x \notin P$, then for some $n$, $x \upharpoonright n \in S$, so that $x \notin AV(S)$. On the other hand, suppose that $x \notin AV(S)$ and let $x = (x \upharpoonright n)^\frown\tau^\frown y$ for some $n < \omega$ and some $\tau \in S$. Then $\tau \notin T_P$ and thus $\tau^\frown y \notin P$. Since $P$ is subsimilar, it follows that $x \notin P$. A similar argument works for any alphabet $\Sigma$.

The proof of the effective version of this proposition is left as an exercise. $\square$

### Exercises

2.5.1. Show that the diagonally non-computable functions form a $\Pi^0_1$ class in $\{0,1\}^{\mathbb{N}}$.

2.5.2. Show that $S(K_0, K_1)$ is a $\Pi^0_1$ class of sets with no computable members, that is, $K_0$ and $K_1$ are computably inseparable.

2.5.3. Suppose that $P = [T]$ where $T$ is a tree with no dead ends. Show the following.

(a) $P$ is closed under supersets if and only if, for every $\sigma \in T$ and every $\tau$ such that $\sigma \subseteq \tau$, $\tau \in T$.

(b) $P$ is closed under intersection if and only if, for every $\sigma$ and $\tau$ in $T$, $\sigma \cap \tau \in T$.

2.5.4. Let $P$ be a closed subset of $\{0,1\}^{\mathbb{N}}$. Then $P$ is the class of supersets of some set $A$ if and only if $P$ is closed under supersets and $P$ is closed under intersection.

2.5.5. Show that, for any nonempty $\Pi^0_1$ class $P$ of sets, the following are equivalent:

(i) $P$ is the class of supersets of a $\Sigma^0_1$ set $A$;

(ii) $P$ is the class of supersets of some set $A$;

(iii) $P$ is closed under supersets and under intersection.

2.5.6. Use Proposition 2.5.5 to show that for any subset $A$ of $\mathbb{N}$, if $\{A\}$ is a $\Pi^0_1$ class, then $A$ is a computable set.

2.5.7. Show that a $\Pi^0_1$ class $Q$ is subsimilar if and only if $Q = AV(S)$ for some c. e. set $S$.

## 2.6 Retraceability

For any infinite set $A$, recall that the *principal function* $p_A$ enumerates the elements $a_0 < a_1 < \cdots$ in increasing order and that $A$ is *hyperimmune* if, for any computable function $f$, there is an $n$ such that $a_n > f(n)$. A c. e. set is *hypersimple* if its complement is hyperimmune. $A$ is said to be *retraceable* if there is a partial computable function $\phi$ such that $\phi(a_{n+1}) = a_n$ for all $n$. Retraceable sets were introduced by Dekker and Myhill [51], who proved that any retraceable noncomputable $\Pi^0_1$ set $A$ is hyperimmune. For $\Pi^0_1$ sets $A$, a stronger characterization can be given.

**Theorem 2.6.1.** *The following are equivalent for any infinite $\Pi^0_1$ set $A$:*

*(a) $A$ is retraceable*

*(b) There is a total computable function $\Phi$ such that, for all $n$, $\Phi(a_{n+1}) = a_n$ and, for all $y$, $\{x : \Phi(x) = y\}$ is finite.*

*(c) There is a total computable function $\Psi$ such that, for all $n$, $\Psi(a_n) = n$ and $\{x : \Psi(x) = n\}$ is finite*

*Proof.* Let $A = \{a_0 < a_1 < \cdots\}$ be an infinite $\Pi^0_1$ set and let $A$ be the decreasing intersection of uniformly computable sets $A_s$.

$(a) \to (b)$: Let $\phi$ be a partial computable retracing function for $A$. Assume, without loss of generality, that $\phi(a_0) = a_0$. Then for any $x$, we define $\Phi(x)$ as follows. Look for the least $s$ such that either $x \notin A_s$, or such that $\phi_s(x) = y \leq x$ converges and, for all $z$ with $y < z < x$, $z \notin A_s$. In the former case, we let $\Phi(x) = x$ and in the latter case, we let $\Phi(x) = \phi(x)$. Note that if $x \notin A$, then the former case will obtain and if $x \in A$, then the latter case will obtain, so that $\Phi$ is total and is a retracing function for $A$. It follows from the definition that for every $x$, $\Phi(x) \leq x$ and there are no elements of $A$ between $\Phi(x)$ and $x$. Now for any $y$, let $a$ be the least such that $a > y$ and $a \in A$. Then $\Phi(x) = y$ implies that $x \leq a$, so that $\{x : \Phi(x) = y\}$ is finite, as desired.

$(b) \to (c)$: Let $\Phi$ be given as described. Then we define $\Psi(x)$ to be length $n$ of the chain $x > \Phi(x) > \Phi(\Phi(x)) > \cdots > \Phi^n(x) = a_0$, if there is such an $n$-chain, and $\Psi(x) = x$ if $\Phi^{i+1}(x) = \Phi^i(x)$ for some $i$. Thus for $a_n \in A$, we obtain $\Psi(a_n) = n$. To complete the proof, we show by induction that, for each $n$, there are only finitely many $n$-chains $x > \Phi(x) > \cdots > \Phi^n(x) = a_0$ of length $n$. For $n = 1$, this follows from the assumption that $\Phi(x) = a_0$ for only finitely many $x$. Suppose now that there are only finitely many such $n$-chains of length $n$. Then any $n+1$-chain must extend one of these and, by our assumption, there are only finitely many ways to extend each chain. Thus there can be only finitely many $n+1$-chains.

$(c) \to (a)$: Let $\Psi$ be given as described. Then for $a = a_{n+1} \in A$, $\Psi(a) = n+1$ and the retracing function $\phi(a_{n+1}) = a_n$ may now be computed by searching for the least $s$ such that exactly $n+1$ elements of $A_s$ are less than $a$ and taking $a_n$ to be the largest of those. □

We say that an infinite set $A = \{a_0 < a_1 < \cdots\}$ is *second-retraceable* if there is a (total) computable function $\Phi$ such that, for any $m < n$, $\Phi(a_m, a_n) = m$. In general, $A$ is $k$-retraceable if there is a computable $\Phi$ such that $\Phi(a_{m_1}, a_{m_2}, \ldots, a_{m_k}) = m_1$ for any $m_1 < m_2 < \cdots < m_k$. Of course, any $k+1$-retraceable set is also $k$-retraceable.

A subset $F$ of the set $\{a_0 < a_1 < \cdots\}$ is said to be an *initial subset* of $A$ if $a_{n+1} \in F$ implies $a_n \in F$ for all $n$. Thus the initial subsets of $A$ are $A$ together with the finite sets $\{a_0, \ldots, a_{n-1}\}$ for each $n$. Let $I_1(A)$ denote the class of initial subsets of $A$. In general, the $k$-initial subsets $I_k(A)$ are the subsets $F$ of $A$ such that for any elements $a < b_1 < b_2 < \cdots b_k$ of $A$, if $b_1, \ldots, b_k \in F$, then $a \in F$.

**Theorem 2.6.2.** *For each finite $k$, the set $A = \{a_0 < a_1 < \ldots\}$ is $\Pi^0_1$ and $k$-retraceable if and only if the class $I_k(A)$ of $k$-initial subsets of $A$ is a $\Pi^0_1$ class.*

*Proof.* Suppose first that $A$ is $k$-retraceable via the function $\Phi$ and that $A$ is a $\Pi^0_1$ set. Let $A_s$ denote the computable approximation to the set $A$ at stage $s$, so that $A = \cap_s A_s$. Now define the computable tree $T$ as follows.

$\lfloor b_0, b_1, \ldots, b_n, s \rfloor \in T \iff$

1. $(\forall i \leq n)(b_i \in A^s)$ and

2. if $n \geq k$, then $\Phi(b_{n-k+1}, b_{n-k+2}, \ldots, b_{n-1}, b_n) = n - k + 1$.

It is easy to check that $[T] = I_k(A)$, so that $I_k(A)$ is a $\Pi^0_1$ class.

Now suppose that $I_k(A)$ is a $\Pi^0_1$ class and let $T$ be a computable tree so that $I_k(A) = [T]$. We will explain how to compute a $k$-retracing function $\Phi$. Given $b_1 = a_{m_1} < b_2 = a_{m_2} < \ldots < b_k = a_{m_k}$, observe that there is only one possible string $\sigma = \lfloor a_0, a_1, \ldots, a_{m_1 - 1}, b_1, b_2, \ldots, b_k \rfloor^\frown 1$ of the form $\lfloor c_0, c_1, \ldots, c_r, b_1, b_2, \ldots, b_k \rfloor^\frown 1$ which has an extension in $T$; $\Phi(b_1, \ldots, b_k) = m_1$ is then easily computed from $\sigma$. To find $\sigma$, we just search through all strings of length $m > b_k$ until we find $m$ large enough so that all strings $\tau$ in $T$ of length $m$ and with $\tau \lceil b_k + 1$ of the desired form, start with the same initial segment ($\sigma$) of length $b_k + 1$.

To see that $A$ is a $\Pi^0_1$ set, recall that $Ext(T)$ is $\Pi^0_1$ and observe that

$a \in A \iff (\exists \sigma)[|\sigma| = a + 1 \& \sigma \in Ext(T) \& \sigma(a) = 1]$. □

We can now give a quick proof that any retraceable non-computable $\Pi^0_1$ set is hyperimmune.

**Theorem 2.6.3.** *[Dekker-Myhill] If $A = \{a_0 < a_1 < \cdots\}$ is a retraceable non-computable $\Pi^0_1$ set, then $A$ is hyperimmune.*

*Proof.* By Theorem 2.6.2, $P(A)$ is a $\Pi^0_1$ class. Now suppose by way of contradiction that $f$ were a computable function which dominated $p_A$, that is, $f(n) > p_A(n)$ for all $n$. Then the set $\{A\}$ would be the intersection of $I(A)$ with the following $\Pi^0_1$ class:

$\{B : (\forall n)(card(B \cap \{0, 1, \ldots, f(n)\}) \geq n\}$.

Thus $\{A\}$ would be a $\Pi^0_1$ class, so that $A$ would be computable (this is seen below in Exercise 4.1.11. This contradiction now demonstrates the result. □

For any $k$-retraceable $\Pi^0_1$ set $A$, the $\Pi^0_1$ class $I_k(A)$ is provides an example of a class with C-B rank $k$.

**Theorem 2.6.4.** *For any set $A$, $D^k(I_k(A)) = \{A\}$.*

*Proof.* It is easy to see that $D(I(A)) = \{A\}$ and that, for each $k$, $D(I_{k+1}(A)) = I_k(A)$. □

It follows that if $A$ is $k$-retraceable, then $A$ has rank $k$ in $I_k(P)$ and thus has rank $\leq k$.

We next give a result which shows how to define a retraceable $\Pi^0_1$ set by $\Pi^0_1$-recursion.

**Theorem 2.6.5.** *Suppose that the set $A = \{a_0 < a_1 < \ldots\}$ is defined recursively by a $\Pi^0_1$ relation $Q(x, y)$ such that, for all $n$ and $x$, $x = a_n \iff Q(x, < a_0, \ldots, a_{n-1} >)$. Then $A$ is a $\Pi^0_1$ set and is retraceable.*

*Proof.* Define the $\Pi^0_1$ relation $R(n,x)$ by

$R(n,x) \iff (\exists x_0 < \cdots < x_{n-1} < x_n = x)(\forall i < n)Q(x_i, < x_0, \ldots, x_{i-1} >)$.

Then the set A is $\Pi^0_1$ since

$a \in A \iff (\exists n \leq a)R(n,a)$.

Define the uniformly computable relation $R_s(n,x)$ as in the definition of $R$ above with $Q_s$ in place of $Q$.

The counting function $\Psi$ such that $\Psi(a_n) = n$ may of course be defined by the fact that $n$ is the unique $y$ such that $R(y, a_n)$. Since, given $a \in A$, there is just one $n \leq a$ such that $a = a_n$, $\Psi(a) = n$ may be computed by searching for an $s$ large enough so that $R_s(n,a)$ for only one number $n \leq a$. □

This result can now be applied to give a quick proof of the following theorem of Dekker and Myhill [51] (Theorem T3).

**Theorem 2.6.6.** *[Dekker-Myhill] Every r.e. set $B$ is Turing equivalent to a retraceable $\Pi^0_1$ set $A$.*

*Proof.* Let the c.e. set B be the union of uniformly computable sets $B_s$ and define the set $A$ by $\Pi^0_1$ recursion as follows. There are two cases in the definition of $a_n$. If $n \notin B$, then $a_n = a_{n-1} + 1$ and if $n \in B$, then $a_n$ is the least $s > a_{n-1}$ such that $n \in B_s$. It is clear that this is a $\Pi^0_1$-recursion, so that $A$ is a $\Pi^0_1$ retraceable set. The definition also shows that $A$ is computable in $B$. On the other hand, for any $n$, we have $n \in B \iff n \in B_{a_n}$, so that $B$ is computable in $A$. □

**Definition 2.6.7.** *The Cantor-Bendixson (C-B) rank of a set $A$ is the least ordinal $\alpha$ such that $A$ has rank $\alpha$ in some $\Pi^0_1$ class $P \subset \{0,1\}^{\mathbb{N}}$.*

It follows from Theorems 2.6.4 and 2.6.6 that every non-zero r.e. degree contains a set $A$ of C-B rank one. A slightly better result was obtained in [22] by Cenzer, Downey, Jockusch and Shore, hereafter abbreviated as C-D-J-S.

**Theorem 2.6.8.** *[C-D-J-S] Every c.e. non-computable set $B$ is Turing equivalent to a hypersimple c.e. set $E$ of rank one; furthermore there is a computable tree $U$ with no dead ends such that $D([U]) = \{E\}$.*

*Proof.* Let $A = a_0 < a_1 < \cdots$ be the $\Pi^0_1$ retraceable set defined in Theorem 2.6.6 and let $A$ be the intersection of the uniformly computable, decreasing sequence $A_s$. Define the computable tree $S$ to be a slight extension of the tree $T$ defined in Theorem 2.6.2. That is, we let $\Phi$ be the retracing function given by Theorem 2.6.1 for $A$ so that $\Phi(a_{n+1}) = a_n$ and so that $\{x : \Phi(x) = y\}$ is finite for each $y$, and define

$\lfloor c_0, \ldots, c_n, c_{n+1} \rfloor \in S \iff c_0 = a_0 \;\&\; (\forall i \leq n)[c_i \in A_{c_n} \;\&\; (i > 0 \to \Phi(c_i) = c_{i-1})]$.

$S$ has no dead ends because for any string $\sigma \in S$, it is clear that $\sigma^\frown 0 \in S$.

We leave it to the reader to check that $D(P) = \{A\}$.

To obtain the c.e. set $E$, we note that the complement function $F(C) = \mathbb{N} \setminus C$ is a computable homeomorphism of $\{0,1\}^\omega$ to itself, so that the c.e. set $E = \mathbb{N} \setminus A$

has rank one in the $\Pi^0_1$ class of complements $\{F(C) : C \in P\}$. Finally, any retraceable noncomputable $\Pi^0_1$ set is hyperimmune by Theorem 2.6.3, so that $E$ is hypersimple. □

**Exercises**

2.6.1. Verify that $D(I(A)) = \{A\}$ and that, for each $k$, $D(I_{k+1}(A)) = I_k(A)$.

2.6.2. Show that $D(P) = \{A\}$, in the proof of Theorem 2.6.8.

2.6.3. For any computable function $f$, the set $K(f) = \{x : (\forall n)x(n) \leq f(n)\}$ is a c. b. $\Pi^0_1$ class. Show that $K(p_A)$ is also a $\Pi^0_1$ class, where $A$ is a infinite $\Pi^0_1$ set and $p_A$ is the principal function defined above.

2.6.4. Show that for the total retracing function $\Phi$ of Theorem 2.6.1, if $g(x) = card(\{y : \Phi(y) = x\})$ is computable, then $A$ is a computable set. However, show that for the retraceing function from Theorem 2.6.6, there is always a computable upper bound for the cardinality.

## 2.7 Reducibility

In this section, we consider some connections between subclasses of $\{0,1\}^{\mathbb{N}}$, computably bounded $\Pi^0_1$ classes, bounded classes and $\Pi^0_n$ classes. Our goal is to reduce every class to a class of sets.

Computably bounded $\Pi^0_1$ classes play a fundamental role and occur frequently in the applications. A very useful result which simplifies the theory of c. b. $\Pi^0_1$ classes is that every such class is computably homeomorphic to a subclass of $\{0,1\}^{\mathbb{N}}$.

**Definition 2.7.1.** *Classes $P$ and $Q$ are computably homeomorphic if there is a (total) computably continuous functional $F : \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ such that $F$ maps $P$ one-to-one and onto $Q$.*

Notice that we do not require that $F$ be one-to-one onto $\mathbb{N}^{\mathbb{N}}$ or that it map $\mathbb{N}^{\mathbb{N}}$ onto $\mathbb{N}^{\mathbb{N}}$.

**Lemma 2.7.2.** *If there is a partial computable function $\Phi$ mapping a subset of $\mathbb{N}^{\mathbb{N}}$ to a subset of $\mathbb{N}^{\mathbb{N}}$ such that $\Phi$ is total on $P$ and maps $P$ one-to-one and onto $Q$, then $P$ and $Q$ are computably homeomorphic.*

*Proof.* Let $\Phi$ have a representation $\phi : \mathbb{N}^* \to \mathbb{N}^*$; that is, let $\phi(\sigma)$ be the longest sequence of the form $(\Phi(0,\sigma), \Phi(1,\sigma), \ldots, \Phi(n-1,\sigma))$. Let $P = [T]$ and $Q = [S]$ for computable trees tree $S$ and $T$, Define a new mapping $f : \mathbb{N}^* \to \mathbb{N}^*$ by letting $f(\sigma) = \phi(\sigma)$ for $\sigma \in T$ and letting $f(\sigma^\frown n) = f(\sigma)^\frown n$ if $\sigma^\frown n \notin T$. Then $f$ represents a computably continuous function $F$ such that $F(x) = \Phi(x)$ for $x \in P$ and $F(x) = \phi(x \restriction k)^\frown(x(k), x(k+1), \ldots)$ if $x \notin P$ and $k$ is the least such that $x \restriction k+1 \notin T$. □

**Theorem 2.7.3.** *Any c. b. $\Pi^0_1$ class $P$ is computably homeomorphic to a $\Pi^0_1$ class $Q$ of sets.*

*Proof.* Let $T$ be a highly computable tree such that $P = [T]$ and let $h$ be a computable function such that $\sigma(n) < h(n)$ for all $\sigma \in T$ and all $n < |\sigma|$.

The homeomorphism $\Phi$ will be defined by

$$\Phi(x) = 0^{x(0)}10^{x(1)}\ldots$$

and the class $Q \subset \{0,1\}^{\mathbb{N}}$ is simply $\{\Phi(x) : x \in P\}$.

The functional $\Phi$ is clearly one-to-one and maps $P$ onto $Q$. $\Phi$ is computably continuous since it is represented by the computable map taking $\sigma = (\sigma(0), \ldots, \sigma(n-1))$ to $0^{\sigma(0)}1\ldots 0^{\sigma(n-1)}$. $Q = \Phi[P]$ is a $\Pi^0_1$ class by Theorem 2.4.8.

More specifically, $Q = [S]$, where

$$0^{x(0)}10^{x(1)}\ldots 0^{x(k-1)}10^i \in S \iff (x(0), \ldots, x(k-1)) \in T \;\&\; i < h(k)\}.$$

□

This result can be relativized to an oracle. Also, we can apply the same argument to $\Pi^0_1$ classes which are not computably bounded.

**Theorem 2.7.4.** *(a) Any strong $\Pi^0_2$ class $P$ which is highly computable in $\mathbf{0}'$ is computably homeomorphic to a strong $\Pi^0_2$ class $Q$ of sets.*

*(b) For any $\Pi^0_1$ class $P$, there exists a $\Pi^0_1$ class $Q \subset \{0,1\}^{\mathbb{N}}$ and a one-to-one degree-preserving correspondence between the non-computable members of $P$ and the non-computable members of $Q$.*

*Proof.* (a) This is just the relativization of Theorem 2.7.3 to the oracle $\mathbf{0}'$.

(b) Consider the representation of the mapping from Theorem 2.7.3. We can use the same mapping as above, but in this case Theorem 2.4.8 only tells us that the image is a strong $\Sigma^1_1$ class. If we look at the definition of the tree $S$ such that $Q = [S]$, we have to remove the condition $i < h(k)$. This potentially introduces computable elements $0^{x(0)}10^{x(1)}\ldots 0^{x(k-1)}0^{\omega}$ into $Q$. However, any non-computable element has infinitely many 1's and hence will be the image of an element of $P$. □

If the same technique is applied to a bounded $\Pi^0_1$ class, we get one half of the following result from [86].

**Theorem 2.7.5.** *[Jockusch-Lewis-Remmel]*

*(a) Any bounded $\Pi^0_1$ class $P$ is computably homeomorphic to a strong $\Pi^0_2$ class $Q$ of sets.*

(b) *For any strong $\Pi^0_2$ class $P$ which is highly computable in $\mathbf{0}'$, there is a bounded $\Pi^0_1$ class $Q$ and an effective one-to-one degree-preserving correspondence between $P$ and $Q$.*

*Proof.* (a) This is left as an exercise.

(b) Let $P = [T]$, where $S$ is highly computable in $0'$. By Theorem 2.7.4, we may assume that $T$ is a binary tree. It now follows from Proposition 2.3.7 that $T$ may be assumed to be a $\Sigma^0_1$ tree. Thus there is a computable relation $R \subset \mathbb{N} \times \{0,1\}^*$ such that

$$x \in P \iff (\forall m)(\exists n)R(n, x \upharpoonright m).$$

Now we may define $Q$ by

$$z = x \oplus y \in Q \iff (\forall m)[R(y(m), x \upharpoonright m) \ \& \ \forall i < y(m) \neg R(i, x \upharpoonright m)].$$

Then for each $x \oplus y \in Q$, we have $x \in P$ and for each $x \in P$, there is a unique $y$ such that $x \oplus y \in Q$ and that $y$ is defined so that $y(m)$ is the least $n$ such that $R(n, x \upharpoonright m)$. Thus $y$ is computable in $x$ and therefore $x \oplus y$ has the same degree as $x$. □

The proof of part (b) can be modified for an arbitrary $\Pi^0_2$ class to give a theorem from [87]. The proof is left as an exercise.

**Theorem 2.7.6** (Jockusch-McLaughlin)**.** *For any $\Pi^0_2$ class $P$, there is a $\Pi^0_1$ class $Q$ and an effective one-to-one degree-preserving correspondence between $P$ and $Q$.* □

Jockusch and Soare showed in Theorem 1 of [90] that an arbitrary $\Pi^0_1$ class $P$ with no computable members can be represented by a c. b. $\Pi^0_1$ class $Q$ in the sense that the degrees of members of $P$ are a subset of the degrees of members of $Q$. We give this result together with a relativized version. Let $\mathcal{D}(P)$ denote the set of degrees of members of the class $P$.

**Theorem 2.7.7.** *(a) For any $\Pi^0_1$ class $P \subset \mathbb{N}^{\mathbb{N}}$, there is a $\Pi^0_1$ class $R$ of sets such that (1) $\mathcal{D}(P) \subset \mathcal{D}(R)$ and (2) there is a one-to-one correspondence between the computable members of $P$ and the computable members of $R$. (So that $R$ has no computable members if $P$ has no computable members.) Furthermore, there is a primitive recursive function $k$ such that for $P = P_e$, we have $R = P_{k(e)}$.*

(b) *For any $\Pi^0_1$ class $P \subset \mathbb{N}^{\mathbb{N}}$ with no members computable in $\mathbf{0}'$, there is a strong $\Pi^0_2$ class $R$ of sets with no members computable in $\mathbf{0}'$ such that $\mathcal{D}(P) \subset \mathcal{D}(R)$. Furthermore, there is a primitive recursive function $h$ such that for $P = P_e$, we have $R = P^2_{h(e)}$.*

*Proof.* (a) Let $Q$ be a $\Pi^0_1$ class of sets with no computable member. Let $P = [S]$, let $Q = [T]$ and assume without loss of generality that $S \subset (\mathbb{N} \setminus \{0,1\})^*$.

It suffices, by Theorem 2.7.3 to obtain a class $R$ which is computably bounded and otherwise meets the requirements of the conclusion. Define the tree $U$ to be the set of strings

$$mu = \sigma_1 * \tau_1 * \sigma_2 * \tau_2 * \cdots * \sigma_n * \tau_n$$

such that $\sigma_i \neq \emptyset$ for $i > 1$ and $\tau_j \neq \emptyset$ for $j < n$, such that $s(\mu) = \sigma_1 * \cdots * \sigma_n \in S$ and $\tau_j \in T$ for all $j$, and such that $\mu(k) \leq k+1$ for all $k$.

We claim that $R = [U]$ satisfies the requirements of the theorem. Note first that the construction is uniformly computable in the tree $T$, so that there is a primitive recursive function $k$ such that the for $T = T_e$, the tree $U = T_{k(e)}$.

The tree $U$ is finite-branching by the restriction that $\mu(k) \leq k+1$. Thus $R$ is a computably bounded $\Pi^0_1$ class. Now for any $x \in P$ we can define $z \in R$ with the same degree as $x$ as follows. First of all, define a computable sequence $\emptyset = t_0, t_1, \ldots$ such that $t_j$ is the lexicographically least string in $T$ of length $j$. Now given $x$, let

$$z_x = t_{i_0} * (x(0)) * t_{i_1} * x(1)) \cdots ,$$

where for each $n$, $i_n$ is the least such that

$$x(n) \leq |t_{i_0} * (x(0)) * \cdots * t_{i_n}| + 1.$$

Then $z_x$ is computable in $x$ by the definition, and $x$ is computable in $z_x$, since it is the subsequence of $z_x$ consisting of the entries $z_x(n) > 1$.

Now let $z$ be any element of $R$ which is not of the form $z_x$ for any $x$. There are two cases.

(Case 1): Suppose that $z(i) > 1$ for infinitely many $i$ and let $i_0, i_1, \ldots$ enumerate $\{i : z(i) > 1\}$. Define $x$ by $x(n) = z(i_n)$. Then $x$ is computable in $z$ and that $x \in P$, so that $x$ is not computable and therefore $z$ is not computable.

(Case 2): Suppose that $z(i) > 1$ for only finitely many $i$ and let $m$ be the largest such that $z(m) > 1$. Define $y$ by $y(n) = z(m+n)$. Then $y$ is computable in $z$ and that $y \in Q$, so that $y$ is not computable and therefore $z$ is not computable.

(b) The proof is just a modification of the proof of (a). Let $Q = [T]$ in this case be a strong $\Pi^0_2$ class of sets with no member computable in $\mathbf{0}'$. Then we define a tree $U$ computable in $\mathbf{0}'$ with $\mu(k) \leq k+1$ as above so that $R = [U]$ is a c. b. strong $\Pi^0_2$ class with the desired properties and apply Theorem 2.7.5(a). $\square$

We will consider members of $\Pi^0_1$ classes in detail in Chapter 3.

**Exercises**

2.7.1. Show that there exist $\Pi^0_1$ classes $P$ and $Q$ which are computably homeomorphic, but for which there can be no homeomorphism $F$ of $\mathbb{N}^{\mathbb{N}}$ onto itself which maps $P$ one-to-one and onto $Q$.

2.7.2. Show that if $P = [T]$ where $T$ is a finite-branching, $\Sigma^0_1$ tree, then $P$ is highly computable in $\mathbf{0}'$.

2.7.3. Show that there is a computably continuous map on $\mathbb{N}^{\mathbb{N}}$ such that the image is not even a closed set.

## 2.8 Thin and minimal classes

A $\Pi^0_1$ class $P$ is said to be *thin* if, for every $\Pi^0_1$ subclass $Q$ of $P$, there is a clopen set $U$ such that $Q = U \cap P$. An infinite $\Pi^0_1$ class $C$ is said to be *minimal* if every $\Pi^0_1$ subclass $Q$ of $C$ is either finite or cofinite in $C$. Thus the notion of a minimal $\Pi^0_1$ class is the analog of the notion of a co-maximal $\Pi^0_1$ subset of $\omega$. In particular, if $C$ is a co-maximal set, then the class of subsets of $C$ containing either one or no elements is an example of a minimal $\Pi^0_1$ class which is not thin. (See the exercises.)

We have seen that any isolated element of a computably bounded $\Pi^0_1$ class must be computable. For a thin $\Pi^0_1$ class, the converse also holds. (Exercise 2). It follows that a perfect thin class has no computable members.

The first construction of a thin $\Pi^0_1$ class is due to Martin and Pour-El [131].

**Theorem 2.8.1.** *(Martin–Pour-El) There exists a perfect thin $\Pi^0_1$ class with no computable member.*

*Proof.* Let $P_e = [T_e]$ be the $e$'th $\Pi^0_1$ class as in Theorem 2.3.3 and let $\phi_e$ be the $e$'th partial recursive function from $\omega$ into $\{0,1\}$. We will construct a recursive tree $S$ with corresponding $\Pi^0_1$ class $P = [S]$ and a homeomorphism $F$ from $\{0,1\}^\omega$ onto $P$. $F$ will be constructed by means of a map $f : \{0,1\}^{<\omega} \to S$ such that $\sigma \prec \tau \iff f(\sigma) \prec f(\tau)$; then for $x \in \{0,1\}^\omega$, $F(x) = \cup_n f(x\lceil n)$.

To ensure that $P$ is thin, we construct $f$ to satisfy the following requirement for each $e$.

$R_e$: For each $\sigma \in \{0,1\}^{e+1}$, if $f(\sigma) \in T_e$, then $(\forall \tau)(\sigma \prec \tau \to f(\tau) \in T_e)$.

To see that this makes $P$ thin, let $U = \cup\{I(f(\sigma) : |\sigma| = e+1 \ \& \ f(\sigma) \in T_e\}$ and observe that if $P_e \subset P$, then $P_e = P \cap U$.

The map $f$ is defined in uniformly computable stages $f_s$, beginning with $f_0$ as the identity function.

(Stage $s+1$): Look for $e < s$ and $\sigma \in \{0,1\}^{e+1}$ and $\tau \succ \sigma$ with $|\tau \leq s+1$ such that $f_s(\sigma) \in T_e$, but $f_s(\tau) \notin T_e$. If such $e$, $\sigma$ and $\tau$ exist, then we take the least such $e$ and the lexicographically least $\sigma$ and $\tau$ for that $e$. Then we let $f_{s+1}(\sigma) = f_s(\tau)$ and in general, for any $\rho$ we let

$f_{s+1}(\sigma^\frown\rho) = f_s(\tau^\frown\rho)$ and

$f_{s+1}(\rho) = f_s(\rho)$ for $\rho$ incomparable with $\sigma$.

If no such $e$, $\sigma$ and $\tau$ exist, then we just let $f_{s+1} = f_s$.

It is easy to see by induction on $|\sigma|$ that for each $\sigma$, $f_s(\sigma)$ converges to a limit $f(\sigma)$. Then we see by induction on $e$ that the requirements $R_e$ are satisfied. $\square$

Countable thin classes were studied in [22].

The connection between thin and minimal classes is given by the following.

**Theorem 2.8.2.** *(C-D-J-S) The following are equivalent for any $\Pi^0_1$ class $P$.*

*(a) $P$ is thin and $D(P)$ is a singleton.*

*(b) $P$ is minimal and has a non-computable member.*

*Proof.* (a) $\rightarrow$ (b): Suppose that $P$ is thin and that $D(P) = \{A\}$. Then $A$ is non-computable by Exercise 2. Let $Q$ be a $\Pi^0_1$ class such that $Q \subset P$. Then $Q = U \cap P$ for some clopen $U$, so that $P \setminus Q$ is also a $\Pi^0_1$ class. If both $Q$ and $P \setminus Q$ were infinite, then both would contain limit points, contradicting the assumption that $P$ has only one limit point.

(b) $\rightarrow$ (a): Suppose that $P$ is minimal and has a nonrecursive member $A$. Then $A \in D(P)$ by Corollary 2.2.16. For any $B \neq A$ in $P$, let $U$ be an interval such that $A \in U$ and $B \notin U$. Then $U \cap P$ is infinite, and therefore $P \setminus U$ must be finite, which implies that $B \notin D(P)$. Therefore $D(P) = \{A\}$. Now let $Q$ be any $\Pi^0_1$ subclass of $P$. For any $B \neq A$ in $P$, let $U(B)$ be an interval such that $U(B) \cap P = \{B\}$. Since $P$ is minimal, there are two cases.

Case 1: $Q$ is finite. Then $Q = P \cap \cup_{B \in Q} U(B)$.

Case 2: $P \setminus Q$ is finite. Then $Q = P \cap [2^\omega \setminus \bigcup_{B \in P \setminus Q} U(B)]$. $\square$

We next construct a minimal, thin class.

**Theorem 2.8.3** ([22])**.** *There exists a minimal, thin $\Pi^0_1$ class $P$; furthermore, $P$ is decidable.*

*Proof.* Let $P_e = [T_e]$ be the $e$'th $\Pi^0_1$ class as above. We will construct a set $A$, a sequence $\tau_0 \prec \tau_1 \prec \dots$ of strings with $A = \cup_i \tau_i$ and a $\Pi^0_1$ class $P$ such that

(1) $D(P) = \{A\}$.

(2) For any $e$ and any extension $B \in P$ of $\tau_e$, if $A \in [T_e]$, then $B \in [T_e]$.

Properties (1) and (2) imply that $P$ is minimal, by the following argument.

Note first that, for all $B \in P$, if $B \neq A$, then the set $B$ is isolated in $P$ by property (1), so that there exists a clopen set $U(B)$ such that $P \cap U(B) = \{B\}$. Suppose now that $[T_e]$ is a subset of $P$. Then there are two cases.

(Case 1) If $A \notin P_e$, then, since $A$ is the only limit point of $P$ and every infinite class has a limit point, it follows that $Pe$ is finite.

(Case 2) If $A \in P_e$, then it follows from property (2) that every extension of $\tau_e$ is also in $T_e$. Now the set $P \setminus I(\tau_e)$ of paths through $T$ which are not extensions of $\tau_e$ is a closed set and has no limit point (since $A$ is the only limit point of $P$). Thus $P \setminus I(\tau_e)$ is finite and, since $P \setminus P_e \subset P \setminus I(\tau_e)$, $P \setminus [T_e]$ is also finite.

It also follows from properties (1) and (2) that $A$ is not computable. To see this, suppose by way of contradiction that $A$ were computable. Then $\{A\}$ would be a $\Pi^0_1$ class, so that $\{A\} = P_e$ for some $e$. Now by property (2), we have $P \cap I(\tau_e) \subset P_e$, which makes $A$ isolated in $P$, contradicting property (1). It now follows from Theorem 2.8.2 that $P$ is thin.

It remains to construct the set $P$. The construction will proceed in stages. At stage $s$ we will have, for $e \leq s$, finite sequences $\tau^s_e$ such that, for all $e < s$, $\tau^{s\frown}_e 1 \prec \tau^s_{e+1}$. The construction will ensure the existence of the limits $\tau_e =$

$\lim_s \tau_e^s$ for each $e$. The point $A$ will the union of $\{\tau_e : e \in \omega\}$. At the same time we will be defining a sequence $k(0) < k(1) < \cdots$ so that $s \le k(s)$ and constructing a computable tree $T$ in stages $T^s$. At stage $s$, we will have decided whether each finite sequence of length $k(s)$ is in $T$. This will ensure that $T$ is computable. We will always put $\sigma^\frown 0$ into $T$ whenever $\sigma$ is in $T$. This will imply that $x_e = \tau_e^\frown 0^\omega \in P$ for all $e$; since $A$ is non-computable and therefore infinite, there are infinitely many distinct $x_e$, so that $A \in D(P)$. This also implies that $P$ is decidable, that is, $T$ has no dead ends. To obtain $D(P) = \{A\}$, we do the construction so that, whenever $\tau_e^{s+1} = \tau_e^s$, then there are no new branches added below $\tau_e^s$. Thus once we have reached a stage $s$ such that $\tau_e^s = \tau_e$ and counted the number $n$ of distinct branches of $T^s$ not passing through $\tau_e^s$, then we know that all but $n$ points of $P$ will pass through $\tau_e$. Now suppose that some path $B$ is in $D(P)$ but is different from $A$. Just let $k$ be the least number such that $A(k-1) \neq B(k-1)$ and let $e$ be least such that $A\lceil k \subset \tau_e$. Then no extension of $B\lceil k$ passes through $\tau_e$. It follows that the set of extensions of $B\lceil k$ in $P$ is finite, so that $B$ is isolated in $P$. This will take care of property (1).

In order to satisfy property (2), we want the construction to ensure the following requirements for each $e$.

($R_e$): If $\tau_e \in T_e$, then every extension of $\tau_e$ which is in $T$ is also in $T_e$.

We begin the construction by setting $k(0) = 1$, putting (0) and (1) in $T^0$ and setting $\tau_0^0 = \emptyset$.

Now suppose we have completed the construction as far as stage $s$. At stage $s+1$, we look for the least number $e \le s$ such that $\tau_e^s \in T^s \cap T_e$ but $\tau_e^s$ has some extension $\tau \in T^s$ which is not in $T_e$. If such an $e$ exists, then we act on requirement $R_e$ at stage $s+1$, as follows. Let $\tau$ be the lexicographically least extension of $\tau_e^s$ of length $k(s)$ which is in $T^s \setminus T_e$. Then let $\tau_e^{s+1} = \tau$. For $i < e$, let $\tau_i^{s+1} = \tau_i^s$. For $i \le s-e+1$, let $\tau_{e+i}^{s+1} = \tau^\frown 1^i$. Now let $k(s+1) = k(s)+s-e+1$ and define $T^{s+1}$ to be the union of $T^s$ with the set of the following strings. First, for any $\sigma \in T^s$ of length $k(s)$ and any $i \le s-e+1$, the extension $\sigma^\frown 0^i$. Next, for any $i \le s-e+1$, and any $j \le s-e+1-i$, the extension $\tau^\frown(1^i)^\frown(0^j)$.

If there is no such $e$, just let $\tau_i^{s+1} = \tau_i^s$ for all $i \le s$ and let $\tau_{s+1}^{s+1} = \tau_s^{s\frown} 1$. Let $k(s+1) = k(s)+1$ and let $T^{s+1}$ be the union of $T^s$ with the set of all strings $\sigma^\frown 0$ where $\sigma \in T^s$ and the string $\tau_{s+1}^{s+1}$.

Observe that in either case, we have extended all nodes in $T^s$ by at least one node in $T^{s+1}$, so that $T$ will have no dead ends.

**Claim 1**: For every $e$, the sequence $\tau_e^s$ converges to some limit $\tau_e$.

Proof of Claim 1: This is by induction on $e$. Suppose therefore that Claim 1 is proved for all $i < e$ and that we have reached a stage $s$ such that $\tau_i^s = \tau_i$ for all $i < e$. There are two cases. If $\tau_e^r = \tau_e^s$ for all $r > s$, then the limit $\tau_e = \tau_e^s$ and we are done. Otherwise, let $r > s$ be least such that $\tau_e^r \neq \tau_e^s$. It follows from the construction that we must have $\tau_e^r \notin T_e$. After stage $r$, there is no way that $\tau_e^t$ can be different from $\tau_e^r$. Thus the limit $\tau_e = \tau_e^r$. □

Since $\tau_e^s \prec \tau_{e+1}^s$ for all $s$ and $e$, it follows that $\tau_e \prec \tau_{e+1}$ for all $e$. Thus we can define the set $A$ to have characteristic function $\cup_e \tau_e$.

**Claim 2**: For any e and any s, if $\tau_e^{s+1} = \tau_e^s$, then there are no new branches in $T^{s+1} \setminus T^s$ which do not pass through $\tau_e^s$.

Claim 2 is immediate from the construction. $\square$

It now follows that, for any $e$, all but finitely many points of $P$ pass through $\tau_e$. It follows from the discussion preceding the construction that $D(P) = \{A\}$.

**Claim 3** If $\tau_e \in T_e$, then every extension of $\tau_e$ which is in T is also in $T_e$.

Proof of Claim 3: Suppose by way of contradiction that $\tau_e \in T_e$ but that $\tau_e$ has some extension $\tau \in T$ such that $\tau \notin T_e$. Consider a stage $s > lh(\tau)$ such that $\tau_i^s = \tau_i$ for all $i \leq e$ and $\tau \in T^s$. Then at stage $s+1$, we have $\tau \in T^s$ so the construction dictates that we act on requirement $R_e$ and make $\tau_e^{s+1} = \tau$, contradicting the assumption that $\tau_e^s = \tau_e$. $\square$

This establishes property (1) and (2) above and thus completes the proof $\square$

We close the section with two important properties of thin classes.

**Lemma 2.8.4.** *Let $Q \subseteq \{0,1\}^{\mathbb{N}}$ be a thin $\Pi^0_1$ class. Then*

1. *Every $\Pi^0_1$ $P \subseteq Q$ is thin;*
2. *For any computable $\Phi : \{0,1\}^{\mathbb{N}} \to \{0,1\}^{\mathbb{N}}$, $\Phi[Q]$ is thin.*

*Proof.* (1) is trivial. For (2), suppose that $R \subseteq \Phi[Q]$. Then $\Phi^{-1}[R]$ is a $\Pi^0_1$ subclass of $Q$ and hence $\Phi^{-1}[R] = U \cap Q$ for some clopen $U$. Thus $R = \Phi[U] \cap Q$ and $\Phi[U]$ is clopen since $\{0,1\}^{\mathbb{N}}$ is compact. $\square$

**Exercises**

2.8.1. Show that for any maximal c.e. set $A$, the class containing the empty set together with all singletons $\{m\}$ where $m \notin A$, is an example of a minimal $\Pi^0_1$ class which is not thin.

2.8.2. Show that any computable element of a thin $\Pi^0_1$ class must be isolated.

## 2.9 Mathematical Logic

In this section, we set up the framework for the representation and application of $\Pi^0_1$ classes, using the area of logical theories. There is a very close connection between $\Pi^0_1$ classes and logical theories and we will return to this topic in later sections as we develop the theory of $\Pi^0_1$ classes.

Recall the arbitrary first-order effective language $\mathcal{L}$ described in Chapter 1. Let $Sent(\mathcal{L})$ be the set of sentences of $\mathcal{L}$. For any subset $\Gamma$ of $Sent(\mathcal{L})$, the set $Con(\Gamma)$ of consequences of $\Gamma$ is the closure of $\Gamma$ under logical deduction and the set $Ref(\Gamma)$ of refutations of $\Gamma$ is the set of negations of the consequences of $\Gamma$. A subset $\Gamma$ of $Sent(\mathcal{L})$ is a first order logical theory if $\Gamma$ is closed under logical deduction. $\Sigma$ is said to be a set of axioms for $\Gamma$ if $\Gamma = Con(\Sigma)$ and $\Gamma$

is (computably) *axiomatizable* if $\Gamma$ has a computable set of axioms; the modifier (computably) will normally be omitted. It is not hard to see that $\Gamma$ is axiomatizable if and only if $\Gamma$ is computably enumerable. A theory is said to be *decidable* if it is computable. It follows from Post's Theorem that a complete axiomatizable theory is decidable.

The usual idea for the application of $\Pi^0_1$ classes is that the set of solutions to some computable problem should correspond to a $\Pi^0_1$ class. The problem here is to find a complete consistent extension of a given computable or axiomatizable theory. The classical result here is the Completeness Theorem of Gödel that any consistent theory has an extension to a complete consistent theory and follows as usual from Zorn's Lemma. The other fundamental result is the Compactness Theorem, which states that if all finite subsets of a theory $\Gamma$ have a model, then $\Gamma$ has a model; this follows from König's Lemma.

Shoenfield observed in [174] that in general, the family of complete, consistent extensions of an axiomatizable first order theory can be represented by a $\Pi^0_1$ class. Now the undecidability of arithmetic was discovered by Turing and Church (independently) in 1936, following soon after Gödel's incompleteness theorem. This result stated that there is no decidable complete consistent extension of Peano Arithmetic and also showed, in our terminology, that there is a nonempty c. b. $\Pi^0_1$ class with no computable member. This led to the definition of an *essentially undecidable* theory as a theory with no decidable complete consistent extension. Now if $\Sigma$ is any consistent complete extension of a theory $\Gamma$, then $\Sigma$ separates the set $T$ of consequences of $\Gamma$ from the set $R$ of refutations of $\Gamma$. A theory is said to be *separable* if the consequences and refutations can be separated by a computable set and is otherwise said to be *inseparable*. Rosser[166] observed also in 1936 that Peano arithmetic is an inseparable theory and that any inseparable theory is essentially undecidable. This also provided the first example of computably inseparable c. e. sets. Ehrenfeucht showed in 1961 [63] that there are separable theories which are essential undecidable. His construction, using theories of propositional calculus, also shows that every $\Pi^0_1$ class may be represented as the set of complete consistent extensions of a theory. A complete, consistent extension of Peano Arithmetic is of course just the the theory of some (possibly non-standard) model of Peano arithmetic. The theory of Peano arithmetic is of great interest in mathematical logic, due in part to the connection with Gödel's Incompleteness Theorem, and has been developed in the papers of Jockusch and Soare [91, 90], Knight [100], Marker [129] and many others.

**Theorem 2.9.1.** *(Shoenfield [174] For any c. e. theory $\Gamma$ of an effective language $\mathcal{L}$, both the class of consistent extensions of $\Gamma$ and the class of complete consistent extensions of $\Gamma$ can be represented as $\Pi^0_1$ classes. Furthermore, if $\Gamma$ is a decidable theory, then these classes can be represented by computable trees with no dead ends.*

*Proof.* Let $\mathcal{L}$ be an effective first-order language and let $S = Sent(\mathcal{L})$ have an effective enumeration as $\gamma_0, \gamma_1, \ldots$. Then the sentence $\gamma_i$ may be identified with the number $i$, so that a theory $\Gamma$ is represented by the set $\{i : \gamma_i \in \Gamma\}$, and a class of theories is represented by a class in $\{0,1\}^\omega$. Let $\Gamma \vdash_s \gamma_i$ be the computable relation of which means that there is a proof of $\gamma_i$ from $\Gamma$ of length $s$. Then the class $P(\Gamma)$ of complete consistent extensions of $\Gamma$ may be represented by the set of infinite paths through the computable tree $T$ defined so that for any $\sigma = (\sigma(0), \ldots, \sigma(n-1))$, $\sigma$ is in $T$ if and only if the following conditions hold.

**(1)** For any $i < n$, if $\Gamma \vdash_n \gamma_i$, then $\sigma(i) = 1$.

**(2)** For any $i, j < n$, if $\Gamma \vdash_n \gamma_i \to \gamma_j$ and $\sigma(i) = 1$, then $\sigma(j) = 1$.

**(3)** For any $i, j, k < n$, if $\gamma_k = (\gamma_i \;\&\; \gamma_j)$, $\sigma(i) = 1$ and $\sigma(j) = 1$, then $\sigma(k) = 1$.

**(4)** For any $i, j < n$, if $\sigma(i) = 1$ and $\gamma_j = \neg\gamma_i$, then $\sigma(j) = 0$.

**(5)** For any $i, j < n$, if $\gamma_j = \neg\gamma_i$, then either $\sigma(i) = 1$ or $\sigma(j) = 1$.

Let $x$ be an infinite path through $T$ and let $\Delta = \{\gamma_i : x(i) = 1\}$. Condition (1) ensures that $\Gamma \subseteq \Delta$, while conditions (1), (2), and (3) ensure that $\Delta$ is a theory. Condition (4) ensures that $\Delta$ is consistent and condition (5) ensures that $\Delta$ is complete. To represent the class of consistent extensions of $\Gamma$, simply omit the final clause (5).

If $\Gamma$ is decidable, then in each case we can modify the clauses given above as follows to get a tree $S$ with no dead ends which has the same class of infinite paths. First, combine the first three clauses into the statement:

**(1′)** : For any $k < n$, if $\Gamma \vdash \wedge\{\gamma_i : i < n \;\&\; \sigma(i) = 1\} \to \gamma_k$, then $\sigma(k) = 1$.

Next, replace clause (4) with

**(4′)** It is not the case that $\Gamma \vdash [\wedge\{\gamma_i : i < n \;\&\; \sigma(i) = 1\} \to (\gamma_0 \& \neg\gamma_0)]$.

It follows that for any $\sigma \in S$, $\Gamma \cup \{\gamma_i : i < |\sigma| \;\&\; \sigma(i) = 1\} \cup \{\neg\gamma_i : i < n \;\&\; \sigma(i) = 0\}$ is consistent and therefore has an extension to a complete consistent theory $\Gamma(\sigma)$ which will be represented by an extension of $\sigma$. Thus $S$ has no dead ends. $\square$

We can now apply Theorem 2.2.15 to logical theories.

**Theorem 2.9.2.** *For any consistent, axiomatizable first-order theory $\Gamma$:*

*(i) $\Gamma$ has a complete consistent extension which is computable in $\mathbf{0}'$.*

*(ii) If $\Gamma$ is decidable, then $\Gamma$ has a complete, consistent, decidable extension.*

Next we turn to the other direction of our correspondence, that is, representing an arbitrary $\Pi^0_1$ class by the set of complete consistent extensions of some axiomatizable theory.

**Theorem 2.9.3.** *Any c. b. $\Pi^0_1$ class $P$ may be represented by the set of complete, consistent extensions of an axiomatizable theory $\Gamma$ in propositional logic. Furthermore, if $P$ is a decidable $\Pi^0_1$ class, then $\Gamma$ may be taken to be a decidable theory.*

*Proof.* We give the proof due to Ehrenfeucht [63]. Let the language $\mathcal{L}$ consist of a countable sequence $A_0, A_1, \ldots$ of propositional variables. For any $x \in \{0,1\}^{\mathbb{N}}$, we can define a complete consistent theory $\Delta(x)$ for $\mathcal{L}$ to be $Con(\{C_i : i \in \omega\})$, where $C_i = A_i$ if $x(i) = 1$ and $C_i = \neg A_i$ if $x(i) = 0$. It is clear that every complete consistent theory of $\mathcal{L}$ is one of these. Thus for any $\Pi^0_1$ class $P \subseteq \{0,1\}^{\mathbb{N}}$, we want a theory $\Gamma$ such that $\Delta(P) = \{\Delta(x) : x \in P\}$ is the set of complete, consistent extensions of $\Gamma$.

For each finite sequence $\sigma = (\sigma(0), \ldots, \sigma(n-1))$, let $P_\sigma = C_0 \wedge C_1 \wedge \cdots \wedge C_{n-1}$, where $C_i = A_i$ if $\sigma(i) = 1$ and $C_i = \neg A_i$ if $\sigma(i) = 0$. Let the binary tree $T$ be given such that $P = [T]$ and define the theory $\Gamma(T)$ to consist of all $P_\sigma \to A_n$ such that $\sigma \in T$ and $\sigma^\frown 0 \notin T$ and all $P_\sigma \to \neg A_n$ such that $\sigma \in T$ and $\sigma^\frown 1 \notin T$, where $|\sigma| = n$. We claim that $\Delta(P)$ is in fact equal to the set of complete consistent extensions of $\Gamma(T)$. Suppose first that $x \in P$ and let $Con(\{C_i : i \in \omega\}) = \Delta(x)$. Now any $\gamma \in \Gamma(T)$ is of the form $P_\sigma \to \pm A_i$ for some $\sigma \in T$; say that $|\sigma| = n$. There are several cases. If $\sigma \neq x\lceil n$, then $\Delta(x) \vdash \neg P_\sigma$, so that we always have $\Delta(x) \vdash P_\sigma \to \pm A_n$. Thus we may suppose that $\sigma = x\lceil n$. If $\sigma^\frown 0 \notin T$, then of course $x(n) = 1$, so that $C_n = A_n \in \Delta(x)$ and therefore $\Delta(x) \vdash P_\sigma \to A_n$. Similarly, if $\sigma^\frown 1 \notin T$, then $\Delta(x) \vdash P_\sigma \to \neg A_n$. Thus $\Delta(x)$ is a complete consistent extension of $\Gamma(T)$. On the other hand, let $\Delta$ be a complete consistent extension of $\Gamma(T)$. Then, for each $i$, we have either $\Delta \vdash A_i$ or $\Delta \vdash \neg A_i$; let $C_i = A_i$ if $A_i \in \Delta$ and $C_i = \neg A_i$ otherwise. Define $x \in \{0,1\}^\omega$ so that $x(i) = 1$ if and only if $\Delta \vdash A_i$. Then clearly $\Delta = \Delta(x)$. It remains to be shown that $x \in P$. Now if $x \notin P$, then there is some $n$ such that $\sigma = x\lceil n+1 \notin T$ and $x\lceil n \in T$. Then $P_\sigma = C_0 \wedge \cdots \wedge C_{n-1}$, so that $\Delta \vdash P_\sigma$, and $P_\sigma \to \neg C_i \in \Gamma(T)$, so that $\Delta$ is not consistent with $\Gamma(T)$. This contradiction proves that $\Delta = \Delta(x)$.

Now suppose that $P$ is a decidable class, so that the tree $T$ has no dead ends. Let a sentence $\gamma = \gamma(A_0, \ldots, A_{n-1})$ of the language $\mathcal{L}$ be given. We claim that $\Gamma(T) \vdash \gamma$ if and only if $\bigwedge\{P_\sigma \vdash \gamma : \sigma \in T \ \& \ |\sigma| = n\}$, that is, if and only if $P_\sigma \vdash \gamma$ for all $\sigma \in T$ with $|\sigma| = n$. This claim clearly implies that $\Gamma(T)$ is decidable.

We argue by the contrapositive. Suppose first that $\Gamma(T) \not\vdash \gamma$. Then there is some $x \in [T]$ such that $\Delta(x) \vdash \neg\gamma$. Since $\gamma$ only depends on $A_0, \ldots, A_{n-1}$, it follows that $P_\tau \vdash \neg\gamma$, where $\tau = x\lceil n \in T$. Thus $P_\tau \vdash \gamma$ is clearly false, making it also false that $\bigwedge\{P_\sigma \vdash \gamma : \sigma \in T \ \& \ |\sigma| = n\}$. Suppose next that $\bigwedge\{P_\sigma \vdash \gamma : \sigma \in T \ \& \ |\sigma| = n\}$ is false. Then $P_\tau \vdash \gamma$ is false for some fixed $\tau \in T$, which means that $P_\tau \vdash \neg\gamma$ (since $\gamma$ depends only on $A_0, \ldots, A_{n-1}$). Since $T$ has no dead ends, there is some $x \in P$ such that $\tau \prec x$ and therefore $\Delta(x) \vdash \neg\gamma$ and therefore $\Gamma(T) \not\vdash \gamma$. □

This proof can be adapted to first order logic; see Exercise 1 below.

This representation theorem has the following corollary.

**Theorem 2.9.4.** *There is a consistent axiomatizable first-order theory Gamma which has no computable consistent complete extension.*

The perfect thin class constructed by Martin and Pour-El (see Theorem

2.8.1) was designed to produce a certain type of axiomatizable theory. A theory $\Gamma$ is said to be *Martin–Pour-El* if every axiomatizable extension of $\Gamma$ is generated by a single proposition. It follows from the proof of the next theorem that an axiomatizable theory $\Gamma$ is a Martin–Pour-El theory if and only if the class of complete consistent extensions of $\Gamma$ is a thin $\Pi^0_1$ class.

**Theorem 2.9.5.** *There exists an axiomatizable, essentially undecidable theory $T$ such that each axiomatizable extension of $T$ is a finite extension of $T$.*

*Proof.* Let $P$ be a perfect thin class with no computable members and let the axiomatizable theory $T$ be given by Theorem 2.9.3 such that $P$ represents the family of complete consistent extensions of $T$. Now suppose that $\Gamma$ is an axiomatizable extension of $T$. Then the family of complete consistent extensions of $\Gamma$ is represented by a $\Pi^0_1$ subclass $Q$ of $P$. Since $P$ is thin, there is a clopen set $U$ such that $Q = U \cap P$. Let $U = I(\sigma_1) \cup I(\sigma_2) \cup \cdots \cup I(\sigma_n)$ for some distinct finite sequences $\sigma_i$ all having the same length $k$ and let $\phi_i = P_{\sigma_i}$ as in the proof of Theorem 2.9.3. Then the complete consistent extensions of $\Gamma$ are exactly those complete consistent extensions of $\Delta$ which satisfy $P_1 \vee \cdots \vee P_n$. □

**Exercises**

2.9.1. Show that any c. b. $\Pi^0_1$ class may be represented as the set of complete consistent extensions of a first order logical theory in the language of one binary relation $R$ Jockusch and Soare in ([91], p. 54). Hint: the underlying axioms assert that $R$ is an equivalence relation and that, for any $n$, there are either one or two equivalence classes consisting of exactly $n$ members. The propositional statement $A_n$ in the proof above is replaced by the statement that there is exactly one equivalence class with $n$ elements.

2.9.2. Let the propositional language $\mathcal{L}$ have variables $A_0, A_1, \ldots$. Variables and their negations $\neg A_i$ are said to be *literals*. Show that a consistent theory $\Gamma$ for $\mathcal{L}$ is c. e. if and only if the set $C(\Gamma)$ of conjunctions of literals, consistent with $\Gamma$, is co-c. e. and that $\Gamma$ is decidable if and only if $C(\Gamma)$ is computable. Show that this is not true for the set of *literals* consistent with $\Gamma$.

# Chapter 3

# Members of $\Pi^0_1$ Classes

In this chapter, we study the complexity of members of $\Pi^0_1$ classes. We present some "basis theorems" and "anti-basis theorems". The class $\Gamma \subset \mathbb{N}^{\mathbb{N}}$ is said to be a *basis* for a family $\Theta$ of subclasses of $\mathbb{N}^{\mathbb{N}}$ if every nonempty class from $\Theta$ has a member from $\Gamma$. For example, the class $\Delta^0_0$ of computable reals is a basis for the family of open subclasses of $\mathbb{N}^{\mathbb{N}}$. This is an example of a "basis theorem". We have already given the simple positive result 4.2.3 that the class $P = [T]$ of infinite paths through the tree $T$ contains a member computable from $Ext(T)$, the set of nodes of $T$ which have an infinite extension in $P$. Recall in particular, that if $T$ is computably bounded, then $P$ has a member computable in $\mathbf{0}'$ and if $P$ is decidable, then $P$ has a computable element. We will give several more basis results, including the Low Basis Theorem of Jockusch and Soare [91] is given.

On the other hand, the class of computable reals is not a basis for the family of closed subclasses of $\mathbb{N}^{\mathbb{N}}$ since every singleton is a closed class. This is an example of an "anti-basis theorem". One result given is that the set of Boolean combinations of c. e. sets is not a basis for the c. b. $\Pi^0_1$ classes. Any c. b. $\Pi^0_1$ class with no computable members is perfect and has a set of continuum many mutually Turing incomparable elements [91]. There is a c. b. $\Pi^0_1$ class of positive measure which has no computable element.

## 3.1 Basis theorems

One of the most cited results in the theory of $\Pi^0_1$ classes is the Low Basis Theorem of Jockusch and Soare [91]. We will introduce the method of *forcing with $\Pi^0_1$ classes* in connection with this theorem. First we consider the notion a *generic* real.

**Definition 3.1.1.** *An element $x \in \mathbb{N}^{\mathbb{N}}$ is said to be* 1-generic *if, for every $\Pi^0_1$ class $P$, there exists $n$ such that either $I(x \upharpoonright n) \subset P$ or $I(x \upharpoonright n) \cap P = \emptyset$.*

The existence of a generic real is obtained by forcing. Let $P = [T]$. The idea here is that the finite sequence $\sigma = x\lceil n$ "forces" $x \notin P$ if $\sigma \notin Ext(T)$, that is, $\sigma \Vdash x \notin P$ if *no* extension of $\sigma$ is in $P$, and similarly $\sigma \Vdash x \in P$ if $I(\sigma) \subseteq P$, that is, if *every* extension of $\sigma$ is in $P$. Then we write $\Vdash x \in P$ ($\Vdash x \notin P$) if there is some $\sigma$ such that $\sigma \Vdash x \in P$ ($\sigma \Vdash x \notin P$). With this notation, $x$ is 1-generic if, for every $\Pi^0_1$ class $P$, either $\Vdash x \in P$ or $\Vdash x \notin P$, or equivalently $x \in P$ implies $\Vdash x \in P$. (See Section III.6 of Hinman [80] for a presentation of arithmetical forcing.)

A subset $D$ of $\{0,1\}^*$ is said to be *dense* if, for any $\sigma$, there exists a $\tau \succ \sigma$ such that $\tau \in D$. Now let $\mathcal{D} = \{D_i : i \in I\}$ be a family of dense sets. The element $x$ of $\{0,1\}^{\mathbb{N}}$ is said to be $\mathcal{D}$-*generic* if, for each $i$, $x \restriction n \in D_i$ for some $n$.

The standard forcing theorem shows that any countable family of dense sets possesses a generic set. We observe that this is an effective version of the Baire Category Theorem.

**Lemma 3.1.2.** *If $\mathcal{D} = \{D_i : i < \omega\}$ is a sequence of subsets of $\mathbb{N}^*$ uniformly computable in $\mathbf{0}'$, there exists a $\mathcal{D}$-generic $x \leq_T \mathbf{0}'$.*

*Proof.* Let $m$ be the least such that $\sigma_m \in D_0$ and let $\tau_0 = \sigma_m$. Then for each $n$, find the least $m$ such that $\sigma_m$ is a proper extension of $\tau_n$ and let $\tau_{n+1} = \sigma_m$. Then $x = \cup_n \tau_n$ will be the desired generic real. This construction is computable using an oracle for the sequence $\mathcal{D}$. □

Recall that by Theorem 2.3.3, there is a uniformly primitive recursive enumeration of trees $T_e$ such that $P_e = [T_e]$ is the $e$th $\Pi^0_1$ class. Then the standard family of dense sets is now $D_i = \{\sigma : \sigma \notin T_i \ \vee \ (\forall \tau \succeq \sigma)\tau \in T_i\}$. Observe each $D_i$ is dense and that the sequence of sets is uniformly $\Pi^0_1$. The element $x \in \{0,1\}^{\mathbb{N}}$ is 1-generic if it is generic for this sequence of dense sets. Then the remarks above imply the existence of a 1-generic real. The crucial property of a 1-generic real is given by the following well-known fact.

**Theorem 3.1.3.** *For any 1-generic $x \in \{0,1\}^{\mathbb{N}}$, if $x \leq_T \mathbf{0}'$, then $x' \leq_T x \oplus \mathbf{0}'$.*

*Proof.* Let $x$ be 1-generic. For each $e$, let the $\Pi^0_1$ class $P_e = \{y : \phi^y_e(e)\uparrow\}$, so that $P_e = [U_e]$, where $\sigma \in U_e \iff \phi^\sigma_e(e)\uparrow$. Thus $e \in x' \iff x \notin P_e$. If $e \in x'$, then of course there is some $n$ such that $x\lceil n \notin U_e$. Since $x$ is 1-generic, if $e \notin x'$, then there is some $n$ such that $(\forall \tau \succeq x\lceil n)\tau \in U_e$. Let $f(e)$ be the least $n$ such that either $x\lceil n \notin U_e$ or $(\forall \tau \succeq x\lceil n)\tau \in U_e$. Then $f$ is computable in $x \oplus \mathbf{0}'$. But then we have $e \in x'$ if and only if $x\lceil f(n) \notin U_e$, so that $x'$ is also computable in $x \oplus \mathbf{0}'$. □

It follows that if a 1-generic real $x$ is computable in $\mathbf{0}'$, then $x' = \mathbf{0}'$, that is, $x$ is low. For the low basis theorem of Jockusch and Soare, a modification of this argument is used.

**Theorem 3.1.4** (Low Basis Theorem)**.** *(a) Every nonempty c. b. $\Pi^0_1$ class $P$ contains a member of low degree.*

*(b) There is a low degree $\mathbf{a}$ such that every nonempty r. b. $\Pi^0_1$ class contains a member of degree $\leq \mathbf{a}$.*

*Proof.* (a) We may assume that $P \subset \{0,1\}^\omega$. Let $P = [T]$ and, as above, let $\sigma \in U_e \iff \phi_e^\sigma(e) \uparrow$. We will define, computably in $\mathbf{0}'$, a sequence $T = S_0 \supset S_1 \supset S_2 \supset \cdots$ of infinite subtrees of $T$ and show that any member $x$ of $\cap_e[S_e]$ has low degree. There are two cases in the definition of $S_{e+1}$.

1. If $S_e \cap U_e$ is finite, then $S_{e+1} = S_e$.

2. If $S_e \cap U_e$ is infinite, then $S_{e+1} = S_e \cap U_e$.

Observe that this construction is computable in $\mathbf{0}'$, in that there is a function $f \leq_T \mathbf{0}'$ such that $S_e = U_{f(e)}$ for each $e$. This is because the determination of whether $S_e \cap U_e$ is finite can be made using a $\mathbf{0}'$ oracle. Since each $S_e$ is infinite by the construction, it follows that each $[S_e]$ is nonempty, so that $\cap_e[S_e]$ is nonempty. Now suppose that $x \in \cap_e[S_e]$. Then, for any $e$, we have $e \in x'$ if and only if $x \notin [U_e]$, and it follows from the construction that $x \notin [U_e]$ if and only if $S_e \cap U_e$ is finite. It follows by the observation above that $x' \leq_T 0'$.

(b) It suffices to prove the result for classes in $\{0,1\}^{\mathbb{N}}$. Let $P_e = [T_e]$ be an effective enumeration of the $\Pi^0_1$ in $\{0,1\}^{\mathbb{N}}$ and let $p_e$ be the $e$'th prime number. Let the tree $T$ be the amalgamation of the nonempty $\Pi^0_1$ classes, in the following sense. Let $\sigma \in T$ if, for each $e$ such that $P_e$ is nonempty and each $k$ such that $p_e^k < |\sigma|$, $(\sigma(p_e), \sigma(p_e^2), \ldots, \sigma(p_e^k)) \in T_e$. Since we can test computably in $\mathbf{0}'$ whether $P_e$ is nonempty, the tree $T$ is computable in $\mathbf{0}'$. Thus the construction above can be carried out to produce a member $x$ of $P = [T]$ of low degree. Then any nonempty class $P_e$ has a member $(x(p_e), x(p_e^2), \ldots)$ computable in $x$. $\square$

The same technique can be used to prove other basis results. For example, a generalization shows that if the c. b. class $P$ contains no computable member, then for any degree $\mathbf{b}$, $P$ has a member $A$ of degree $\mathbf{a}$ such that $\mathbf{a} \oplus \mathbf{0}' = \mathbf{a}' = \mathbf{b}'$. It follows (for $\mathbf{b} = \mathbf{0}'$) that any $\Pi^0_1$ class $P$ has a member $A$ of degree $\mathbf{a}$ such that $\mathbf{a} \oplus \mathbf{0}' = \mathbf{a}' = \mathbf{0}''$. Now a degree $\mathbf{a}$ is said to be *high* if $\mathbf{a} \leq \mathbf{0}'$ and $\mathbf{a}' = \mathbf{0}''$. It will be shown later that not every r. b. $\Pi^0_1$ class contains a member of high degree.

The following result is from Jockusch and Soare [90].

**Theorem 3.1.5** (Jockusch and Soare)**.** *Every nonempty c. b. $\Pi^0_1$ class $P$ contains a member of hyperimmune-free degree, that is, contains an almost computable member.*

*Proof.* We sketch the proof indicated in Soare [181] (p. 109). Let $P = [T]$, where $T$ is an infinite computable binary tree. Recall that $A$ is hyperimmune-free if every function $f$ computable in $A$ is majorized by some computable function. Thus we want to find $A \in P$ such that $\phi_e^A$ whenever total, is majorized by a computable function. We define the decreasing sequence $S_e$ of computable subtrees of $T$ beginning with $S_0 = T$. Then for each $e$ and $i$, let $U_e^i = \{\sigma \in S_e : \phi_e^\sigma(i) \uparrow\}$. There are again two cases in the definition of $S_{e+1}$.

1. If $U_e^i$ is finite for every $i$, let $S_{e+1} = S_e$.

2. If $U^i_e$ is infinite for some $i$, choose such an $i$ and let $S_{e+1} = S_e \cap U^i_e$.

Suppose now that $A \in \cap_e S_e$ and that $f = \phi^A_e$ is total. If the second case applied in the definition of $S_{e+1}$, then $\phi^A_e(i) \uparrow$, so that $\phi^A_e$ is not total. If the first case applied, then $\Phi^A_e$ is defined for all $A \in [S_{e+1}]$. Since $[S_e] \subseteq P$, $[S_e]$ is computably bounded and it follows from Theorem 2.4.8 that $\Phi_e[S_e]$ is also bounded by some computable function $h$, which thus majorizes $\Phi^A_e$. □

We will show in the next section that a nonempty $\Pi^0_1$ class $P \subseteq \{0,1\}^{\mathbb{N}}$ need not contain any sets which are c. e. or co-c. e.. However, it always contains elements which corresponds to c. e. and co-c. e. Dedekind cuts.

**Theorem 3.1.6.** *Any nonempty $\Pi^0_1$ class $P \subseteq \{0,1\}^{\mathbb{N}}$ contains elements $x$ and $y$ such that the Dedekind cut $L(r_x)$ is c. e. and the Dedekind cut $L(r_y)$ is co-c. e..*

*Proof.* Let $P = [T]$ where $T$ is a computable tree. Let $x$ be the "leftmost" element of $P$ under the lexicographic order. For each $n$, let $\sigma_n$ be the leftmost node in $T \cap \{0,1\}^n$ and let $q_n = r_\sigma = \sum_{i=0}^n \sigma(i)2^{-i-1}$. It is clear that $\{q_n\}_{n<\omega}$ is an increasing sequence and hence has limit $r$ such that $L(r)$ is c. e. set by Proposition 2.1.8.3. We claim that $x = lim_n \sigma_n$ converges to a path in $P$ and that $r = r_x$. For each $n$, $\sigma_n(0) \leq \sigma_{n+1}(0)$, since otherwise $\sigma_{n+1}\lceil n <_{lex} \sigma_n$. Thus the sequence $\sigma_n(0)$ converges to some $x(0)$. Once $\sigma_n(0), \dots, \sigma_n(k-1)$ have all converged, then $\sigma_n(k)$ becomes increasing and thus also converges to some limit which we call $x(k)$. It remains to show that $x \in P$. Fix $n$ and choose $s > n$ such that $\sigma_m(i) = x(i)$ for all $i < n$ and all $m \geq s$. Then $x\lceil n \prec \sigma_s$ and hence $x\lceil n \in T$.

A similar argument shows that the "rightmost" element $Y$ of $P$ corresponds to a real with a $\Pi^0_1$ Dedekind cut. □

Some further basis results are given below in Chapter 4.

**Exercises**

3.1.1. The Baire Category Theorem for $\mathbb{N}^{\mathbb{N}}$ states that the countable intersection of a sequence of dense open sets is nonempty. Use forcing to prove this theorem.

3.1.2. Show that $x$ is 1-generic if and only if it belongs to every $\Sigma^0_1$ co-meager set (equivalently, every non-meager $\Sigma^0_1$ set).

3.1.3. Show that if the c. b. class $P$ contains no computable member, then for any degree $\mathbf{b}$, $P$ has a member $A$ of degree $\mathbf{a}$ such that $\mathbf{a} \oplus \mathbf{0}' = \mathbf{a}' = \mathbf{b}'$. Then show that any $\Pi^0_1$ class $P$ has a member of degree $\mathbf{a}$ such that $\mathbf{a} \oplus \mathbf{0}' = \mathbf{a}' = \mathbf{0}''$.

3.1.4. Show that if $x$ is an isolated member of $P$, then $x$ is computable. (Define a decidable subclass of $P$ and use Theorem 2.2.15.)

3.1.5. Show that any nonempty $\Pi^0_1$ class which is almost recursively bounded must contain an element computable in $\mathbf{0}'$.

3.1.6. Show that for any $x$ and $y$ with $x <_{lex} y$ such that $L(r_X)$ is $\Sigma^0_1$ and $L(r_y)$ is $\Pi^0_1$, there is a $\Pi^0_1$ class with leftmost element $x$ and rightmost element $y$.

## 3.2 Special $\Pi^0_1$ classes

A c. b. $\Pi^0_1$ class is said to be *special* if it has no computable members.

We have seen in Section 3.2.5(Exercise (3)) that the diagonally non-computable sets form a $\Pi^0_1$ class with no computable element. We give an improvement of this result due to Jockusch [84]. Recall that a set $A$ is *immune* if it has no infinite recursive subset; $A$ is said to be *bi-immune* if both $A$ and $\mathbb{N}\setminus A$ are immune. It is elementary that any infinite c. e. set has an infinite recursive subset, and it follows that the difference of two c. e. sets cannot be bi-immune and then by induction that no Boolean combination of c. e. sets can be bi-immune.

**Theorem 3.2.1** (Jockusch)**.** *There is a nonempty $\Pi^0_1$ class of sets containing only bi-immune sets.*

*Proof.* Let $W_e$ be the $e$'th c.e. set and let $D_n$ be the $n$'th finite set. Let $\psi$ be a partial recursive function such that, whenever $|W_e| \geq e+3$, then $\psi(e)$ is defined and $D_{\psi(e)} \subset W_e$ and $|D_{\psi(e)}| = e+3$. Define the $\Pi^0_1$ class $P = \cap_e P_e$, where $A \in P_e$ if and only if, if $\psi(e)$ is defined, then $A \cap D_{\psi(e)} \neq \emptyset$ and $(\mathbb{N}\setminus A) \cap D_{\psi(e)} \neq \emptyset$. Any element $A$ of $P$ is clearly bi-immune. To see that $P$ is nonempty, note that for each $e$, $\{0,1\}^{\mathbb{N}} \setminus P_e$ has measure $\leq 2^{-e-2}$. (For $A \notin P_e$, either all $e+3$ elements of $D_{\psi(e)}$ are in $A$ or all $e+3$ elements are not in $A$, which allows only 2 of the $2^{e+3}$ possibilities.) It follows that $\{0,1\}^{\mathbb{N}} \setminus P$ has measure $\leq \sum_e 2^{-e-2} = \frac{1}{2}$, so that $P \neq 0$. □

This immediately implies the following.

**Theorem 3.2.2.** *(Jockusch) There is a nonempty c. b. $\Pi^0_1$ class with no member a Boolean combination of r.e. sets.* □

A set $A$ is said to be *effectively immune* if there is a computable function $g$ such that for any $e$, if $W_e \subset A$, then $|W_e| \leq g(e)$. For the $\Pi^0_1$ class $P$ constructed in the proof of Theorem 3.2.1, it is clear that any set $A \in P$ is effectively bi-immune via the function $g(e) = e+3$. This yields the following corollary, which we note is essentially exercise 4.2 on page 87 of [181].

**Theorem 3.2.3.** *There is a nonempty c. b. $\Pi^0_1$ class such that if $\mathbf{a}$ is the degree of a member of $P$ and $\mathbf{b}$ is a c.e. degree and $\mathbf{a} \leq \mathbf{b}$ then $\mathbf{b} = \mathbf{0}'$.*

*Proof.* Let $P$ be the $\Pi^0_1$ class defined in the proof of Theorem 3.2.1. Then every member of $P$ is effectively immune. Now suppose that $P$ had a member $C$ of c. e. degree. We claim that $C$ must have degree $\mathbf{0}'$. By the Modulus

Lemma (Soare [181], $C = \lim_s C_s$ with a *modulus function* $m(i)$ computable in $C$ such that $s \geq m(i)$ implies that $i \in C \iff i \in C_s$. We may assume without loss of generality that each $C_s$ is infinite and let $c_{0,s}, c_{1,s}, \dots$ enumerate in increasing order the elements of $C_s$. Let $C = \{c_0, c_1, \dots\}$ enumerate $C$ in increasing order, so that for each $n$, $c_n = \lim_s c_{n,s}$. Now let the complete c. e. set $K$ have enumeration $K_s$ and let the partial recursive function $\theta$ be defined so that $\theta(i) = s$ if and only if $s$ is the least such that $x \in K_s$.

By the recursion theorem, define the computable function $h$ so that $W_{h(i)} = \emptyset$, if $i \notin K$ and otherwise $W_{h(i)} = \{c_{0,\theta(i)}, c_{1,\theta(i)}, \dots, c_{g(h(i)),\theta(i)}\}$.
Let $r(i)$ be the least $s$ such that, for all $j \leq g(h(i))$ and all $t \geq s$, $c_{j,t} = c_j$. Then the function $r$ can be computed from $C$ using the modulus function $m$. If $r(i) \leq \theta(i)$, then $W_{h(i)} \subset C$, so that $C$ has $g(h(i)) + 1$ elements, contradicting the hypothesis on $g$. It follows that $\theta(i) < r(i)$ for all $i$, so that $K$ is computable from $C$, as desired. □

It follows that the only possible c. e. degree of a member of $P$ is $\mathbf{0}'$. There is a more general version of this result, Theorem 5 of [90].

**Theorem 3.2.4.** *For any c. e. degree* $\mathbf{c}$*, there is a nonempty c. b.* $\Pi^0_1$ *class* $P$ *such that the c. e. degrees of member of* $P$ *are precisely the c. e. degrees above* $\mathbf{c}$.

*Proof.* For $\mathbf{c} = 0$, this is trivial. For $\mathbf{c} \neq 0$, let $A$ be the simple but not hypersimple c. e. set of degree $\mathbf{c}$ from Theorem 1.11.8. Let $D_{f(n)}$ be a disjoint strong array such that $D_{f(n)} \setminus A \neq \emptyset$ for all $n$ and assume without loss of generality that $Card(D_{f(n)} \setminus A) \geq 2$. Now define the $\Pi^0_1$ class $P$ by

$$D \in P \iff D \cap A = \emptyset \;\&\; (\forall n)(D \cap D_{f(n)} \neq \emptyset).$$

For any $D \in P$ of c. e. degree, it follows from Theorem 1.11.8 that $A \leq_T D$.

Now let $E$ be a set of any degree $\mathbf{d} \geq c$ and find $D \in P$ such that $D \leq_T E$ and

$$Card(D_{f(n)} \setminus D) = 1 \iff n \in E.$$

Clearly $E \leq_T D$. Since $A \leq_T D$, we can compute from $D$ a sequence of pairs $\langle a_n, b_n \rangle$ with $a_n, b_n$ both in $D_{f(n)} \setminus A$. Let $E = \{e_0 < e_1 < \dots$. Then we can code $D$ into $E$ by letting $D_{f(e_i)} \setminus D = \{a_{e_i}\}$ if $i \in D$ and $D_{f(e_i)} \setminus D = \{b_{e_i}\}$ otherwise. Thus $D$ will be a member of $P$ with degree $\mathbf{d}$. □

We next show that every special c. b. $\Pi^0_1$ class satisfies a weakened form of Theorem 3.2.3.

**Theorem 3.2.5.** *(Jockusch-Soare [90]) For any special* $\Pi^0_1$ *class, there exists a non-zero c. e. degree* $\mathbf{a}$ *such that* $P$ *has no members of degree* $\leq \mathbf{a}$.

*Proof.* We give the proof from [90]. Let $P = [T]$ where $T$ is a computable tree. We will construct a simple (and hence noncomputable) c. e. set $A$ in stage $A_s$, such that $\phi^A_e \notin P$ for any $e$.

Two binary computable functions will be used in the construction.

$p(e,s) = max\{n \le s : \Phi_e(A_s\lceil s)\lceil n) \in T\}$;

$q(e,s) = (least\ j) : |\Phi_e(A_s\lceil j)| \ge p(e,s)\}$.

Initially, $A_0 = \emptyset$. At stage $s+1$, let $e_{s+1}$ be the least $e \le s$ such that $W_{e,s} \cap A_s = \emptyset$ and $W_{e,s}$ contains $u \ge max\{q(e',s) : e' \le e\} \cup \{2e\}$. Let $A_{s+1} = A_s \cup \{u\}$, where $u$ is the least such number for $e = e_{s+1}$. (If no such $e$ exists, let $A^{s+1} = A_s$.)

**Claim 1** No element of $P$ is computable in $A$.

Proof of Claim 1: Suppose by way of contradiction that $y = \phi^A_e \in P$ (and is thus a total function). It follows that $lim_s p(e,s) = \infty$. For the contradiction, we will show that $y$ is computable. Let $t$ be a stage such that for all $e' < e$, if $W_e \cap A \ne \emptyset$, then $W_{e,t} \cap A_t \ne \emptyset$. Then for all $s \ge t$, either $e_s$ is undefined or $e_s \ge e$. To compute $y(n)$, find $s_n \ge t$ such that $n < p(e,s_n)$, so that $\Phi_e(A\lceil s_n, n)$ is defined and in $T$. We claim that $y(n) = \Phi_e(A\lceil s_n, n)$. To show this, it suffices to prove that no number $u < q(e,s_n)$ enters $A$ after stage $s_n$. But if $u \in A^{s+1} - A_s$ and $s+1 > s_n$, then $e_{s+1} \ge e$ and so $u \ge q(e,s_n)$ by the construction.

**Claim 2** For fixed $e$, $q(e,s)$ is bounded over all $s$.

Proof of Claim 2:

Let $n$ be the least such that $\Phi^A_e\lceil n \notin T$ and let $t$ be as in the proof of Claim 1. It follows that $p(e,s) \le n$ for all $s \ge t$. Choose $j$ such that $\Phi^A_e\lceil n \preceq \Phi_e(A\lceil j)$. It follows that $q(e,s) \le j$ for all sufficiently large $s$.

**Claim 3** $A$ is simple.

Proof of Claim 3: For each c. e. set $W_e$, $A$ contains at most one member $u \ge 2e$ from $W_e$, so that $A$ contains at most $e$ elements $\le 2e$ and the complement of $A$ is infinite. Fix $e$ and let $j = max\{q(e',s) : e' \le e, s \in \mathbb{N}\}$. Let $t$ be given from Claim 1. Then if $s \ge t$ and $W_{e,s}$ contains a member $\ge max\{j, 2e\}$, then $W_{e,s+1} \cap A \ne \emptyset$ by the construction. Hence if $W_e$ is infinite, then $W_e \cap A \ne \emptyset$.

This completes the proof of the theorem. $\square$

As we saw in Exercise 4, any isolated member of a c. b. $\Pi^0_1$ class is computable. Hence, if $P$ is a special c. b. $\Pi^0_1$ class, then it is perfect and hence has cardinality $2^{\aleph_0}$. It follows that there are $2^{\aleph_0}$ different degrees of members of $P$. Several results deal with the comparability of these degrees.

It is easy to see that if $F$ is a computable function from $\{0,1\}^{\mathbb{N}}$ to $\{0,1\}^{\mathbb{N}}$, $r \in \{0,1\}^{\mathbb{N}}$, and $P$ is a $\Pi^0_1$ class such that $F(x) = r$ for all $x \in P$, then $r$ is computable. (See the exercises.) The next lemma, from Jockusch-Soare [91], improves this observation.

**Lemma 3.2.6.** *Let $P$ be a nonempty c. b. $\Pi^0_1$ class $P$ and let $\Phi : \mathbb{N} \times \mathbb{N}^{\mathbb{N}} \to \mathbb{N}$ be a partial computable functional. Suppose that for any $n \in \mathbb{N}$ and any $x_1, x_2 \in P$, $\Phi(n,x_1) = \Phi(n,x_2)$ whenever they are both defined. Then there is a nonempty $\Pi^0_1$ class $Q \subseteq P$ such that, for all $y \in Q$, if $\lambda n\Phi(n,y)$ is total then it is computable.*

*Proof.* Let $P = [T]$ where $T$ is a computable tree and suppose that $P$ and $\Phi$ satisfy the hypothesis. There are two possibilities. Suppose first that for some $n$, $\Phi_e(n, \sigma)$ is undefined for infinitely many $\sigma \in T$. Then we may let $S$ be the subtree $\{\sigma \in T : \Phi_e(n, \sigma) \uparrow\}$ and let $Q = [S]$, since $y \in Q$ then implies that $\Phi_e(n, y) \uparrow$. Thus we may suppose that for any given $n$, $\Phi(n, \sigma) \downarrow$ for all but finitely many $\sigma \in T$. Thus $\Phi(n, y) \downarrow$ for all $y \in P$ and by the hypothesis there is a unique $k_n$ such that $\Phi(n, y) = k_n$ for all $y \in P$. Let $r(n) = k_n$ for each $n$. It now follows from the remark above that $r$ is computable. Thus in fact $\Phi(n, y) = r(n)$ for all $y \in P$ and all $n$. □

When $F$ is the identity function, we obtain the following corollary.

**Lemma 3.2.7.** *If $P$ is a nonempty c. b. $\Pi^0_1$ class with no computable elements, then any $\sigma \in Ext(T)$ has incompatible extensions $\tau_1, \tau_2 \in Ext(T)$.* □

**Theorem 3.2.8.** *(Jockusch-Soare [91]) For any c. b. $\Pi^0_1$ class $P$ with no computable members and any countable set $\{\mathbf{a}_i : i < \omega\}$ of noncomputable degrees, $P$ has continuum many mutually incomparable members $x$ such that the degree of $x$ is incomparable with each $\mathbf{a}_i$.*

*Proof.* We may assume without loss of generality that $P \subseteq \{0,1\}^{\mathbb{N}}$ and let $P = [T]$ for some computable tree $T \subseteq \{0,1\}^*$. For simplicity, we construct the elements of $P$ to be incomparable to a single noncomputable degree $\mathbf{a}$ and let $z$ have degree $\mathbf{a}$; for the general argument simply work on $\mathbf{a_0}, \ldots, \mathbf{a_n}$ at level $n$.

We will define a function $\psi : \{0,1\}^* \to Ext(T)$ such that $\sigma_1 \prec \sigma_2$ implies $\psi(\sigma_1) \prec \psi(\sigma_2)$ and infinite trees $S_\sigma \subseteq T \cap I(\psi(\sigma)$ such that, for all $n$

(1) If $|\sigma| = n + 1$ and $x \in [S_\sigma]$, then $\Phi^x_n \neq z$.

(2) If $|\sigma = |\tau| = n + 1$ and $\sigma \neq \tau$, then for any $x \in S_\sigma$ and $y \in S_\tau$, $\Phi^x_n \neq y$.

Initially, we let $F(\emptyset) = \emptyset$. Now suppose that we have defined $\psi(\sigma)$ and $S_\sigma$ for $|\sigma| = n$ and let $\Phi = \Phi_n$. By Lemma 3.2.6, there are two possible cases. First suppose that there is a nonempty $\Pi^0_1$ subclass $Q$ of $S_\sigma$ such that $\Phi^y_n$ is either computable or not total for all $y \in Q$. Then apply Lemma 3.2.7 to obtain incompatible extensions $\tau_0$ and $\tau_1$ of $F(\sigma)$ from $Ext(Q)$, by Lemma 3.2.7 and let $F(\sigma^\frown i) = \tau_i$ and $S_{\sigma^\frown i} = I(\tau_i) \cap Q$ for $i = 0, 1$.

In the second case, by Lemma 3.2.6, $F(\sigma)$ has extensions $\rho_0, \rho_1 \in Ext(S_\sigma)$ such that $\Phi_n(m, \rho_1) \neq \Phi_n(m, \rho_2)$ for some $m$. Without loss of generality, $\Phi_n(m, \rho_1) \neq z(m)$ and thus we first extend $F(\sigma)$ to $\rho_1$ to satisfy condition (1) and then take extensions $\tau_0$ and $\tau_1$ as above. In this case, $F(\sigma^\frown i)$ will be an extension of the provisional value $\tau_i$ for $i = 0, 1$ after we take care of condition (2). The provisional values for $S_{\sigma^\frown i}$ are $S_\sigma \cap I(\tau_i)$.

Now we will indicate how to satisfy condition (2). Let $\sigma_1, \sigma_2$ have length $n + 1$ with provisional values $\tau_1$ and $\tau_2$ for $F(\sigma_1)$ and $F(\sigma_2)$ and provisional values $Q_1$ and $Q_2$ for $S_{\sigma_1}$ and $S_{\sigma_2}$ with $Q_i \subseteq P \cap I(\tau_i)$. Our goal is to ensure that $\Phi^x \neq y$ for any $x \in S_{\sigma_1}$ and $y \in S_{\sigma_2}$. If $\sigma_1$ came under case 1 above, then this is already satisfied. Otherwise, find $m$ and extensions $\rho_1, \rho_2 \in S_{\sigma_1}$ such that

$\Phi(m,\rho_1) \neq \Phi(m,\rho_2)$ as above and find an extension $\tau$ of $\tau_2$ such that, without loss of generality, $\tau(m) \neq \Phi(m,\rho_1)$. Then the new provisional value of $F(\sigma_1)$ is $\rho_1$, the new provisional value of $F(\sigma_2)$ is $\tau$; the new provisional value of $S_{\sigma_1}$ is $Q_1 \cap I(\rho_1)$ and the new provisional value of $S_{\sigma_2}$ is $Q_2 \cap I(\tau)$. This satisfies one of the finitely many subconditions of (2).

We repeat this procedure for each pair $\sigma_1, \sigma_2 \in \{0,1\}^{n+1}$ to obtain final provisional values which will satisfy all of the necessary conditions. □

This leads to the following. Two noncomputable reals $x, y$ are said to be a *minimal pair* if any real computable both in $x$ and in $y$ is itself computable.

**Corollary 3.2.9.** *(Jockusch-Soare [91] Let $P \subset \{0,1\}^{\mathbb{N}}$ be a $\Pi^0_1$ class with no computable members. Then $P$ contains a minimal pair.*

*Proof.* Let $x$ be any member of $P$. By Theorem 3.2.8 $P$ contains an element $y$ which is incomparable with each of the (countably many) noncomputable reals which are computable in $x$. □

**Theorem 3.2.10.** *(Jockusch-Soare [90]) There is an infinite c. b. $\Pi^0_1$ class $P$ such that any two different members of $P$ are Turing incomparable.*

*Proof.* We will use a priority argument to define a uniformly computable sequence $f_s : \{0,1\}^* \to \{0,1\}^*$ of one-to-one, computable tree homomorphisms and corresponding perfect $\Pi^0_1$ classes $P_s = [T_s]$, where

$$T_s = \{\tau : (\exists\sigma)[f_s(\sigma) \preceq \tau]\}.$$

Then the desired (perfect) $\Pi^0_1$ class $P$ is defined to be $P = \cap_s P_s$. In particular, each $f_s$ satisfies the following for all $\sigma \in \{0,1\}^*$.

(1) $f_s(\sigma^\frown 0)$ and $f_s(\sigma^\frown 1)$ are incompatible extensions of $f_s(\sigma)$ for all $s$;

(2) $range(f_{s+1}) \subseteq range(f_s)$;

(3) $\lim_s f_s(\sigma) = f(\sigma)$ exists.

It follows that $f$ induces a homeomorphism $F$ from $\{0,1\}^{\mathbb{N}}$ onto $P$, defined by $F(x) = \cup_n f(x\lceil n)$. It follows that that $P$ is a perfect set, hence uncountable and therefore certainly nonempty.

The construction of $P$ will ensure that for each $y \in P$ and each partial function $\Phi_e$, $\Phi_e(y) \neq y$ implies that $\Phi_e(y) \notin P$. To accomplish this, we will ensure that for each $e$ and each $x \in \{0,1\}^{\mathbb{N}}$, if $z = \Phi_e(F(x))$ converges and $z \neq F(x)$, then $\Phi_e(f(x\lceil e))$ is incompatible with $F(x)$. That is, for each $e$ and each $\sigma \in \{0,1\}^{e+1}$, we have the following requirement.

$$\mathbf{R}_{e,\sigma} : (\forall y)[y \in I(f(\sigma)) \ \&\ \Phi_e(y) \text{ total } \&\ \Phi_e(y) \neq y \ \rightarrow\ \Phi_e(y) \notin P].$$

The priority order on the requirements is lexicographic, first on $e$ and then on $\sigma$.

Requirement $R_{e,\sigma}$ is said to be *satisfied at stage* $s$ if $\Phi_e(f(\sigma)) \notin T_s$. If this requirement is satisfied at stage $s$ and $y \in I(f(\sigma))$, then of course requirement $R_{e,\sigma}$ is actually satisfied, since $\Phi_e(f(\sigma)) \preceq \Phi_e(y)$.

Initially we set $f_0(\sigma) = \sigma$ for all $\sigma \in \{0,1\}^*$.

**Stage** $s+1$. Requirement $R_{e,\sigma}$ needs attention at stage $s+1$ if it is not satisfied at stage $s$ and there exists $\sigma' \succ \sigma$ such that $\Phi_e(f_s(\sigma'))$ is incompatible with $f_s(\sigma')$ and extends $f_s(\rho^\frown i)$ for some $\rho \in \{0,1\}^{e+1}$ and some $i \in \{0,1\}$.

If no requirement needs attention at stage $s+1$, then $f_{s+1} = f_\sigma$. Otherwise, let $R_{e,\sigma}$ be the requirement of highest priority which needs attention and let $\sigma'$, $\rho$ and $i$ be given as above. Define $f_{s+1}$ as follows, for any $\nu \in \{0,1\}^*$.

(4) $f_{s+1}(\sigma^\frown \nu) = f_s(\sigma'^\frown \nu)$;

(5) If $\rho \neq \sigma$, then $f_{s+1}(\rho^\frown \nu) = f_s(\rho^\frown(1-i)^\frown \nu)$;

(6) If $\nu$ does not extend either $\rho$ or $\sigma$, then $f_{s+1}(\nu) = f_s(\nu)$.

Now suppose that $f_{s+1}(\sigma) \prec y$. Then $\Phi_e(f_{s+1}(\sigma)) = \Phi_e(f_s(\sigma')) \succ f_s(\rho^\frown i) =\prec \Phi_e(y)$ and is incompatible with $f_{s+1}(\sigma)$. There are two cases. If $\sigma \neq \rho$, then $f_{s+1}(\rho) = f_s(\rho)^\frown(1-i)$ so that $f_s(\rho^\frown i) \notin T_{s+1}$. If $\sigma = \rho$, then $f_s(\rho^\frown i)$ extends $f_s(\sigma)$ but is incompatible with $f_{s+1}(\sigma)$ and therefore again is not in $T_{s+1}$.

The functions $f_s$ clearly satisfy (1) and (2). To verify (3), we show simultaneously that, for each $e$ and $\sigma$, action is taken on each requirement $R_{e,\sigma}$ at most finitely often. Fix $e$ and $\sigma$ and choose $s$ by induction so that for all $t \geq s$ and all requirements $R_{d,\tau}$ of higher priority

(7) $f_s(\tau) = f_t(\tau)$ and

(8) no action is ever taken on requirement $R_{d,\tau}$ after stage $s$.

Then $f_s(\tau) = f_t(\tau)$ for all $\tau \prec \sigma$ so that $f_s(\sigma) \preceq f_t(\sigma)$. Thus once requirement $R_{e,\sigma}$ is satisfied at stage $t$, it will remain satisfied. Hence it requires attention at most one more time after stage $s$ and therefore $f_t(\sigma)$ converges.

Finally, suppose that $y \in P$, $\Phi_e(y)$ is total and $\Phi_e(y) \neq y$. Let $y = F(x)$, $\sigma = x\lceil n$ and $\tau = f(\sigma) \prec y$. Since $\Phi_e(y) \neq y$, there must exist $\sigma' \succeq \sigma$ such that $\Phi_e(f(\sigma'))$ is incompatible with $y$. It follows that in fact $\Phi_e(\tau) \notin T$, since otherwise $R_{e,\sigma}$ would require attention at arbitrarily large stages $s$. $\square$

There is a version of this theorem for classes of separating sets.

**Theorem 3.2.11.** *(Jockusch-Soare [90]) There exist disjoint c. e. sets $A$ and $B$ such that $A \cup B$ is cofinite, but any two members of $Sep(A,B)$ either have finite difference or are Turing incomparable.*

*Proof.* The disjoint c. e. sets $A$ and $B$ will be defined by a priority argument as the effective, increasing unions $A = \cup_s A_s$ and $B = \cup B_s$. The construction will ensure that for each partial function $\Phi_e$ and any pair of $C, D$ of separating sets for $A, B$, the following requirement is satisifed.

$\mathbf{R}_e$: If $C \neq D$ (modulo finite difference), then $D \neq \Phi_e(C)$.

As in the construction of a maximal set ([181], p. 188), an increasing sequence $\{m_i^s : i < \omega\}$ will be defined at stage $s$ so that for each $i$, $m_i^0 \leq m_i^1 \leq \dots$ and $m_i = \lim_s m_i^s$ exists. At any stage $s$, $A_s \cup B_s = \omega \setminus \{m_i^s : i \in \omega\}$, so that $\{m_i : i < \omega\}$ is the complement of $A \cup B$.

For fixed $e$, let

$$D_e = \{m_i : i < e\}.$$

Let $C$ and $D$ be two subsets of $D_e$. The subrequirement $R_e^{C,D}$ associated with $C$ and $D$ asserts that for any separating sets $C_1$ and $D_1$ for $A$ and $B$ such that $C_1 \cap D_e = C$ and $D_1 \cap D_e = D$, $D_1 \neq \Phi_e(C_1)$.

This requirement is said to be *satisfied at stage s* if there exists a string $\sigma$ such that, for any $n$,

(1) if $n \in D_e$, then $n \in C \iff \sigma(n) = 1$;

(2) if $n \in Dom(\sigma)$, then $n \in A_s \to \sigma(n) = 1$ and $n \in B_s \to \sigma(n) = 0$.

(3) $Dom(\sigma) \subseteq D_e \cup A_s \cup B_s$

(4) $\Phi_e^s(m, \sigma)$ is defined and $\neq D(m)$ for some $m$.

That is, let (1) through (4) hold and suppose that $C_1$ and $D_1$ are separating sets for $A$ and $B$ such that $C_1 \cap D_e = C$ and $D_1 \cap D_e = D$. Then $\sigma \prec C_1$ so that $\Phi(m, C) \neq D(m)$.

$R_e^{C,D}$ requires attention at stage $s$ if it is not satisfied at stage $s$ and if there exists a string $\sigma$ satisfying (1) and (2) and such that

(5) $\Phi_e^s(m_e^s, \sigma)$ is defined.

Then we also say that $R_e$ requires attention at stage $s$.

The construction proceeds as follows. Initially $A_0 = B_0 = \emptyset$ and $m_e^0 = e$ for all $e$.

**Stage** $s+1$: Choose the requirement $R_e$ of highest priority which requires attention at stage $s$. (If none exists, let $A_{s+1} = A_s$, $B_{s+1} = B_s$ and $m_i^{s+1} = m_i^s$ for all $i$.) Then choose $C$ and $D$ such that $R_e^{C,D}$ requires attention at stage $s$ and let $\sigma$ be given as above. For $m = m_e^s$, put $m \in A_{s+1}$ if $\Phi_e^s(m, \sigma) = 0$ and otherwise put $m \in B_{s+1}$. For $m < |\sigma|$ such that $m \notin D_e$ and $m \neq m_e^s$, put $m \in A_{s+1}$ if $\sigma(m) = 1$ and put $m \in B_{s+1}$ otherwise.

Then the sequence $m_i^{s+1}$ is defined so that $m_i^{s+1}$ is the $i$th element not in $A_s \cup B_s$.

Subrequirement $R_e^{C,D}$ is now satisfied at stage $s+1$, since $\sigma'$ satisfies the conditions (1) through (4) where $\sigma' = \sigma$ except possibly for $\sigma(m_e^s)$.

It is easy to see that each requirement $R_e$ requires attention only finitely often and that $m_e^s$ converges for each $e$. To verify that all requirements are satisfied, let $C'$ and $D'$ be separating classes for $A$ and $B$ which have infinite difference and suppose by way of contradiction that $C' = \Phi_e(D')$. Choose $s_0$ so that every requirement of priority $R_e$ or higher has ceased to require attention by stage $s_0$. Then for $i \leq e$, $m_i^s = m_i$ for all $s \geq s_0$. Let $C = C' \cap D_e$ and $D = D' \cap D_e$. Now at some stage $s > s_0$, $\Phi_e^s(m_e, D)$ must converge so that there is a $\sigma \prec D$ with $\Phi_e^s(m_e, D)$ defined. But then $R_e^{C,D}$ requires attention at stage $s$, contrary to the assumption on $s_0$. This completes the proof. □

The proof of the following theorem is omitted.

**Theorem 3.2.12.** *(Jockusch-Soare [89]) There exist disjoint pairs $(A_0, B_0)$ and $(A_1, B_1)$ of disjoint, recursively inseparable c. e. sets such that if $C \in S(A_0, B_0)$ and $D \in S(A_1, B_1)$, then $C$ and $D$ form a minimal pair, that is, any set computable in both $C$ and $D$ is computable.* □

**Corollary 3.2.13.** *For any degree* $\mathbf{a}$*, there is a special c. b. $\Pi^0_1$ class which has no members of degree* $\geq \mathbf{a}$.

*Proof.* Let $(A_i, B_i)$ be given for $i = 0, 1$ by Theorem 3.2.12. Each class $S(A_i, B_i)$ contains no computable elements. Suppose that $S(A_0, B_0)$ has a member of degree $\geq a$. Then no member of $S(A_1, B_1)$ can have degree $\geq a$. □

**Exercises**

3.2.1. Theorem 3.2.3 can be improved to say that for any degree $\mathbf{c}$ of a member of $P$ and any c. e. degree $\mathbf{a} \geq \mathbf{c}$, $\mathbf{a} = \mathbf{0}'$. Show this using the full Modulus Lemma, which gives a modulus $m \leq_T A$ for $C$ whenever $A$ is a c.e. set such that $C \leq_T A$.

3.2.2. Show that any nonempty c. b. $\Pi^0_1$ class $P$ contains members $x$ and $y$ whose degrees have greatest lower bound $\mathbf{0}$. If $P$ is special, then this gives a minimal pair of members.

3.2.3. Show that if $F$ is a computable function from $\{0,1\}^{\mathbb{N}}$ to $\{0,1\}^{\mathbb{N}}$, $r \in \{0,1\}^{\mathbb{N}}$, and $P$ is a $\Pi^0_1$ class such that $F(x) = r$ for all $x \in P$, then $r$ is computable.

## 3.3 Measure, Category and Randomness

that is, $\{q \in \mathbb{Q} : q \leq r\}$ is a $\Pi^0_1$ set, since $r$ can be approximated from above as the limit of a decreasing computable sequence. That is, given a computable tree $T \subset 2^{<\omega}$ with $P = [T]$, let $P_n = \bigcup\{I(\sigma) : \sigma \in T \cap 2^n\}$. Then $m(P_n)$ is just $k/2^n$, where $k = card\{\sigma \in T \cap 2^n\}$ and $m(P) = \lim_n m(P_n)$. The measure is not necessarily computable, which will follows from the next two results.

The measure of a $\Pi^0_1$ class of reals will be discussed in detail in Chapter 14.

It is easy to modify the class of diagonally non-computable reals to obtain a $\Pi^0_1$ class which has positive measure but has no computable elements.

**Theorem 3.3.1.** *There is a $\Pi^0_1$ class $P \subset \{0,1\}^{\mathbb{N}}$ with positive measure which has no computable members.*

The proof is left as an exercise.

On the other hand, there is a computable basis result.

**Theorem 3.3.2.** *Any $\Pi^0_1$ class $P \subseteq \{0,1\}^{\mathbb{N}}$ with positive, computable measure has a computable member.*

*Proof.* Let $P$ have computable measure $r > 0$. It follows that the measure of $P \cap I(\sigma)$ is computable uniformly in $\sigma$, since we can approximate the measure from below by subtracting the measure of the complement $P \cap (\{0,1\}^{\mathbb{N}} - I(\sigma))$ from $r$. Thus we can recursively select paths $\sigma = (x(0), \ldots, x(n))$ of length $n$ such that $P \cap I((x(0), \ldots, x(n)))$ always has measure $\geq r/2^n$. The infinite path $x$ will be a computable member of $P$. $\square$

**Theorem 3.3.3.** *For any thin $\Pi^0_1$ class $Q \subseteq \{0,1\}^{\mathbb{N}}$, $\mu(Q) = 0$.*

*Proof.* Let $Q$ be a thin class and let $T$ be a computable tree such that $Q = [T]$. Suppose that $\mu(Q) > 0$ and choose a rational $p > 0$ so that $\mu(Q) \geq p$. Uniformly define the class $P_x$ to be $\{y \in Q : (\forall n) y \upharpoonright n \leq_{lex} y \upharpoonright n\}$. Then let $P = \{x \in Q : \mu(P_x) \geq p$. This is a $\Pi^0_1$ class since

$$x \in P \iff (\forall n) card(\{\sigma \in T : \sigma \leq_{lex} x \upharpoonright n) \geq 2^n p.$$

$P$ has the property that, for $x, y \in Q$, if $x \in P$ and $x \leq_{lex} y$, then $y \in Q$. Since $Q$ is thin, the complement $Q - P$ is also a $\Pi^0_1$ class and $\mu(Q - P) = p$. But as a closed set $Q$ has a greatest element $z$. It follows that $Q - P = P_z$ so that $\mu(P_z) = p$, which would mean that $z \in P$, a contradiction. $\square$

**Definition 3.3.4.**

1. *The real $x \in \{0,1\}^{\mathbb{N}}$ is said to be* random *if $x$ does not belong to any $\Pi^0_1$ class of measure 0.*

2. *The real $x \in \{0,1\}^{\mathbb{N}}$ is said to be* 1-random *if for any computable function $f$ such that $\mu(P_{f(n)}) > 1 - 2^{-n}$ for all $n$, $x \in P_{f(n)}$ for some $n$.*

It follows from this definition that the class of random reals is simply the intersection of all $\Sigma^0_1$ classes of measure 1 and therefore has measure 1 itself. Thus *any* set of positive measure must contain a random real.

We will also consider degrees of members of $\Pi^0_1$ classes. For any class $P \subset \mathbb{N}^{\mathbb{N}}$, $\mathcal{D}(P)$ is the collection of all sets Turing equivalent to a member of $P$ and $\mathcal{U}(P)$ is the collection of all sets $A$ such that some member of $P$ is Turing reducible to $A$. Note that $\mathcal{D}(P) \subseteq \mathcal{U}(P)$

P. Martin-Lof [133] introduced the notion of 1-randomness and showed that there is a universal, increasing sequence $P_{e_n}$ of $\Pi^0_1$ classes such that $\mu(P_{e_n}) > 1 - 2^{-n}$ and such that $\cup_n P_{e_n}$ is precisedly the class of 1-random reals. Thus the class of 1-random reals also has measure 1. The degree of a 1-random real is called a *1-random degree*. Recall that a function $f : \omega \to \omega$ is *fixed-point-free* if there is no $e$ such that $\phi_e = \phi_{f(e)}$.

**Theorem 3.3.5.** *(Martin-Lof) There is a computable function $g$ such that if $Q_n = P_{g(n)}$, then*

1. *$Q_0 \subseteq Q_1 \subseteq \ldots$*

2. *For each $n$, $\mu(Q_n) > 1 - 2^{-n}$*

*3.* $\cup_n Q_n$ *is the set of 1-random reals.*

*Proof.* For each $n$, we construct a c. e. set $U_n \subset \{0,1\}^*$ and let $Q_n = \{0,1\}^{\mathbb{N}} \setminus \cup_{\sigma \in U_n} I(\sigma)$. First enumerate all c. e. Martin-Lof tests as follows. Let $W_e$ denote the $e$th c. e. subset of $\{0,1\}^*$ and let $W_{e,s} \subseteq \{0,1\}^s$ denote the finite subset of $W_{e,s}$ enumerated into $W_e$ by stage $s$. Then let $W_{e,j,s} = \bigcap_{i=1}^j W_{\langle e,j \rangle,s}$ as long as $\mu(\bigcup\{I(\sigma) : \sigma \in W_{e,j,s}\}) \leq 2^{-(j+1)}$ and otherwise $W_{e,j,s} = W_{e,j,s-1}$. Then for $\mathcal{W}_{e,j} = \bigcup\{I(\sigma) : s \in \mathbb{N}, \sigma \in W_{e,j,s}\}$. the c. e. Martin-Lof tests consist exactly of the sequences $\mathcal{W}_{e,0}, W_{e,1}, \dots$ for $e \in \mathbb{N}$.

Now define

$$U_n = \cup_{e \in N} W_{e,n+e+1},$$

and let

$$\mathcal{U}_n = \cup\{I(\sigma) : \sigma \in U_n\}.$$

Note that $\mu(\mathcal{U}_n) \leq \sum_{e \in N} \mu(\mathcal{W}_{e,n+e+1}) \leq \sum_e 2^{-(n+e+1)} \leq 2^{-n}$. Then $\{\mathcal{U}\}_{n<\omega}$ is a c. e. Martin-Lof test. Let $Q_n = \{0,1\}^{\mathbb{N}} \setminus \mathcal{U}_n$.

If $x$ is 1-random, then $x \in \cup_n Q_n$ since it passes all c. e. Martin-Lof tests. If $x$ is not 1-random, suppose that it fails the test $W_{e,n_{e<\omega}}$, Then $x \in \cap_{n \geq e+1} \mathcal{W}_{e,n}$ and hence $x \in \cap_n \mathcal{U}_n$. □

**Theorem 3.3.6.** *(Kucera). For any* $\Pi^0_1$ *class* $P$ *of positive measure,* $\mathcal{D}(P)$ *contains every 1-random degree.*

*Proof.* Suppose $\mu(P) > 0$. Let $W$ be a c. e. set such that $\{0,1\}^{\mathbb{N}} \setminus P = \cup_{\sigma \in W} I(\sigma)$. For each $n$, define a sequence of uniformly c. e. sets $U^k$ by $U^0 = W$ and for each $k$,

$$U^{k+1} = \{\tau^\frown\sigma : \tau \in U^k \ \&\ \sigma \in W\}.$$

Then for each $k$, let $V^k = \bigcup_{\sigma \in U^k} I(\sigma)$. It is easy to see that, for all $k$, $V^{k+1} \subseteq V^k$ and that $\mu(V^k) = \mu(V)^k$. Choose $m$ such that $\mu(V^k) < \frac{1}{2}$. Then for each $n$, $\mu(V^{nk}) < 2^{-n}$. Thus $\cap_m V^m$ contains no 1-random elements. Suppose now that $A$ is 1-random. It follows that $A \notin V^j$ for some $j$; let $k$ be the least such. If $k = 0$, then $A \in P$. If $k > 0$, then $A \in V^{k-1} \setminus V^k$, so that $A = \tau^\frown B$ for some $\tau \in U^{k-1}$ and some $B \in P$. □

It follows from this proof that for any $\Pi^0_1$ class $P$ of positive measure and any random real $A$, the iterated shift $\sigma^n(A) \in P$ for some $n$.

**Corollary 3.3.7.** *For each* $n$*,* $\mathcal{D}(Q_n)$ *equals the set of 1-random degrees.*

*Proof.* By Theorem 3.3.5, every element of $Q_n$ is 1-random and by Theorem 3.3.6, $\mathcal{D}(Q_n)$ contains all 1-random degrees. □

This also proves the following result from [91], which significantly improves Theorem 3.3.1, since now we know that $\mathcal{D}(Q_n)$ includes the 1-random degrees and thus has measure one.

**Corollary 3.3.8.** *(Jockusch-Soare) There is a c. b.* $\Pi^0_1$ *class* $P$ *with no computable members such that* $\mu(\mathcal{D}(P)) = 1$*.*

The following is due to Kucera [106].

**Theorem 3.3.9.** *Every 1-random set is bi-effectively immune.*

*Proof.* Let $Q_n$ be defined as in Theorem 3.3.5 above. Following Kucera's proof, we will construct computable functions $f$ and $g$ such that for all $k, x \in \mathbb{N}$:

$$A \in Q_n \ \& \ W_x \subseteq A \implies card(W_x) \leq f(n,x);$$
$$A \in Q_n \ \& \ W_x \subseteq \mathbb{N} - A \implies card(W_x) \leq g(n,x).$$

Now define the uniformly $\Pi^0_1$ classes $P_{x,n} \subseteq \{0,1\}^{\mathbb{N}}$ by

$$A \in P_{x,m} \iff [A \text{ contains the first } m+1 \text{ elements of } W_x \to card(W_x) < m.$$

Then $\mu(P_{x,m}) > 1 - 2^{-m}$. Now let $e = f(n,x)$ be defined so that $\phi_e(j)$ is an index of $P_{x,j}$ for all $j \in \mathbb{N}$. It follows from the definition of $Q_e$ that $Q_e \subseteq P_{x,e}$, so that $Q_k \subseteq P_{x,e}$. Thus if $A \in Q_k$ & $W_x \subseteq A$, then $card(W_x) \leq e$. The argument for $\mathbb{N} - A$ and $g$ is similar. □

The next two results are taken from [91].

**Theorem 3.3.10.** *For any closed subset $P$ of $\mathbb{N}^{\mathbb{N}}$ with no computable member, $\mathcal{U}(P)$ (and hence $\mathcal{D}(P)$) is meager.*

*Proof.* Let $P = [T]$ where $T$ is not necessarily computable and assume by way of contradiction that $\mathcal{U}(P)$ is not meager. Then for some $e$, $S_e = \{A : \phi_e^A \in P\}$ is not meager. Now $S_e$ is not nowhere dense, so there exists $\sigma$ such that every extension $\tau$ of $\sigma$ can be extended to some $A \in S_e$. Note that $S_e$ is not necessarily a closed set. We will use $e$ and $\sigma$ to construct a computable member of $P$. By assumption there is an extension $A$ of $\sigma$ in $S_e$ so that $\phi_e^A \in P$. Thus we can find $\tau_0 \succ \sigma$ such that $\phi_e(\tau_0, 0)$ is defined. Proceeding recursively, we can find $\tau_{n+1} \succ \tau_n$ such that $\phi_e(\tau_n) \in T$ and $|\phi_e(\tau_n)| > n$. It follows that the computable function $f(n) = \phi_e(\tau_n, n) \in P$. (Although $x \cup_n \tau_n$ is not necessarily an element of $S_e$.) □

This theorem implies the result of Sacks [167] that $\mathcal{U}(\{A\}$ is meager for any noncomputable set $A$ and the following theorem from [91] will imply the result from [167] that $\mu(\mathcal{U}(\{A\})) = 0$ for noncomputable $A$.

**Theorem 3.3.11.** *(Jockusch-Soare) For any $\Pi^0_1$ class $P = S[A,B]$ where $A$ and $B$ are computably inseparable, $\mu(\mathcal{U}(P)) = 0$.*

*Proof.* Suppose by way of contradiction that $\mu(\mathcal{U}(P)) > 0$. Then for some $e$ and some rational $m$, $Q_e = \{C : \phi_e^C \in P\}$ has positive measure $p$ with $m < p < 3m/2$ for some rational $m$. Let $Q_e = [T]$ for some computable tree $T$. For $i = 0, 1$, let

$$C_i = \{n : \mu(\{C : \phi_e^C(n) = i\}) > m/2\}.$$

Then each $C_i$ is a c. e. set, since to test whether $n \in C_i$, simply find $k$ such that $card(\{\sigma \in \{0,1\}^k : \phi_e^\sigma(n) = i\}) > m2^{k-1}$. Since $\phi_e^C$ is total for a set of measure

$> m$, it follows that $C_0 \cup C_1 = \mathbb{N}$. Hence by the reduction principle, there are disjoint c. e. sets $E_0 \subseteq C_0$ and $E_1 \subseteq C_1$ such that $E_0 \cup E_1 = \mathbb{N}$ and therefore $E_0$ and $E_1$ are computable. We claim that $E_1$ is a separating set for $A$ and $B$. If $n \in A$, then $\mu(\{C : \phi_e^C \in P\}) > m$, so that $\mu(\{C : \phi_e^C(n) = 1\}) > m$ and hence $\mu(\{C : \phi_e^C(n) = 0\}) < m/2$, so that $n \notin C_0$ and therefore $n \in E_1$. Similarly if $n \in B$, then $n \in E_0$. But this contradicts the assumption that $A$ and $B$ are computably inseparable. □

**Exercises**

3.3.1. Construct a $\Pi^0_1$ class which has positive measure $m$ but has no computable members. Show that for any real number $\epsilon > 0$, we can make $m > 1 - \epsilon$. (Hint: we can ensure that $\phi_e \notin P$ by making $\phi_e \lceil n \notin T$ for some large value of $n$. )

3.3.2. Say that a real $r$ is $\Sigma^0_1$ if $\{q \in \mathbb{Q} : q < r\}$ is a $\Sigma^0_1$ set. Show that $r$ is $\Pi^0_1$ (respectively, $\Sigma^0_1$) if and only if $r$ is the limit of a computable, decreasing (resp. increasing) sequence of rationals. Thus $r$ is computable if and only if $r$ is both an increasing and a decreasing limit of rationals.

3.3.3. Show that the class of 1-generic reals has measure 0 and is not a basis for the family of $\Pi^0_1$ classes of positive measure. (Hint: $x$ is 1-generic if it never belongs to the boundary of any $P_e$.)

3.3.4. Show that 1-generic reals and 1-random reals are also random.

3.3.5. Show that if a $\Pi^0_1$ class $P$ contains a random element, then $\mu(P) > 0$.

## 3.4 Mathematical Logic: Peano Arithmetic

In this section, we consider further the connection between $\Pi^0_1$ classes and mathematical logic and, in particular, Peano Arithmetic. We begin with some applications of the present chapter together with sections 2.2.9.

**Theorem 3.4.1.** *Any axiomatizable theory $\Gamma$ has a complete consistent extension of low c. e. degree.*

*Proof.* Let $\Gamma$ be an axiomatizable theory and let the $\Pi^0_1$ class $P$ represent the family of complete consistent extensions of $\Gamma$, by Theorem 2.2.9.1. Then $P$ has a member of low c. e. degree by Theorem 3.1.4. □

On the other hand, we have the following.

**Theorem 3.4.2.** *There is a (propositional) axiomatizable theory $\Gamma$ which has no c. e. complete consistent extension.*

*Proof.* By Theorem 3.2.1, there is a $\Pi^0_1$ class $P$ with no c. e. member. Now, by Theorem 2.2.9.3, there is an axiomatizable theory $\Gamma$ such that $P$ represents the set of complete consistent extensions of $\Gamma$. □

**Theorem 3.4.3.** *For any c. e. degree* $\mathbf{c}$*, there is an axiomatizable theory* $\Gamma$ *such that the c. e. degrees of complete consistent extensions of* $\Gamma$ *are exactly the c. e. degrees above* $\mathbf{c}$*.*

*Proof.* This is an immediate consequence of Theorems 2.2.9.3 and 3.2.4. □

**Theorem 3.4.4.** *For any essentially undecidable, axiomatizable theory* $\Gamma$*, there exists a c. e. degree* $\mathbf{a}$ *such that* $\Gamma$ *has no complete consistent extensions of degree* $\leq \mathbf{a}$*.*

*Proof.* This follows from Theorems 2.2.9.1 and 3.2.5. □

**Theorem 3.4.5.** *For any essentially undecidable, axiomatizable theory* $\Gamma$*, there exists two complete consistent extensions,* $\Delta_1$ *and* $\Delta_2$*, of* $\Gamma$ *such that any set computable from* $\Delta_1$ *and computable from* $\Delta_2$ *is in fact computable.*

*Proof.* This follows from Theorems 2.2.9.1 and 3.2.12. □

**Theorem 3.4.6.** *There is an axiomatizable theory* $\Gamma$ *such that any two complete consistent extensions of* $\Gamma$ *are Turing incomparable.*

*Proof.* This follows from Theorems 2.2.9.3 and 3.2.10. □

### 3.4.1 Peano Arithmetic

The standard model $\mathbb{N} = (\mathbb{N}, S, +, \cdot, <)$ of arithmetic is fundamental in mathematics. The language of arithmetic is often defined to consist of a one-place function symbol $S$, representing the successor function, and binary function symbols $+$ for addition and $\cdot$ for multiplication. The usual linear ordering $\leq$ may then be defined by

$$\begin{aligned} x \leq y &\iff (\exists z)(x + z = y); \\ x < y &\iff x \leq y \ \& \ x \neq y. \end{aligned}$$

Each natural number $n$ may be represented by the term $S^n 0$. We will generally identify $S^n 0$ with $n$ for simplicity of expression. Peano Arithmetic is a first order theory for $\mathbb{N}$ the consisting of nine axioms needed to define the functions and also the axiom scheme of induction. The eight axioms of *Robinson* arithmetic are the following.

**S1** $Sx = Sy \implies x = y$

**S2** $(\forall x) Sx \neq 0$

**L1** $(\forall x) \neg x < 0$

**L2** $(\forall x)(\forall y)[x < Sy \iff (x < y \vee x = y)]$

**A1** $(\forall x) x + 0 = x$

**A2** $(\forall x)(\forall y) x + Sy = S(x + y)$

**M1** $(\forall x)x \cdot 0 = 0$

**M2** $(\forall x)(\forall y)x \cdot Sy = x \cdot y = y$

The axiom scheme of induction provides for each formula $\phi$ with one free variable:

**IP** $[\phi(0) \;\&\; (\forall x)(\phi(x) \;\rightarrow\; \phi(Sx))] \rightarrow (\forall y)\phi(y)$

The fundamental results of Gödel and others [92, 192] connecting Peano Arithmetic and computability theory is the following.

**Theorem 3.4.7.** *For any c. e. set $A$, there is a formula $\phi$ (which represents $A$) such that, for all $m$, the following are equivalent:*

*(1)* $m \in A$;

*(2)* $\mathbb{N} \models \phi(m)$ *and*

*(3)* $PA \vdash \phi(S^m 0)$.

A partial converse to this result is the result of Gödel that *any* axiomatizable theory $\Gamma$ is computably enumerable (in the sense that the set of Gödel numbers of members of $\Gamma$ is a c. e. subset of $\mathbb{N}$). This means that in particular $PA$ itself is a c. e. set. The Incompleteness Theorem of Gödel follows from these results and tells us in particular, that $PA$ can have no axiomatizable, complete consistent extension.

We shall now consider the connection between Peano Arithmetic and $\Pi^0_1$ classes. It follows from Theorem 2.9.1 that the set of complete consistent extensions of $PA$ as well as the set of consistent extensions of $PA$ may be represented as $\Pi^0_1$ classes. It then follows that there is a complete consistent extension of $PA$ with c. e. degree. Now any axiomatizable extension $\Gamma$ of $PA$ is necessarily incomplete by Gödel's theorem, hence the family of complete consistent extensions of $\Gamma$ will be an uncountable $\Pi^0_1$ class. Certainly not every $\Pi^0_1$ class may be represented as the set of complete extensions of such a $\Gamma$. However, the Scott Basis Theorem shows that these classes are as complicated as an arbitrary $\Pi^0_1$ class in a certain sense.

**Theorem 3.4.8.** *(Scott [171]) For any consistent extension $\Gamma$ of $PA$, the sets computable in $\Gamma$ form a basis for the c. b. $\Pi^0_1$ classes.*

*Proof.* Let $\Gamma$ be a consistent extension of $PA$, let $P$ be a c. b. $\Pi^0_1$ class and let $T_P \subseteq \mathbb{N}^*$ be the tree of nodes which have an extension in $P$. Then $T_P$ is a $\Pi^0_1$ set and it follows from Theorem 3.4.7 that there is a formula $\psi$ such that, for all strings $\sigma$,

$$\sigma \notin T_P \implies PA \vdash \phi(\langle\sigma\rangle) \implies \phi(\langle\sigma\rangle \in \Gamma.$$

Now we can define a subtree $T$ of $T_P$ which is computable in $\Gamma$ by

$$\sigma \in T \iff \phi(\langle\sigma\rangle \notin \Gamma.$$

The tree $T$ has no dead ends and thus there $P$ contains an infinite path which is computable in $T$ and hence computable in $\Gamma$. $\square$

Tennenbaum [193] observed that no nonstandard model of $PA$ can be computable and the following improvement of that result was noted by Jockusch and Soare [90].

**Theorem 3.4.9.** *If $\mathcal{M} = (\mathbb{N}, +^M, \cdot^M)$ is any nonstandard model of Peano arithmetic, then the sets computable in $M$ form a basis for the c. b. $\Pi^0_1$ classes.*

*Proof.* We follow the proof of Cohen [49]. Since $PA$ and the set of negations of sentences in $PA$ form a pair of computabley inseparable c. e. sets, it suffices by Theorem 3.4.8 to show that $\mathcal{M}$ can compute a separating set for any pair of disjoint c. e. sets $A$ and $B$. First observe that we can recursively compute the standard numbers in $\mathcal{M}$ by starting with $a_0$ and $a_1$ which represent 0 and 1 and letting $a_{n+1} = a_n +^M a_1$ for all $n$. Similarly we may compute from $\mathcal{M}$ the sequence $p_n$ of (finite) prime numbers of $\mathcal{M}$. Now by the Chinese Remainder Theorem, there must exist an infinite element $a$ of $\mathcal{M}$ such that, in $\mathcal{M}$, $a = 0(mod p_n)$ if and only if $a \in A$ and $a = 1(mod p_n)$ if and only if $n \in B$. A separating set for $A$ and $B$ may now be computed from $\mathcal{M}$ as follows. Given $n$, compute the unique $q$ and unique $r < p_n$ such that $a = q \cdot^M p_n +^M r$ and put $n \in C$ if and only if $r \neq 1$. It is immediate that $A \subset C$ and $B \cap C = \emptyset$. □

We have the following corollary to Theorem 3.2.3.

**Corollary 3.4.10.** *If $\mathbf{a}$ is the degree of any consistent extension or nonstandard model of Peano Arithmetic and $\mathbf{b}$ is a c. e. degree such that $\mathbf{a} \leq b$, then $\mathbf{b} = \mathbf{0}'$.*

It now follows that no consistent extension of $PA$ can have c. e. degree $< \mathbf{0}'$. However, can get degree exactly $\mathbf{0}'$, as announced by Scott and Tennenbaum in [172].

**Corollary 3.4.11.** *There is a complete consistent extension of Peano Arithmetic of degree $\mathbf{0}'$.*

*Proof.* Let $P$ be the c. b. $\Pi^0_1$ class of all complete consistent extensions of $PA$. Then $P$ has a member of c. e. degree $\mathbf{b}$ and it follows from Corollary 3.4.10 that $\mathbf{b} = \mathbf{0}'$. □

Here is a nice corollary to Theorem 3.2.8.

**Corollary 3.4.12.** *(Jockusch-Soare [91]) There is a complete consistent extension $\Gamma$ of Peano Arithmetic such that every set definable in $\Gamma$ is either computable or not arithmetical.*

*Proof.* By Theorem 3.2.8, there is a theory $\Gamma$ whose degree is incomparable with $\mathbf{0}^{(n)}$ for each $n > 0$. The corollary now follows from the fact that each set definable in $\Gamma$ is computable from $T$. □

# Chapter 4

# The Cantor-Bendixson Derivative

The perfect set theorem states that any closed subset of $\{0,1\}^{\mathbb{N}}$ is the union of a perfect closed set (the perfect kernel) and a countable set. The perfect kernel results from iterating the Cantor Bendixson derivative $D^{\alpha}(P)$ until a fixed point (an analytic set) is reached. The effective version of this theorem is that the perfect kernel of a $\Pi^0_1$ class is a $\Sigma^1_1$ and that the iteration must stop by the first non-computable ordinal $\omega_1^{C-K}$.

For a countable $\Pi^0_1$ class $P$, the perfect kernel is the empty set and the iteration must stop at some computable ordinal (the C-B rank of $P$. The Kreisel basis theorem [105] showed that isolated members of $\Pi^0_1$ classes must be hyperarithmetic in general, must be computable in $\mathbf{0}'$ if $P$ is bounded, and must be computable if $P$ is computably bounded. Furthermore, any countable closed set must have an isolated element.

A finer analysis was given in [19] for $\Pi^0_1$ classes in $\{0,1\}^{\mathbb{N}}$. The C-B rank of a real $x$ in a class $P$ is the least ordinal $\alpha$ such that $x \notin D^{\alpha}(P)$. In particular, if $x$ has C-B rank $\lambda + n$ in $P$ for some limit ordinal $\lambda$ and finite $n$, then $x$ is computable in $\mathbf{0}^{\lambda+2n}$. It follows that every element of a countable class is hyperarithmetic. The C-B rank of a real $x$ is defined to be the minimum of the ranks of $x$ in any $\Pi^0_1$ class.

## 4.1 Cantor-Bendixson derivative and rank

The Cantor-Bendixson derivative $D(P)$ of a compact subset $P$ of $\mathbb{N}^{\mathbb{N}}$ is the set of nonisolated points of $P$. Thus a point $x \in P$ is not in $D(P)$ if and only if there is some open set $U$ containing $x$ which contains no other point of $P$. Equivalently, $x \notin D(P)$ if and only if there is some clopen set $U$ such that $U \cap P = \{x\}$. Another useful observation is that, for any compact set $P$, $D(P)$ is empty if and only if $P$ is finite.

The iterated Cantor-Bendixson derivative $D^\alpha(P)$ of a closed set $P$ is defined for all ordinals $\alpha$ by the following transfinite induction.

**Definition 4.1.1.** *1. $D^0(P) = P$; $D^{\alpha+1}(P) = D(D^\alpha(P))$ for any $\alpha$;*

*2. $D^\lambda(P) = \bigcap_{\alpha<\lambda} D^\alpha(P)$ for any limit ordinal $\lambda$.*

There is a related derivative which we can define for a tree $T \subseteq \{0,1\}^*$.

**Definition 4.1.2.** *1. $d(T) = \{\sigma \in T : (\exists\tau)[\sigma \prec \tau \ \&\ \tau^\frown 0 \in Ext(T) \ \&\ \tau^\frown 1 \in Ext(T)]\}$.*

*2. $d^0(T) = T$, $d^{\alpha+1}(T) = d(d^\alpha(T))$ and $d^\lambda(T) = \cap_{\alpha<\lambda} d^\alpha(T)$ for limit $\lambda$.*

Then $d(T) \leq_T T^{(2)}$ and is in fact $\Sigma^0_2$ in $T$. We iterate this derivative by

**Lemma 4.1.3.** *For any countable ordinal $\alpha$ and any tree $T \subseteq \{0,1\}^*$, $D^\alpha([T]) = [d^\alpha(T)]$.*

*Proof.* The proof is by transfinite induction. The equality is clear for $\alpha = 0$.

The interesting case is when $\alpha = 1$. Suppose that $x \in [d(T)]$ and fix $n \in \omega$. Then $x\lceil n \in d(T)$, so that there exists $\tau \succ x\lceil n$ such that both $\tau^\frown 0$ and $\tau^\frown 1$ are in $Ext(T)$. Thus $I(x\lceil n) \cap [T]$ contains at least two elements. It follows that there is some path $y_n \in [T]$ such that $x\lceil n \prec y_n$ and $x \neq y_n$. Hence $x \in D([T])$.

Next suppose that $x \in D([T])$ and again fix $n \in \omega$. Since $x$ is not isolated in $[T]$, there must exist $y \in [T]$ such that $y \neq x$ and $x\lceil n \prec y$. Let $m > n$ be the least such that $y\lceil m+1 \neq x\lceil m+1$ and let $\tau = x\lceil m$. Then $x\lceil n \prec \tau$ and $\tau^\frown 0$ and $\tau^\frown 1$ are both in $Ext(T)$, which demonstrates that $x\lceil n \in d(T)$. Hence $x \in [d(T)]$. v Now suppose that $D^\alpha([T]) = [d^\alpha(T)]$. Then

$$D^{\alpha+1}([T]) = D(D^\alpha([T]) = D([d^\alpha(T)]) = [d([d^\alpha(T))] = [d^{\alpha+1}(T)].$$

For a limit ordinal $\lambda$,

$$D^\lambda([T]) = \cap_{\alpha<\lambda} D^\alpha([T]) = \cap_{\alpha<\lambda}[d^\alpha(T)] = [d^\lambda(T)].$$

This completes the proof. □

The Cantor-Bendixson ($CB$) rank of a closed set $P$ is the least ordinal $\alpha$ such that $D^{\alpha+1}(P) = D^\alpha(P)$; then $D^\alpha(P)$ is a perfect closed set, denoted $K(P)$, called the *perfect kernel* of $P$ and $P \setminus K(P)$ is a countable set. For $A \in P \setminus K(P)$, the C-B rank $rk_P(A)$ of $A$ in $P$ is the least ordinal $\alpha$ such that $A \in D^\alpha(P) \setminus D^{\alpha+1}(P)$; the C-B rank $rk(P)$ is the least $\alpha$ such that $D^\alpha(P) = K(P)$. The set $A$ is *ranked* if there is a $\Pi^0_1$ class $P$ such that $A \in P \setminus K(P)$, and the C-B rank $rk(A)$ is the least $\alpha$ such that $rk_P(A) = \alpha$ for some $\Pi^0_1$ class $P$. Kreisel [105] used the Boundedness Principle of Spector [184] to show that $rk(P) \leq \omega_1^{\mathrm{C-K}}$ for any $\Pi^0_1$ class $P$, so that any ranked point $A$ has $rk(A) < \omega_1^{\mathrm{C-K}}$.

**Theorem 4.1.4** (Kreisel)**.** *For any $\Pi^0_1$ class $P$, $K(P)$ is a $\Sigma^1_1$ class and $rk(P) \leq \omega_1^{C-K}$.*

*Proof.* Fix a computable tree $T$ such that $P = [T]$ and define the $\Sigma^0_2$ monotone inductive operator $\Gamma$ on $\{0,1\}^*$ as follows.

$$\sigma \in \Gamma(A) \iff \sigma \notin T \ \vee \ \sigma \notin d(\{0,1\}^* - A).$$

It follows that $\Gamma^1 = \{0,1\}^* - T$, $\Gamma^1 = \{0,1\}^* - d(T)$, and so on, so that $Cl(\Gamma) = \{0,1\}^* - T_{K(P)}$. It follows from Theorem II.1.13.8 that $T_{K(P)}$ is $\Sigma^1_1$, so that $K(P) = [T_{K(P)}]$ is also $\Sigma^1_1$. It follows from Corollary 1.13.13 that $rk(P) \leq \omega_1$. $\square$

We will show in Section VI.5.7 that there is a $\Pi^0_1$ class $P$ such that $rk(P) = \omega_1^{C-K}$ and $K(P)$ is not $\Delta^1_1$.

For the sake of completeness, let $rk_P(A) = rk(P)$ if $A \in K(P)$. This means that the rank function will define a prewellordering on $P$.

A fundamental idea here is that the complexity of an element $x$ of a $\Pi^0_1$ class $P$ is related to the Cantor-Bendixson rank of $x$ in $P$. Kreisel [105] first noticed that the Turing degree of a member $x$ of a $\Pi^0_1$ class is related to the $CB$ rank; he used this to show that every member of a countable $\Pi^0_1$ class is hyperarithmetic. In particular, it is easy to see that a real has rank 0 if and only if it is computable.

We need a series of lemmas from [42]. Note first that for two closed sets $P$ and $Q$, $D(P \cup Q) = D(P) \cup D(Q)$ but $D(P \cap Q)$ is in general only a subset of $D(P) \cap D(Q)$.

More generally, we have

**Lemma 4.1.5.** *For any closed sets $P$ and $Q$ and any ordinal $\alpha$:*

*(a)* $D^\alpha(P \cup Q) = D^\alpha(P) \cup D^\alpha(Q)$ *and*

*(b)* $D^\alpha(P \cap Q) \subset D^\alpha(P) \cap D^\alpha(Q)$.

*Proof.* The proofs are by induction on $\alpha$. For $\alpha = 0$, these are trivial.

(a) For $\alpha = 1$, $D(P)$ and $D(Q)$ are subsets of $D(P \cup Q)$ since $P$ and $Q$ are subsets of $P \cup Q$. For the reverse inclusion, suppose that $x \notin D(P) \cup D(Q)$. Then there are clopen sets $U$ and $V$ with $U \cap P = \{x\}$ and $V \cap Q = \{x\}$. Thus $U \cap V \cap (P \cup Q) = \{x\}$, so that $x \notin D(P \cup Q)$. Now assume that $D^\alpha(P \cup Q) = D^\alpha(P) \cup D^\alpha(Q)$. Then using the case $\alpha = 1$, we have

$$\begin{aligned} D^{\alpha+1}(P \cup Q) &= D(D^\alpha(P \cup Q)) = D(D^\alpha(P) \ \cup \ D^\alpha(Q)) \\ &= D(D^\alpha(P)) \ \cup \ D(D^\alpha(Q)) = D^{\alpha+1}(P) \ \cup \ D^{\alpha+1}(Q). \end{aligned}$$

(b) This is left as an exercise. $\square$

Note that equality holds in (b) if one of the sets is clopen.

**Lemma 4.1.6.** *For any compact subset $Q$ of $\{0,1\}^\omega$, any clopen set $K$ and any ordinal $\alpha$,*

*(a)* $D^\alpha(K \cap Q) = K \cap D^\alpha(Q)$.

*(b)* $rk_{K\cap Q}(A) = rk_Q(A)$ *for any* $A$.

The proof is left as an exercise.

**Lemma 4.1.7.** *For any set $A$ and any computable ordinal $\alpha$, $rk(A) \leq \alpha$ if and only if there is some $\Pi^0_1$ class $P$ such that $D^\alpha(P) = \{A\}$.*

*Proof.* The "if" direction is immediate from the definition of rank. Suppose now that $rk(A) \leq \alpha$, so that $A \in D^\alpha(Q) \setminus D^{\alpha+1}(Q)$ for some $\Pi^0_1$ class $Q$. Thus $A$ is isolated in $D^{\alpha(Q)}$, so that for some clopen $K$, $K \cap D^\alpha(Q) = \{A\}$. Let $P = K \cap Q$. It follows from Lemma 4.1.6 that
$D^\alpha(P) = D^\alpha(K \cap Q) = K \cap D^\alpha(Q) = \{A\}$. □

**Lemma 4.1.8.** *(Cenzer-Smith [42])*

*(a) Let $\Phi$ be a continuous map from $\{0,1\}^\omega$ into $\{0,1\}^\omega$ and let $P, Q$ be compact sets such that $\Phi(P) = Q$. Then for any $y \in Q$, $rk_Q(y) \leq max\{rk_P(x) : x \in P \ \&\ \Phi(x) = y\}$.*

*(b) For any sets $A, B$, if $A \leq_{tt} B$ and $B$ is ranked, then $A$ is ranked and $rk(A) \leq rk(B)$.*

*(c) For any sets $A, B$, if $A \equiv_{tt} B$ and $B$ is ranked, then $A$ is ranked and $rk(A) = rk(B)$.*

*Proof.* Let $rk_Q(y) = \beta$, so that $y \in D^\beta(Q) \setminus D^{\beta+1}(Q)$. It is shown in Lemma 1.2 of [42] that $D^\alpha(\Phi(P)) \subset \Phi(D^\alpha(P))$ for any $\alpha$. Since $y \in D^\beta(Q)$, it follows that $y = \Phi(x)$ for some $x \in D^\beta(P)$, so that $rk_Q(y) = \beta \leq rk_P(x)$, where it is possible that $x$ is not ranked in $P$.

(b) By Theorem 1.9.10, if $A \leq_{tt} B$, then there is a computable function $\Phi : \{0,1\}^\omega \to \{0,1\}^\omega$ such that $\Phi(B) = A$. Now let $rk(B) = \alpha$ and, by Lemma 4.2, let $P$ be a $\Pi^0_1$ class such that $D^\alpha(P) = \{B\}$, so that $rk_P(B) = max\{rk_P(x) : x \in P\}$, and let $Q = \Phi(P)$. It follows from (a) that $rk(A) \leq rk_Q(A) \leq rk_P(B) = rk(B)$.

(c) This is immediate from (b). □

The following improvement of Lemma 1.4 of [42] is due to J. Owings and C. Laskowski [153]. Recall that $\alpha \oplus \beta$ is the Hessenberg sum of two ordinals, that $A \oplus B$ is the disjoint union of two sets and that $P \oplus Q = \{A \oplus B : A \in P \ \&\ B \in Q\}$ for two classes $P$ and $Q$ of sets.

**Theorem 4.1.9.** *(Owings, [153]) For any sets $A, B \in \{0,1\}^{\mathbb{N}}$ and any compact $P, Q \subset \{0,1\}^{\mathbb{N}}$, $rk_{P\oplus Q}(A \oplus B) = rk_P(A) \oplus rk_Q(B)$.*

*Proof.* We first show by induction on $rk_P(A) \oplus rk_Q(B)$ that $rk_{P\oplus Q}(x \oplus y) \leq rk_P(A) \oplus rk_Q(B)$. If $rk_P(A) \oplus rk_Q(B) = 0$, then $A$ is isolated in $P$ and $B$ is isolated in $Q$, so that there are open intervals $I$ and $J$ such that $P \cap I = \{A\}$ and $Q \cap J = \{B\}$. It follows that $(P \oplus Q) \cap (I \oplus J) = \{A \oplus B\}$, so that $rk_{P\oplus Q}(A \oplus B) = 0$. Now let $rk_P(A) = \alpha$ and $rk_Q(B) = \beta$ and suppose the inequality holds for all $x, y$ such that $rk_P(x) \oplus rk_Q(y) < \alpha \oplus \beta$. By intersecting with open intervals, as above, we may assume that $D^\alpha(P) = \{A\}$ and that $D^\beta(Q) = \{B\}$. It suffices to show that $rk_{P\oplus Q}(x\oplus y) < \alpha\oplus\beta$ for all $x\oplus y \neq A\oplus B$ in $P \oplus Q$. But if $x \oplus y \neq A \oplus B$, then either $x \neq A$ or $y \neq B$, so that either $rk_P(x) < \alpha$ or $rk_Q(y) < \beta$. In either case, $rk_P(x) \oplus rk_Q(y) < \alpha \oplus \beta$, so that $rk_{P\oplus Q}(x \oplus y) < \alpha + \beta$.

For the reverse inequality, we prove by induction on $\alpha \oplus \beta$ that

$D^\alpha(P) \oplus D^\beta(Q) \subset D^{\alpha\oplus\beta}(P \oplus Q)$.

For $\alpha \oplus \beta = 0$, this is obvious. We also need the case where $\alpha \oplus \beta = 1$. Suppose without loss of generality that $\alpha = 1$ and $\beta = 0$ and suppose that $A \in D(P)$ and $B \in Q$. Then for any interval $I \subset \{0,1\}^\omega$, there is some $A' \neq A$ in $P \cap I$. Then for any basic open set $I \oplus J \in \{0,1\}^\omega \oplus \{0,1\}^\omega$, there is an element $A' \oplus B \neq A \oplus B$ in $(P \oplus Q) \cap (I \oplus J)$. Thus $A \oplus B \in D(P \oplus Q)$. This shows that

$D(P) \oplus Q \subset D(P \oplus Q)$.

Now suppose the inclusion holds for all ordinals $\sigma, \tau$ with $\sigma \oplus \tau < \alpha \oplus \beta$. There are two cases.

(Case 1) If $\alpha \oplus \beta$ is a limit ordinal, then $\alpha$ and $\beta$ are both limit ordinals and $\alpha \oplus \beta = sup\{\sigma \oplus \tau : \sigma < \alpha \ \& \ \tau < \beta\}$. Thus

$$\begin{aligned} D^{\alpha\oplus\beta}(P \oplus Q) &= \cap_{\gamma<\alpha\oplus\beta} D^\gamma(P \oplus Q) = \cap_{\sigma<\alpha,\tau<\beta} D^{\sigma\oplus\tau}(P \oplus Q) \\ &= \cap_{\sigma<\alpha,\tau<\beta} D^\sigma(P) \oplus D^\tau(Q) = D^\alpha(P) \oplus D^\beta(Q). \end{aligned}$$

(Case 2) If $\alpha\oplus\beta$ is a successor, then either $\alpha$ is a successor or $\beta$ is a successor–without loss of generality say that $\alpha = \gamma + 1$, so that $\alpha \oplus \beta = (\gamma \oplus \beta) + 1$. Then

$$\begin{aligned} D^\alpha(P) \oplus D^\beta(Q) &= D(D^\gamma(P)) \oplus D^\beta(Q) \subset D(D^\gamma(P) \oplus D^\beta(Q)) \\ &\subset D(D^{\gamma\oplus\beta}(P \oplus Q)) = D^{\alpha\oplus\beta}(P \oplus Q). \end{aligned}$$

This completes the proof. □

Thus we have the following corollary.

**Theorem 4.1.10.** *For any sets $A$ and $B$, $max\{rk(A), rk(B)\} \leq rk(A \oplus B) \leq rk(A) \oplus rk(B)$.*

*Proof.* The first inequality follows from Lemma 4.1.8, since both $A$ and $B$ are $\leq_{tt} A \oplus B$. The second inequality follows from Theorem 4.1.9. □

The basic result for rank 0 is the following.

**Lemma 4.1.11.** *For any $x \in \{0,1\}^\omega$, the following are equivalent:*

(a) *$x$ is computable;*

(b) *$\{x\}$ is a $\Pi^0_1$ class;*

(c) *$x$ has Cantor-Bendixson rank 0.*

*Proof.* Suppose first that $x$ is computable. Then $\{x\} = [T]$, where $\sigma \in T \iff (\forall i < |\sigma|)(\sigma(i) = x(i))$. Next suppose that $\{x\}$ is a $\Pi^0_1$ class. Then the rank of $x$ in $\{x\}$ is 0, so that the C-B rank of $x$ is 0. Next suppose that $x$ has C-B rank 0 and let $P = [T]$ be a $\Pi^0_1$ class such that $x$ is isolated in $P$, where $T$ is a computable tree. Then for each sufficiently large $n$, $x\lceil n+1$ is the unique path of length $n$ which has an extension in $P$. Thus we may compute $x\lceil n+1$ (and therefore compute $x(n)$) by searching for the least $m$ such that all strings $\sigma \in T$ of length $m$ have the same initial segment $\sigma\lceil n$. □

We remark that Lemma 4.1.11 and its proof can be relativized to computability in $B$ for any set $B$.

**Exercises**

4.1.1. Give a proof of Lemma 4.1.5(b).

4.1.2. Give a proof for Lemma 4.1.6.

4.1.3. Prove a relativized version of Lemma 4.1.11.

4.1.4. Show by induction on ordinals $\alpha$ that for any continuous map $\Phi : \{0,1\}^{\mathbb{N}} \to \{0,1\}^{\mathbb{N}}$, and any compact set $Q$, $D^\alpha(\Phi(P)) \subseteq \Phi(D^\alpha(P))$.

4.1.5. Show that if $rk(B) = rk(A \oplus B)$, then $A$ is computable in $B$.

4.1.6. Owings [153] defined a Cantor singleton as being the unique noncomputable element of some $\Pi^0_1$ class. By Theorem 2.6.2, every noncomputable $\Pi^0_1$ retraceable set is a Cantor singleton. Show that any noncomputable set which is the union of a computable set with a $\Pi^0_1$ retraceable set is also a Cantor singleton.

4.1.7. Show that if $B$ is a Cantor singleton and $A \leq_{tt} B$, then $A$ is a Cantor singleton.

4.1.8. Show that if $A \oplus B$ is a Cantor singleton, then either $B$ is computable or $A$ is computable in $B$.

4.1.9. A set $A$ is said to be *autoreducible* if there is a computablee functional $F$ such that, for all $n$, $A(n) = F(n, A \setminus \{n\})$. Show that every ranked set $A$ is autoreducible (Owings). (Hint: note that, for any $n$, $A \setminus \{n\}$ and $A \cup \{n\}$ are both $\equiv_{tt} A$ and thus have rank $\alpha$.)

## 4.2 Basis results

In this section, we consider basis results for countable $\Pi^0_1$ classes. The key observation here is the following.

**Theorem 4.2.1.** *Any countable closed $P \subset \{0,1\}^{\mathbb{N}}$ has countable rank and has an isolated point.*

*Proof.* If $P$ has no isolated points, then $P$ is perfect and hence uncountable. For the rank, observe that for $\alpha < rk(P)$, $D^{\alpha}(P) \setminus D^{\alpha+1}(P)$ is nonempty. $\square$

Basis results for countable $\Pi^0_1$ classes can thus be obtained from the following.

**Theorem 4.2.2.** *(Kreisel [105]) Let $P$ be a $\Pi^0_1$ class.*

*(a) Any isolated member of $P$ is hyperarithmetic; if $P$ is finite, then every member of $P$ is hyperarithmetic.*

*(b) Suppose that $P$ is bounded. Then any isolated member of $P$ is computable in $0'$; if $P$ is finite, then every member of $P$ is computable in $0'$.*

*(c) Suppose $P$ is computably bounded. Then any isolated member of $P$ is computable; if $P$ is finite, then every member of $P$ is computable.*

*Proof.* Let $x$ be isolated in $P = [T]$ and take $n$ large enough so that $P \cap I(x\lceil n) = \{x\}$. Now define the $\Pi^0_1$ class $Q$ to be $P \cap I(x\lceil n)$. That is, $Q = [S]$, where $S$ is the computable tree consisting of all strings in $T$ which are compatible with $x\lceil n$. It follows that $Ext(S) = \{x\lceil n : n < \omega\}$, so that $x$ is computable in $Ext(S)$. Now consider the three cases.

(a) For an (unbounded) tree $S$, $Ext(S)$ is $\Sigma^1_1$ by Theorem 2.2.13 and since $[S]$ is a singleton, we have

$\sigma \in Ext(S) \iff (\forall \tau \in \omega^{|\sigma|}(\tau \neq \sigma \rightarrow \tau \notin Ext(S))$,

so that $Ext(T)$ is also $\Pi^1_1$. It follows that $Ext(T)$ and hence $x$ are hyperarithmetic.

(b) For a finitely branching tree $T$, $Ext(T)$ is $\Pi^0_2$ by Theorem 2.2.13. Now it follows from König's Lemma that for each $n$, there is some $k \geq n$ such that every sequence in $T$ of length $k$ is an extension of $x\lceil n$. Thus we have

$\sigma \in Ext(S) \iff (\exists k \geq |\sigma|)(\forall \tau \in \omega^k)(\tau \in S \rightarrow \sigma \prec \tau)$,

so that $Ext(S)$ is also $\Sigma^0_2$. It follows that $Ext(S)$ and hence $x$ are computable in $0'$.

(c) In this case, $Ext(S)$ is $\Pi^0_1$ by Theorem 2.2.13 and is also $\Sigma^0_1$ since

$\sigma \in Ext(S) \iff (\exists n \geq |\sigma|)(\forall \tau \in \{0,1,\ldots,f(n)\}^n)(\tau \in S \rightarrow \sigma \prec \tau)$,

where $f$ is a computable bounding function for $S$. Thus $Ext(S)$ and $x$ are both computable.

Suppose now that $P$ is finite. The conclusion in each case follows from the fact that every member of $P$ will be isolated. $\square$

Combining this with the previous result, we have

**Theorem 4.2.3.** *(Kreisel) Let $P$ be a countable $\Pi^0_1$ class.*

*(a) $P$ has a hyperarithmetic member.*

*(b) If $P$ is bounded, then $P$ has a member computable in $0'$.*

*(c) If $P$ is computably bounded, then $P$ has a computable member.*

As a corollary to the proof of Theorem 4.2.2, we also have the following.

**Theorem 4.2.4.** *Let $P = [T]$ be a $\Pi^0_1$ class.*

*(a) Suppose that $T$ is finite branching. Then any isolated member of $P$ is computable in $T'$; if $P$ is finite, then every member of $P$ is computable in $T'$. Thus if $P$ is countable, then $P$ has a member computable in $T'$.*

*(b) Suppose $T$ is computably bounded. Then any isolated member of $P$ is computable in $T$; if $P$ is finite, then every member of $P$ is computable in $T$. Thus if $P$ is countable, then $P$ has a member computable in $T$.*

## 4.3 Ranked Points and Rank-Faithful Classes

The following notions were introduced by G. Martin [132] and J. Owings [153]. A $\Pi^0_1$ class is said to be *rank-faithful* if $rk_P(x) = rk(x)$ for all $x \in P$. The $\Pi^0_1$ classes constructed in Theorems 2.6.8, 4.4.5 and 4.4.8 are clearly rank-faithful. A real $x$ is said to be a *Cantor Singleton* if it is the unique non-computable member of some $\Pi^0_1$ class. Theorem 2.ref2.2 implies that every non-computable $\Pi^0_1$ retraceable set is a Cantor singleton. G. Martin improved this result by showing that any noncomputable set which is the union of a computable set with a $\Pi^0_1$ retraceable set is a Cantor singleton. (See the exercises.)

Theorem 2.2.6.2 was improved to the following.

**Theorem 4.3.1.** *(G. Martin) For each computable ordinal $\alpha$ and every non-zero r.e. degree $\mathbf{a}$, there are c. e. sets $A$ and $B$ of degree $\mathbf{a}$ and rank $\alpha$ such that $A$ is a Cantor singleton and $B$ belongs to a rank-faithful $\Pi^0_1$ class.*

Recall from Theorem 2.8.2 that a minimal $\Pi^0_1$ class is a thin class of rank one such that all computable elements are isolated and the unique nonisolated element is noncomputable. It follows that any minimal $\Pi^0_1$ class is rank-faithful. This can be extended to arbitrary thin classes as follows.

**Theorem 4.3.2.** *Any thin $\Pi^0_1$ class $P$ is rank-faithful.*

*Proof.* Suppose that $A \in P$ and that $rk(A) = \alpha$. Let $D^\alpha(Q) = \{A\}$. Then $A \in P \cap Q$ and $P \cap Q = P \cap U$ for some clopen set $U$. It follows from Lemma 4.1.6 that $rk_P(A) = rk_{P\cap Q}(A) \leq rk_Q(A)$. Equality follows since $rk_Q(A)$ is minimal by assumption. □

Here are two interesting results from Owings [153].

**Theorem 4.3.3.** *(Owings) (a) If $A \oplus B$ is a Cantor singleton, then either $B$ is computable or $A$ is computable in $B$, so that the degree of $A \oplus B$ is either the degree of $A$ or the degree of $B$.*

*(b) If $rk(B) = rk(A \oplus B)$, then $A$ is computable in $B$.*

*Proof.* (a) Suppose that $A \oplus B$ is the unique noncomputable element of $P$ and that $B$ is noncomputable. Let $Q = P \cap (\{0,1\}^\omega \oplus \{B\}$ and observe that $B \leq_{tt} C$ for every $C \in Q$, so that any $C \in Q$ is noncomputable. Thus $A \oplus B$ is the unique element of $Q$ and is therefore computable in $B$ by the relativized version of Lemma 4.1.11.

(b) Let $\alpha = rk(A \oplus B)$ and let $P$ be a $\Pi^0_1$ class such that $D^\alpha(P) = \{A \oplus B\}$. As in (a), let $Q = P \cap (\{0,1\}^\omega \oplus \{B\})$. It follows from Lemma 4.3(b) that $rk(C) \geq \alpha$ for all $C \in Q$, so that in fact $Q = \{A \oplus B\}$. Then $A \oplus B$ is computable in $B$ as above. $\square$

Recall that a set $A$ is *autoreducible* if there is a computable functional $F$ such that, for all $n$, $A(n) = F(n, A \setminus \{n\})$.

**Theorem 4.3.4.** *(Owings) Every ranked set $A$ is autoreducible.*

*Proof.* Let $rk(A) = \alpha$ and let $P = [T]$ be a $\Pi^0_1$ class such that $D^\alpha(P) = \{A\}$. Note that for any $n$, $A \setminus \{n\}$ and $A \cup \{n\}$ are both $\equiv_{tt} A$ and thus also have rank $\alpha$. It follows that only one of them can belong to $P$. Thus, given $n$ and $A \setminus \{n\}$, we search for the least $k$ such that every $\sigma \in T$ of length $k$ consistent with $A \setminus \{n\}$, except possibly at $n$, has the same value $\sigma(n)$. Then $A(n) = \sigma(n)$. $\square$

**Exercises**

4.3.1. Prove G. Martin's result that any noncomputable set which is the union of a computable set with a $\Pi^0_1$ retraceable set is a Cantor singleton.

4.3.2. For any sets $A, B$ such that $B$ is a Cantor singleton and $A \leq_{tt} B$, show that $A$ is a Cantor singleton (Owings [153].) Hint: recall the proof of Lemma 4.4.3.

## 4.4 Rank and Complexity

In this section, we examine the connection between the rank and the hyperarithmetic complexity of a set.

Our first theorem will be a generalization of Theorem 4.2.2 (c) to arbitrary rank.

First we consider the complexity of the tree resulting from the iterated C-B derivative.

**Lemma 4.4.1.** *For any computable tree $T \subseteq \{0,1\}^*$, any computable limit ordinal $\lambda$ and any finite $n > 0$,* *(a)* $d^n(T) \leq_T \mathbf{0}^{(2n)}$ *and is* $\Sigma^0_{2n}$*;*

*(b)* $d^\lambda(T) \leq_T \mathbf{0}^{(\lambda+1)}$ *and is* $\Pi^0_{\lambda+1}$*;*

*(c)* $d^{\lambda+n}(T) \leq_T \mathbf{0}^{(\lambda+2n)}$ *and is* $\Sigma^0_{\lambda+2n+1}$*.*

*Proof.* Let $T$ be given and define the inductive operator $\Gamma$ on $\{0,1\}^*$ by

$$\sigma \in \Gamma(U) \iff \sigma \notin d(T) \ \vee \ \sigma \notin d(\{0,1\}^* - U).$$

Then $\Gamma$ is $\Pi^0_2$ by Definition 4.1.2 and it is easy to see that for all ordinals $\alpha$,

$$d^\alpha(T) = \{0,1\}^* - \Gamma^\alpha.$$

The result now follows immediately from Theorem 2.1.14.9. □

For decidable trees, there is a slight improvement.

**Lemma 4.4.2.** *For any decidable tree $T \subseteq \{0,1\}^*$, any computable limit ordinal $\lambda$ and any finite $n > 0$, $d^n(T) \leq_T \mathbf{0}^{(2n-1)}$ and is $\Sigma^0_{2n-1}$.*

*Proof.* Since $T$ is decidable, $Ext(T)$ is a computable set and it follows that $d(T)$ is a $\Pi^0_1$ set. The result now follows from Exercise 2.3. □

**Theorem 4.4.3.** *For any $A \in \{0,1\}^{\mathbb{N}}$, any $\Pi^0_1$ class $P$, any finite $n$ and any limit ordinal $\lambda$,*

*(a) If $rk_P(A) = n$, then $A \leq_T 0^{(2n)}$. Furthermore, if $rk_P(A) = n$ for some* **decidable** *$P$, then $A \leq_T 0^{(2n-1)}$.*

*(b) If $rk_P(A) = \lambda + n$, then $A \leq_T 0^{(\lambda+2n+1)}$.*

*Proof.* (a) Let $T$ be a computable binary tree such that $[d^n(T)] = \{A\}$. It follows from Lemma 4.4.1 that $d^n(T) \leq_T 0^{(2n)}$ and then by Theorem 4.2.4 that $A \leq_T 0^{(2n)}$. For a decidable class $P = [T]$, $d^n(T) \leq_T 0^{(2n-1)}$ by Lemma 4.4.2.

(b) Similarly $x \leq_T d^{\lambda+n}(T) \leq_T \mathbf{0}^{(\lambda+2n)}$. □

**Theorem 4.4.4.** *For any countable $\Pi^0_1$ class $P \subseteq \{0,1\}^{\mathbb{N}}$,*

*(a) $rk(P)$ is a computable ordinal;*

*(b) Every element of $P$ is hyperarithmetic.*

*Proof.* Since $P$ is countable, it follows that the perfect kernel of $P$ is empty. Thus the inductive definition $\Gamma$ given in the proof of Lemma 4.4.1 has closure $\{0,1\}^*$ which is a computable set. It follows from the Boundedness Theorem for inductive definability (Theorem II.1.13.12) that $|\Gamma| < \omega_1$ and therefore $rk(P) < \omega_1$. It now follows from Theorem 4.4.3 that every element of $P$ is hyperarithmetic. □

It follows from Theorem 4.4.3 that any set of rank one must be computable in $0''$. We now analyze the rank one sets further. Recall from Theorem 2.6.8 that every c. e. set is Turing equivalent to a hypersimple r.e. set of rank one. This result has the following improvement.

**Theorem 4.4.5.** *(Cenzer-Smith [42] For any noncomputable degree* $\mathbf{b} \leq 0'$*, there is a hyperimmune set* $B$ *with degree* $\mathbf{b}$ *of rank one; furthermore, there is a computable tree* $T$ *with no dead ends such that* $D([T]) = \{B\}$*.*

*Proof.* Let $A$ be a set of degree $\mathbf{b}$. By the limit lemma, there is a computable function $f$ such that, for all $e$,

$A(e) = \lim_n f(n, e)$.

Let $n(0)$ be the least $n > 0$ such that $f(n, 0) = A(0)$ and, for any $e$, let $n(e+1)$ be the least $n > n(e)$ such that, for all $i < e+2$, $f(n, i) = A(i)$. Then $n(0) < n(1) < \cdots$ is a *modulus* for the set $A$, so that $B = \{n(0), n(1), \cdots\}$ has degree $\mathbf{b}$.

We define a $\Pi^0_1$ class $P$ with $rk_P(B) = 1$.

The (possibly finite) set $C = \{m(0) < m(1) < \cdots\}$ is in $P$ if and only if $0 < m(0)$ and for all $e$, $i$, and $m$:

1. $(0 < m < m(0)\ \&\ C \neq 0) \rightarrow f(0, m) \neq f(0, m(0))$;
2. $e < i < card(C) \rightarrow f(m(i), e) = f(m(e), e)$;
3. $(e+1 < card(C)\ \&\ m(e) < m < m(e+1)) \rightarrow (\exists j < e+2)(f(m(e+1), j) \neq f(m, j))$.

It is clear that $P$ may be defined by a tree $T$ without dead ends and in fact closed under extension by 0. Also, $P$ contains all initial subsets of $B$ and in fact is closed under initial subsets, so that $rk_P(B) \geq 1$.

If $C = \{m(0) < m(1) < \cdots\}$ is any infinite set in $P$, then it follows from (ii) that $f(m(e), e) = f(n(e), e) = A(e)$ for all $e$. But it then follows from (i) or, by induction, from (iii) that $m(e) = n(e)$ for all $e$, so that $C = B$.

If $C = \{m(0) < m(1) < \cdots < m(k)\}$ is any finite set in $P$, then it follows from (i) and (iii) that there are at most two extensions $C \cup \{m\}$ of $C$ in $P$ (one with $f(m, k+1) = 0$ and one with $f(m, k+1) = 1$), so that $C$ is isolated in $P$.

Thus $B$ is the only non-isolated element of $P$ and therefore $rk_P(B) = 1$.

To see that $B$ is hyperimmune, suppose by way of contradiction that $h$ were a computable function with $h(e) > n(e)$ for all $e$. Then we could define a $\Pi^0_1$ subclass $Q$ of $P$ by adding the restriction

(iv) $(\forall e)[card(C \cap \{0, 1, \ldots, h(e) - 1\}) > e]$.

$\square$

We shall see later that not every $\Sigma^0_2$ degree contains a ranked set. Thus there is a more complicated result for degrees below $\mathbf{0}''$. The following theorem gives a partial inverse C-B derivative and shows that a large class of degrees contain ranked points.

**Theorem 4.4.6.** *[19] For any real* $B$ *and any tree* $S \subset \{0,1\}^*$ *such that* $S$ *is computable in* $B''$*, there is a tree* $T$ *computable in* $B$ *and a homeomorphism* $\Phi$ *from* $[S]$ *onto* $D([T])$ *such that for all* $x \in [S]$, $x \leq_T \Phi(x) \leq_T x \oplus B'$*.*

*Proof.* Let $F$ be a function, computable in $B$, such that, for all $\sigma \in \{0,1\}^*$,

$$\chi_S(\sigma) = \lim_{p\to\infty} \lim_{n\to\infty} F(p,n,\sigma).$$

For each $p$ and $\sigma$, let $F_p(\sigma) = \lim_{n\to\infty} F(p,n,\sigma)$.

Let $x \in \{0,1\}^{\mathbb{N}}$ be given and assume that $x \in [S]$. We define "outer modulus" values $p = p_k(x)$ such that $F_p(x\lceil k) = 1$ and "inner modulus" values $n = n_k(x)$ such that $F(p_k, n, x\lceil k) = 1$.

Let $p_1 = p_1(x)$ be the least $p > 0$ such that $F_p(x\lceil 1) = 1$ and let $n_1 = n_1(x)$ be the least $n > p_1$ such that, for all $m \geq n$, $F(p,m,x\lceil 1) = 1$. Now inductively define $p_{e+1}(x)$ to the least $p > n_e$ such that $F_p(x\lceil i) = 1$ for all $i \leq e+1$. Since $x \in [S]$, $p_{e+1}$ exists. Now let $n_{e+1}$ be the least $n > p_{e+1}$ such that

(1) for all $i \leq e+1$ and all $m \geq n$, $F(p_{e+1}, m, x\lceil i) = 1$, and

(2) for all $p$ with $n_e < p < p_{e+1}$, there is an $i \leq e+1$ such that, for all $m \geq n$, $F(p,m,x\lceil i) = 0$.

Note that our choice of $n_{e+1}$ not only verifies that $F_{p_{e+1}}(i) = 1$ for all $i \leq e+1$, it also verifies that $p_{e+1}$ is the least $p > n_e$ with this property.

Now for $x \in [S]$, let

**(3)** $H(x) = \{x(0)^\frown 1^\frown 0^{n_1\frown} 1^\frown 0^{p_1\frown} 1x(1)^\frown 1^\frown 0^{n_2\frown} 1^\frown 0^{p_2\frown} \ldots$

It is clear from the definition that $x$ is computable from $H(x)$ and that $H(x)$ is computable in $0' \oplus B'$.

The tree $T$ is to contain all initial segments of each image $H(x)$ together with certain strings which resemble those initial segments when only finitely many values of $F$ are considered. Every element of $[T] - H([S])$ will be isolated in $[T]$ and each point $H(x)$ will be the limit of a distinct sequence of points from $[T] \setminus H([S])$.

Elements of the computable tree $T$ with $D([T]) = \{B\}$ will be initial segments of strings $\sigma$ of the form

**(4)** $\sigma = r(0)10^{n_1}10^{p_1}1r(1)10^{n_2}10^{p_2}\ldots r(k-1)10^{n_k}10^{p_k}1.$

The string $\sigma$ as above is said to be *consistent* if it satisfies the following conditions:

**(5)** $0 < p_1 < n_1 < p_2 < n_2 < \cdots < p_k < n_k$.

**(6)** For all $i$, $j$, $k$ and $m$, if $i \leq j \leq k$ and $n_j \leq m \leq n_k$, then $F(p_j, m, r\lceil i) = 1$.

**(7)** For all $j$, $k$ and $p$, if $j \leq k$ and $n_{j-1} < p < p_j$, then there is an $i \leq j$ such that, for all $m$ with $n_i \leq m \leq n_k$, $F(p,m,i) \neq F(p,m,r\lceil i) = 0$.

If $\tau = \sigma^\frown r$ or $\tau = \sigma^\frown r^\frown 1^\frown 0^n$, then $\tau$ is consistent if $\sigma$ is consistent. The string $\rho = \sigma^\frown r^\frown 1^\frown 0^{n\frown} 1$ is consistent if $\sigma$ is consistent and if conditions (ii) and (iii) hold when $n_k$ is replaced by $n$ and the string $\rho^\frown 1^\frown 0^p$ is consistent if $\rho$ is consistent and $p < n$.

It is clear that for all $x \in [S]$, every initial segment of $H(x)$ is consistent. Now let $y \in \{0,1\}^{\mathbb{N}}$ be an extension of $\sigma$ as in (4) above and suppose that every initial segment of $y$ is consistent. There are two possibilities. Either $y$ has the form in (3) above or $y = \tau^\frown 0^\omega$, where $\tau = \sigma^\frown r(k+1)^\frown \ldots ^\frown r(\ell)^\frown 1$. In either case, the values of $n_j$ are unbounded in the initial segments of $y$. It now follows from (5) that:

**(8)** For all $i \leq j \leq k$, $F_{p_j}(r\lceil i) = 1$.

It also follows from (v) that:

**(9)** For all $j \leq k$ and all $p$ with $n_{j-1} < p < p_j$, there exists $i \leq j$ such that $F_p(r\lceil i) = 0$.

It is now clear that if $y$ has form (3), then for all $j$, $p_j$ is the least $p$ greater than $n_{j-1}$ (or $> 0$ for $j = 1$) such that $F_p(r\lceil i) = 1$ for all $i \leq j$. It follows that $x \in [S]$.

A consistent string $\tau$ with $\sigma \preceq \tau \preceq \sigma^\frown r^\frown 1^\frown 0^n_{k+1}{}^\frown 1^\frown 0^p_{k+1}$ is said to be *exact* if each $n_i$ is minimal, that is,

**(10)** For all $i, k, n$, if $i \leq k$ and $p_i < n < n_i$, then $r(0)10^{n_1}10^{p_1}1 \ldots 1r(i)10^n10^{p_i}1$ is not consistent.

The desired tree $T$ is defined to be the set of all exact strings. It is clear that for each $x \in [S]$, $H(x)$ is in $[T]$; furthermore, for each $k$, $x(0)^\frown 1^\frown 0^{n_1(x)\frown} 1 \ldots ^\frown 1^\frown x(k)^\frown 1^\frown 0^\omega$ belongs to $[T]$. Thus $H(x) \in D([T])$ as desired. It remains to show that all other elements of $[T]$ are isolated.

First suppose that $y \in [T]$ has infinitely many 1's. Then we saw above that $y$ has the form (3) where $x \in [S]$ with each $p_j$ minimal. It now follows from (10) that each $n_i$ is also minimal so that $y = H(x)$. Thus any element of $y$ of $[T] = H([S])$ must have the form

$$y = r(0)10^{n_1}10^{p_1}1r(1)10^{n_2}10^{p_2} \ldots r(k)10^\omega = \tau^\frown 0^\omega$$

Now suppose that $z \neq y$ and $z \in [T] \cap I(\tau)$. Then for some $n_k$ and $p_k$, $\tau^\frown 1^\frown 0^{n_k\frown} 1^\frown 0^{p_k\frown} 1 \prec z$. It follows from (8) and (9) that $p_k$ is uniquely determined and from (10) that $n_k$ is uniquely determined. Thus $y$ is the unique element of $[T]$ which extends $\tau^\frown 1^\frown 0^{n_k+1}$ and hence $y$ is isolated in $[T]$ as desired.

We now have that $H$ is an isomorphism of $[S]$ onto $[S]$ onto $D([T])$. Since $H$ is computable in $B$, it must be continuous and since $[S]$ and $D([T])$ are compact, it follows that $H^{-1}$ is also continuous. This completes the proof. $\square$

We note that this proof is uniform in $S$. That is, there is a primitive recursive function $\phi$ such that if $e$ is an index for $S$ as a $\Pi^0_3$-in-$B$ set, then $\phi(e)$ is an index for $T$ as a $\Pi^0_1$-in-$B$ set.

**Corollary 4.4.7.** *(a) For any finite $n$ and any tree $S \leq_T \mathbf{0}^{(2n)}$, there is a computable tree $T$ and a homeomorphism $H$ from $[S]$ onto $D^n([T])$ such that, for all $x \in [S]$, $x \leq_T H(x) \leq_T x \oplus \mathbf{0}^{(2n-1)}$.*

*(b) For any finite $n$ and any real $x$ such that $\mathbf{0}^{(2n-1)} \leq_T x \leq_T \mathbf{0}^{(2n)}$ or such that $\mathbf{0}^{(2n-2)} \leq_T x \leq_T \mathbf{0}^{(2n-1)}$, there is a computable tree $T$ and a real $y \equiv_T x$ with $|y|_T = |y| = n$.*

*Proof.* Part (a) follows easily from Theorem 4.4.6 by induction on $n$. Part (b) now follows from (a) by letting $[S] = \{x\}$. ☐

This result can be extended to higher levels of the hyperarithmetic hierarchy as follows. The following is Theorem 42 of [35].

**Theorem 4.4.8.** *For any computable ordinal $\alpha$ and any c. b. $\Pi^0_{2\alpha+1}$ class $Q$, there exists a $\Pi^0_1$ class $P$ of sets and a homeomorphism from $Q$ onto $D^\alpha(P)$ such that $x \leq_T H(x) \leq x \oplus 0^{2\alpha-1}$ for all $x \in Q$.*

*Proof.* The proof is by a uniform recursion up to a fixed computable ordinal $\kappa$ with a set of notations as given by Lemma 1.12.16. We may assume that $Q$ is actually a class of sets and build a $P$ which is computably bounded. In fact, we only need to assume that $Q = [S]$, where for all $\sigma \in S$ and all relevant notations $a$, $\sigma(i) \neq a$ for any $i$.

We will actually define computable functions $f$ and $\psi$ such that $\phi^{0^{o(a)-1}}_{\psi(e,a)}$ is a homeomorphism from $P_{e,2o(a)+1}$ onto $D^{o(a)}(P_{f(e,a)})$ and such that $x \leq \phi_{\psi(e,a)}(x)$ for each $x$.

The construction will be presented as a transfinite recursion on $o(a)$, but is actually obtained by the recursion theorem.

We need a series of lemmas. We shall write $A = \sqcup_n B_n$ if $A = \cup_n B_n$ and the elements of $\{B_n : n \in \omega\}$ are pairwise disjoint

**Lemma 4.4.9.** *There is a primitive recursive function $\rho$ such that, for each $\alpha = o(a)$, $P_{e,\alpha+3} = [T]$, with $T = \sqcup_n U_{\rho(e,a,n),\alpha+1}$.*

*Proof.* From the definition, $P_{e,\alpha+3} = [U]$, where $U = [U_{e,\alpha+3}]$ is uniformly $\Pi^0_{\alpha+3}$. It is then easy to define, as in Proposition 2.3.7 a $\Sigma^0_{\alpha+2}$ tree $S$ such that $[U] = [S]$. Thus there exists a $\Sigma^0_\alpha$ relation $R$ such that

$$\sigma \in S \iff (\exists n)(\forall m)R(m,n,e,\sigma)$$

As usual, we may assume that $\neg R(m,n,e,\sigma) \to \neg R(m,n+1,e,\sigma)$ and that $(\forall m)R(m,n,e,\sigma) \to (\forall m)R(m,n+1,e,\sigma)$.

Now define the desired trees by
$\sigma \in U_{\rho(e,a,\langle m,n\rangle),\alpha+1}$ if and only if

1. $(\forall m')(\forall i \leq |\sigma|)R(m', n, e, \sigma\lceil i)$

2. $(\forall n' < n)(\exists m' < m)(\exists i \leq |\sigma|)\neg R(m', n', e, \sigma\lceil i)$

3. $(\forall m' < m)(\exists n' < n)(\forall m'' < m')(\forall i \leq |\sigma|)R(m'', n', e, \sigma\lceil i)$.

□

We need a slightly modified notion from [19] of a *completely ranked* tree.

**Definition 4.4.10.** *A tree $T$ is completely ranked up to $\alpha$ if, for all $y \in [T]$ and all notations $a$ with $o(a) < \alpha$,*

*1. $y \in D^{o(a)}([T])$ if and only if $y(n) = a$ for infinitely many $n$.*

*2. $y \notin D^{o(a)}([T])$ implies that for some $n$, any extension $\tau \in T$ of $y\lceil n$ never has $\tau(i) = a$ for $i \geq n$.*

*$T$ is completely ranked if there exists a recursive ordinal $\alpha$ such that $T$ is completely ranked up to $\alpha$ and, for any $\tau \in T$, any notation $a$ with $o(a) \geq \alpha$, and any $i$, $\tau(i) \neq a$.*

Let $Q$ be a r. b. strong $\Pi^0_{2\alpha+1}$ class of sets. Note that elements of $Q$ are only allowed to contain notations for ordinals $> \alpha$. We will define a $\Pi^0_1$ class $P$, a recursively bounded, tree $T$, completely ranked up to $\alpha+1$, such that $P = [T]$ and a homeomorphism $\Phi$ from $Q$ onto $D^\alpha(P)$ such that $x \leq_T \Phi(x) \leq_T \oplus 0^{2\alpha-1}$ for all $x \in Q$. In fact, $x$ is always a subsequence of $\Phi(x)$. Furthermore, an element $y$ of $P$ will contain infinitely many notations for ordinals $> \alpha$ only if $y = \Phi(x)$ for some $x \in Q$.

For $\alpha = 0$, just let $Q = P$ and let $\Phi$ be the identity.

**Successor Case**: Suppose that $\beta = o(b) = o(a) + 1 = \alpha + 1$ and that $Q = P_{e,2\alpha+3} = [T]$. By Lemma 4.4.9, we have

$T = \sqcup_n U_{\rho(e,a,n),2\alpha+1} = \sqcup_n T_n$.

Now for $x \in Q$, let $n_i$ for each $i$ be the unique $n$ such that $x\lceil i+1 \in T_n$ and let

$$\Gamma(x) = (bx(0)0^{n_0+1}bx(1)0^{n_1+1}b\ldots).$$

Define the $\Pi^0_{2\alpha+1}$ tree $U$ to contain $\tau = (bx(0)0^{n_0+1}bx(1)0^{n_1+1}\ldots x(k)0^n)$ and its initial segments provided that $x\lceil i+1 \in T_{n_i}$ for all $i < k$.

It is easy to see that $G = [U]$ contains exactly $\{\Gamma(x) : x \in Q\}$ together with paths $y = (bx(0)0^{n_0+1}bx(1)0^{n_1+1}bx(k))^\frown 0^\omega$ for all $x \in Q$ and all $k$. Each of the latter will thus be isolated in $G$ and each of the former will have rank 1, so that $Q$ is homeomorphic to $D(G)$. Furthermore, for $x \in Q$, $x$ a subsequence of $\Gamma(x)$ and therefore is recursive in $\Gamma(x)$, since the values of $x$ are just those following immediately after $b$'s. $\Gamma(x)$ is recursive in $x \oplus \mathbf{0}^{2\alpha+1}$, since the computation of the sequence of witnesses $n_i$ may be performed with a $\Pi^0_{2\alpha+1}$ complete oracle.

In addition, for any $y \in G$, $y \in D(G)$ if and only if $y$ has infinitely many occurrences of $b$, and $y$ has no occurrences of any notation $\leq \alpha$. Let $f$ be a recursive function so that $\sigma(i) \leq f(i)$ for all $\sigma \in T$ and all $i$. Then for any $\tau \in U$ and any $i$, $\tau(i) \leq max\{b, f(i)\}$, so that $G$ is recursively bounded. In addition, if $y =$

$(bx(0)0^{n_0+1}bx(1)0^{n_1+1}bx(k))^\frown 0^\omega \notin D(G)$, then $(bx(0)0^{n_0+1}bx(1)0^{n_1+1}bx(k))^\frown 0^{n_k+1}$ may only be extended in $U$ by 0's.

Now by induction, there is a r. b. $\Pi^0_1$ class $P = [S]$ which is completely ranked up to $\beta$ and a homeomorphism $\Psi$ from $G$ onto $D^\alpha(P)$ such that $y \leq_T \Psi(y) \leq_T y \oplus \mathbf{0}^{2\alpha-1}$. We may assume that $y$ is always a subsequence of $\Psi(y)$ and that the values of $y$ follow immediately after the occurrences of $a$ in $\Psi(y)$.

It is immediate that $\Psi$ is also a homeomorphism from $D(G)$ onto $D^\beta(P)$. Now let $\Phi(x) = \Gamma(\Psi(x))$. Then clearly $\Phi$ is a homeomorphism from $Q$ onto $D^\beta(P)$.

It remains to show that $P$ is completely ranked up to $\beta + 1$. Suppose first that $z$ has infinitely many occurrences of $b$. Then, by the construction, $z = \Psi(y)$ for some $y \in G$ such that $y$ has infinitely many occurrences of $b$. Thus $y \in D(G)$, so that $z \in D^\beta(P)$. Next suppose that $z \in D^\beta(P)$. Then $z = \Psi(y)$ for some $y \in D(G)$. Now $y$ has infinitely many occurrences of $b$, so that $z$ must also have infinitely many occurrences of $b$.

Finally, suppose that $z \in P$ has only finitely many occurrences of $b$, so that $z \notin D^\beta(P)$. Observe that by the construction, the occurrences of $b$ may only follow immediately after occurrences of $a$. There are two cases.

First, suppose that $z \notin D^\alpha(P)$. Then $z$ has only finitely many occurrences of $a$ and by induction, there is some $n$ such that no extension $\tau \in S$ of $z\lceil n$ has any occurrences of $a$ past $\tau(n)$. It follows from the observation above that $b$ may not occur in $\tau$ past $\tau(n)$ either.

Next, suppose that $z \in D^\alpha(P)$. Then $z = \Psi(y)$ for some $y \in G$ such that $y$ has only finitely many occurrences of $b$. Thus by the construction, there is some $n$ so that $y\lceil n$ may only be extended by 0's in $U$. Now choose $m$ so that $z\lceil m$ includes the subsequence $y\lceil n$. It follows that no extension of $z\lceil m$ in $S$ may contain any further occurrences of $b$. This concludes the proof that $P$ is completely ranked up to $\beta + 1$.

We note that the construction of $P$ and of $\Psi$ are uniform.

**Limit Case**: Suppose that $\lambda = o(b)$ is a limit ordinal and that $Q = P_{e,\lambda+1} = [T]$. Let $a_0, a_1, \ldots$ enumerate the set of notations for ordinals less than $\lambda$ and let $\alpha_n = o(a_n)$. By definition, we have $Q = \cap_n Q_n = \cap_n [T_n]$, where $T_n = U_{\phi_e(a_n), o(a_n)+1}$.

By induction, we have constructed r.b. classes $P_n = [U_n]$, completely ranked up to $\lambda$ and homeomorphisms $\Phi_a$ from $Q_n$ onto $D^{\alpha_n+1}(P_n)$. For each $x \in Q$ and each $n$, let

$$\Phi_n(x) = (a_n x(0)\sigma_{n,0} a_n x(1)\sigma_{n,1} \ldots,$$

where each witness $\sigma_{n,i}$ contains no occurrence of $a_n$. Now let

$$\Phi(x) = (bx(0)\sigma_{0,0} bx(1)\sigma_{1,0} bx(2)\sigma_{1,0} \ldots.$$

Here the sequence of witnesses $\sigma_{i,n}$ is enumerated in order, first by the sum $i+n$ and then by the value of $n$.

It is immediate that $x \leq_T \Phi(x)$ and it follows from the uniformity of the construction that $\Phi(x) \leq_T x \oplus \mathbf{0}^\beta$.

Define the $\Pi^0_1$ tree $U$ to contain all initial segments of $\tau = (bx(0)\sigma_{0,0}bx(1)\sigma_{0,1}bx(2)\sigma_{1,0}\dots\sigma_{n,i}bx(k)\sigma)$ such that

1. For all $m, j$ with $m + j < n + i$ or with $m + j = n + i$ and $m \leq n$, $a_m x(0)\sigma_{m,0}a_m x(1)\sigma_{m,1}\dots a_m x(j)\sigma_{m,j} \in U_n$ and $\sigma_{m,j}$ contains no occurrence of $a_m$.

2. If $n \neq 0$, then $a_{n-1}x(0)\sigma_{n-1,0}a_{n-1}x(1)\sigma_{n-1,i}\dots a_{n-1}x(i+1)\sigma \in U_{n-1}$

3. If $n = 0$, then $a_{i+1}x(0)\sigma \in U_{i+1}$ and $\sigma$ contains no occurrence of $a_{i+1}$.

Then $P = [U]$ clearly contains $\Phi(x)$ for all $x \in Q$. It is easy to see that if $y \in P$ has infinitely many occurrences of $b$, then $y = \Phi(x)$ for some $x \in Q$. Now suppose that $y$ has only finitely many occurrences of $b$ and let

$$y = (bx(0)\sigma_{0,0}bx(1)\sigma_{0,1}bx(2)\sigma_{1,0}\dots\sigma_{n,i}bx(k))^\frown z,$$

where $z$ has no occurrences of $b$. Let

$$\rho = (bx(0)\sigma_{0,0}bx(1)\sigma_{0,1}bx(2)\sigma_{1,0}\dots\sigma_{n,i}bx(k)).$$

This given, we can define a string $\nu$ as follows. There are two cases.

**Case 1** If $n \neq 0$, then
$u = (a_{n-1}x(0)\sigma_{n-1,0}a_{n-1}x(1)\sigma_{n-1,i}\dots a_{n-1}x(i+1))^\frown z \in P_{n-1}$. Let

$$\nu = a_{n-1}x(0)\sigma_{n-1,0}a_{n-1}x(1)\sigma_{n-1,i}\dots a_{n-1}x(i+1)).$$

**Case 2** If $n = 0$, then $u = a_{i+1}x(0)^\frown z \in P_{i+1}$. Let

$$\nu = a_{i+1}x(0).$$

We will now establish several claims leading to the desired result that $P$ is completely ranked and that $\Phi$ is a homeomorphism of $Q$ onto $D^\beta(P)$. The proof depends on the definition of $\nu$. We will give the proofs for Case 1 and leave the simpler Case 2 to the reader.

We claim first that there is some initial segment $\tau$ of $y$ such that no extension of $\tau$ in $U$ has any further occurrences of $b$. There are two subcases here.

(Subcase a): If $z$ has an occurrence of $a_{n-1}$, then it follows from the definition of $U$ that no further occurrence of $b$ can occur after the first $a_n$ in $z$ has occurred in $y$. Let $\sigma$ be an initial segment of $z$ containing the first $a_n$.

(Subcase b): If $z$ has no occurrences of $a_{n-1}$, then since $U_{n-1}$ is completely ranked, there is some initial segment $\sigma$ of $z$ such that no extension of $\nu^\frown\sigma$ in $U_{n-1}$ may contain any further occurrences of $a_{n-1}$. It follows from the definition of $U$ that $b$ can not occur in $U$ past $\rho^\frown\sigma$.

Next, we claim that the rank $|y|_P$ of $y$ in $P$ equals the rank $|u|_{P_{n-1}}$ of $u$ in $P_{n-1}$. We first observe that for any $z'$, if $\nu^\frown z' \in U_{n-1}$, then $\rho^\frown z' \in U$. This implies that $|u|_{P_{n-1}} \leq |y|_P$. For the other inequality, observe that for any $z'$ extending $\sigma$, if $\rho^\frown z' \in U$, then $\nu^\frown z' \in U_{n-1}$.

It now follows that $\Phi(x)$ has rank at least $\lambda$ in $P$, since it is the limit of the sequence $\rho_n^\frown z_n$, where, for the appropriate value of $k$,

$\rho_n = (bx(0)\sigma_{0,0}bx(1)\sigma_{0,1}bx(2)\sigma_{1,0}\ldots\sigma_{n,i}bx(k))$,

$\nu_n = a_{n-1}x(0)\sigma_{n-1,0}a_{n-1}x(1)\sigma_{n-1,i}\ldots a_{n-1}x(i+1))$, and

$\Phi_n(x) = y = \nu_n^\frown z_n$.

Finally, we show that $P$ is completely ranked up to $\beta+1$. We have already established that $y$ has rank $\geq \beta$ if and only if $y$ has infinitely many occurrences of $b$. Now suppose that $|y|_P < \beta$ and let $\rho$, $\nu$ and $z$ be as above. Then for any $a$ with $o(a) \leq b$, $|y|_P \geq o(a)$ if and only if $\nu^\frown z$ has rank $\geq o(a)$ in $P_{n-1}$, which is if and only if $z$ has infinitely many occurrences of $a$, since $U_{n-1}$ is completely ranked, and this is if and only if $y$ has infinitely many occurrences of $a$. The previous discussion already established the other criterion for being completely ranked.

The uniformity of the proof shows that, using the Recursion theorem, we can actually compute indices for $P$ and for $\Phi$ from an index for $Q$. We omit the details. □

**Corollary 4.4.11.** *[35] For any computable ordinal $\lambda$ which is either 0 or a limit and any finite $n$:*

*(a) there is a $B \equiv_T 0^{(\lambda+2n)}$ with $rk(B) = \lambda + n$;*

*(b) for any degree $\mathbf{a}$ such that $\mathbf{0}^{(\lambda+2n+1)} \leq \mathbf{a} \leq \mathbf{0}^{(\lambda+2n+2)}$, there is a $B$ of degree $\mathbf{a}$ with $rk(B) = \lambda + n + 1$.*

*Proof.* □

Next we briefly consider sets which cannot be ranked.

**Theorem 4.4.12.** *(Cenzer-Smith [42]) For any hyperimmune set $A$, there is a $C \equiv_T A$ which is not ranked.*

*Proof.* Let $A = \{f(0) < f(1) < \cdots\}$ be hyperimmune. Let $[T_0], [T_1], \ldots$ enumerate the $\Pi^0_1$ classes as in Lemma 1.2. We first define $B \leq_T A$ so that $[T_i]$ is uncountable whenever $B \in [T_i]$, which implies that $B$ is not ranked. The characteristic function of $B$ is the limit of a sequence of strings $\sigma_n$ of length $f(n)$ which is computable in $A$. Let $\sigma_0 = 0$. Given $\sigma_n$, $\sigma_{n+1}$ is defined in two cases.

(Case 1) There is some $i \leq f(n)$ and some $\sigma$ of length $f(n)$ such that

1. $\sigma_n \in T_i$
2. $\sigma \notin T_i$
3. $\sigma\lceil n = \sigma_n\lceil n$
4. for any $j < i$, if $\sigma_n \notin T_i$, then $\sigma \notin T_j$.

Then we let $i$ be the least for which there is a $\sigma$ satisfying the conditions and we let $\sigma_{n+1}$ be the (lexicographically) least corresponding to that $i$.

(Case 2) If there is no such $i$, then $\sigma_{n+1} = \sigma_n^\frown 0^{f(n+1)-f(n)}$.

Let $B$ have characteristic function $\cup_n \sigma_n$. It is clear that $B \equiv_T A$. The proof that $B \in [T_i]$ implies $[T_i]$ uncountable is by induction. Suppose true for all $i < j$, suppose that $B \in [T_i]$, and choose $m$ large enough that $B \notin [T_i]$ implies $B\lceil m \notin T_i$ for all $i < j$. Then for any $n > f(m)$, any extension $\sigma \notin T_i$ of $B\lceil n$ of length $f(n)$ will satisfy the conditions of Case 1. Thus the shortest extension $\sigma$ of $B\lceil n$ not in $T_i$ must have length $> f(n)$. If $[T_i]$ were countable, then every $\sigma$ would have an extension not in $T_i$, so that we could define a function $h(n)$ to be the least $k$ such that any string $\sigma$ of length $n$ has an extension of length $k$ which is not in $T_i$. Then for $n > f(m)$, $h(n) > f(n)$, contradicting the assumption that $A$ is hyperimmune.

It follows that $B$ is unranked. Finally, let $C = A \oplus B$. Then $C \equiv_T A$ and $B \leq_{tt} C$, so that $C$ is unranked by Lemma 4.1.8. □

Of course this implies that every r.e. degree contains an unranked set. We say that $\mathbf{a}$ is *completely unranked* if every set $A$ of degree $\mathbf{a}$ is unranked. Jockusch and Shore construct in [88] a $\Sigma^0_2$ degree which is completely unranked. Downey observed that since sets with the same truth-table degree have the same rank and since all sets in a hyperimmune-free degree have the same truth-table degree (see [151], p. 589), the construction of an unranked set of hyper-immune free degree will provide a completely unranked degree. This led to the following improvement in of the Jockusch-Shore result.

**Theorem 4.4.13.** *(Downey [55]) There is a hyperimmune-free degree which is completely unranked.*

On the other hand, Downey also showed the following, again using a hyperimmune free degree $\mathbf{a}$, so that every $A$ of degree $\leq \mathbf{a}$ is in fact $\leq_{tt} \mathbf{a}$.

**Theorem 4.4.14.** *(Downey [55]) There exists a degree $\mathbf{a} \leq \mathbf{0}''$ such that every set $A$ of degree $\leq \mathbf{a}$ is ranked.*

Cenzer and Smith consider in [42] the problem of sets below $\mathbf{0}'$ but with high rank. They showed that for every computable ordinal $\alpha$, there is a $\Delta^0_2$ set $A$ of rank $\alpha$. This was improved by Cholak and Downey in [46].

**Theorem 4.4.15.** *(Cholak-Downey [46]) For each computable ordinal $\alpha$, there is an r.e. set of rank $\alpha$.*

## 4.5 Computable Trees with One or No Infinite Branches

Recall the notions of the height $ht(T)$ of a well-founded tree $T \subseteq \mathbb{N}^*$ and the height $ht_T(\sigma)$ of the nodes of $T$ introduced in Section **??**.1.14. If $T$ has a unique infinite branch, then following Clote [48], we let $\gamma_T = sup\{ht_T(\sigma) + 1 : T[s] \text{ is well-founded}\}$. Theorem **??**.1.14.19 of Clote showed that the hyperarithmetic complexity of the unique infinite branch is bounded in some sense by the height of $T$. In this section we consider the reverse result that every hyperarithmetic set is reducible to the unique infinite branch of some computable tree.

We also look at the complexity of the perfect kernel $K(Q)$ of a $\Pi^0_1$ class in $\{0,1\}^{\mathbb{N}}$.

In section 2.1.9, we showed that any $\Pi^0_1$ class $P \subseteq \mathbb{N}^{\mathbb{N}}$ can be reduced to a $\Pi^0_1$ class $Q \subseteq \{0,1\}^{\mathbb{N}}$. In this section, we consider the connection between the height of a computable tree $T$ such that $P = [T]$ and the rank of the tree $S$ such that $Q = [S]$.

The following theorem is a variant of the result of Clote [48].

**Theorem 4.5.1.** *For any computable ordinal $\alpha$ and any hyperarithmetic index $a \in H^{\alpha\cdot 2}$, there is a computable tree $T$ with unique infinite branch $x$ such that $\gamma_T \leq \omega\alpha$ and $H_a \leq_T x$.*

*Proof.* In fact, we will use the recursion theorem to obtain a function $\psi$ such that for all hyperarithmetic indices $a \in H^{\alpha\cdot 2}$, $T_{\psi(a}$ is a computable tree with unique infinite branch $x_a$ such that $\gamma_{T_{\psi(a)}} \leq \omega\alpha$ and $H_a \leq x_a$. Recall that $H_a = \cup_n \mathbb{N} - H_{\phi_a(n)}$, so that for $a \in H^{\alpha\cdot 2}$, we have

$$i \in H_a \iff (\exists n)(\forall k) i \in H_{\phi_{\phi_a(n)}(k)}(i).$$

Using Lemmas 1.14.6, 1.14.7 and 1.14.8, we can find a function $f$ such that

$$i \notin H_a \iff (\exists \infty n) i \in H_{f(a,n)},$$

where $f(a,n) \in H^{\beta\cdot 2}$ for some $\beta$ with $\beta + 1 \leq \alpha$.

For the base case of $\alpha = 0$, we may assume that $H_a$ is uniformly primitive recursive. Let $p_i$ denote the $i$th prime number. Then for each $i$, we $x_a(p_i) = \langle \chi_{H_a}(i) \rangle$ and we let $x_a(j) = 0$ for all non-primes $j$. Let $T_{\psi(a)} = \{x_a \lceil n : n \in \mathbb{N}\}$. Certainly $\gamma(T_{\psi(a)}) = 0$ for all such $a$.

Now we may assume that for each $a$ and $n$, there is a tree $T_{a,n} = T_{\psi(f(a,n))}$ with unique infinite branch $x_{a,n}$ and $\gamma_{T_{a,n}} \leq \omega \cdot \beta$ for some $\beta$ with $\beta + 1 \leq \alpha$. Furthermore, for every $i$ and $n$, $x_{a,n}(p_i) = \langle \tau \rangle$ for some sequence $\tau$ such that $\tau(0) = \chi_{H_{f(a,n)}}(i)$. Now define $y_a$ and $x_a$ as follows. For all $i$ and $n$,

$$y_a(2^{i+1} \cdot 3^{n+1}) = x_{a,n}(i).$$

For all $i$, there are two cases in the definition of $y_a(p_i)$.

**Case I:** $y_a(p_i) = 0$. Then for all $k$,

$$y_a(p_i^{k+1}) = (\text{least } n > y_a(p_i^k) i \in H_{f(a,n)}.$$

Here the sequence of values $y_a(p_i^{k+1})$ provides an infinite set of witnesses that $i \notin H_a$.

**Case II:** $y_a(p_i) = 1$. Then

$$y_a(p_i^2) = (\text{least } n)(\forall m \geq n) i \notin H_{f(a,m)}$$

and $y_a(p_i^k) = 0$ for all $k > 2$. Here $y_a(p_i^2)$ provides a witness that $i \in H_a$. Finally

$$x_a(j) = \langle y_a(j), y_a(j-1), \ldots, y_a(0)\rangle.$$

Observe that $H_a$ may be computed from $x_a$ by

$$i \in H_a \iff (x_a(i))_0 = 1 \iff \neg(x_a(i))_0 = 0.$$

The computable tree $T = T_{\psi(a)}$ with unique infinite branch $x_a$ is defined as follows. Given a sequence $\sigma = (\sigma(0), \ldots, \sigma(m-1))$ which is a candidate for membership in $T$, we first require that there exists a sequence $e_0, e_1, \ldots, e_{m-1}$ such that for each $j < m$, $\sigma(j)$ has the form $\langle e_j, e_{j-1}, \ldots, e_0\rangle$. Then for $j = 2^{i+1} \cdot 3^{n+1} < m$, we require that

$$(e_{2^{i+1}.3}, e_{2^{i+1}.9}, \ldots, e_{2^{i+1}.3^{n+1}}) \in T_{a,n}.$$

These conditions mean that $\sigma$ provides an answer to whether $i \in H_{f(a,n)}$ for a certain set of $(i, m, n)$. For $j = p_i$, $e_j = 0$ indicates the guess that $i \notin H_a$ and then $\{e_{p_i^{k+1}}\}_k$ is intended to enumerate the infinite set $\{n : i \in H_{f(a,n)}\}$. Similarly $e_j = 1$ indicates the guess that $i \in H_a$. Then for $j = p_i^2$, $e_j = n$ is intended to be the least $n$ such that $i \notin H_{f(a,m)}$ for all $m \geq n$. The string $\sigma$ is in $T$ if and only if the "witnesses" provided by $\sigma$ as to whether $i \in H_a$ are confirmed by the coded subsequences from the trees $T_{a,n}$.

It follows that $x_a$ is the unique infinite branch of $T$. That is, suppose that $x \in [T]$ and has the proper form. Then each of the coded subsequences must be in $T_{a,n}$ so that each infinite coded subsequence must be $x_{a,n}$. Now the coded values of $y_a(p_i^k)$ must be correct since the witnesses from $T_{a,n}$ are all correct.

It remains to check that $\gamma(T) \leq \omega \cdot \omega \cdot \alpha$. Let $\sigma \in T$ be a dead end of length $m$ in the proper form and let $e_0, e_1, \ldots, e_{m-1}$ be given as above. There are two cases.

**Case I:** For some $n$, the coded subsequence $\tau = (e_{2^{i+1}.3}, e_{2^{i+1}.9}, \ldots, e_{2^{i+1}.3^{n+1}})$ is a dead end of $T_{a,n}$. Then it can be shown by induction that $ht_T(\sigma) \leq ht_{T_{a,n}}(\tau) \leq \omega \cdot \beta < \omega \cdot \alpha$ for some $\beta < \alpha$.

**Case II:** All coded subsequences from any $T_{a,n}$ are initial segments of $x_{a,n}$.

**Case IIa:** $e_j = y_a(j)$ for all primes $j$ but some witness $e_j$ where $j = p_i^k$ is different from $y_a(p_i^k)$. Suppose first that $y_a(i) = 0$ is correct and the witnesses should be $n_2 = y_a(p_i^2), n_3 = y_a(p_i^3), \ldots$ but for $j = p_i^k$, $e_j \neq n_j$. Now let $\sigma' \in T$ be any extension of $\sigma$ long enough to predict the value of $x_{a,n}(n_j)$. Then $\tau$ must be incorrect as in Case I and thus $ht_T(\tau) \leq \omega\beta$. Thus $ht_T(\sigma) \leq \omega \times \beta + n < \omega \cdot \alpha$. A similar argument apples when $y_a(i) = 1$ is correct but the witness $e_j$ where $j = p_i^2$ is incorrect.

**Case IIb:** For some $i$ and for $j = p_i$, $e_j \neq y_a(j)$. Suppose first that $e_j = 0$ but $y_a(j) = 1$. Then there can be only a finite number $K$ of witnesses in the sequence $y_a(p_i^k)$ and any extension of $\sigma$ long enough to predict $y_a(p_i^K + 1)$ must code a dead end of $T_{a,K+1}$ as in Case IIa. A similar argument applies when $e_j = 1$ but $y_a(j) = 0$. □

**Corollary 4.5.2.** *For any computable ordinal $\alpha$, hyperarithmetic index $a \in H^{\alpha\cdot 2}$, there is a computable tree $T$ with unique infinite branch $x$ such that $\gamma_T \leq \omega\alpha$ and $x$ is $\Sigma^0_{\alpha\cdot 2}$ complete.*

The next result explores further the relation between trees $T$ in $\mathbb{N}^*$ with a unique infinite branch and the natural images $\Phi(T) \in \{0,1\}^*$, relating the height of $T$ with the rank of $\Phi(T)$. Recall from the proof of Theorem 2.2.7.3 the definition of $\Phi(x) = 0^{x(0)}10^{x(1)}\dots$ mapping $\mathbb{N}^{\mathbb{N}}$ into $\{0,1\}^{\mathbb{N}}$ and the corresponding mapping of trees so that

$$0^{\sigma(0)}10^{x(1)}1\dots 0^{x(k-1)}10^{x(k)} \in S = \Phi(T) \iff (\sigma(0),\dots,\sigma(k-1)) \in T.$$

Then a dead end $\sigma \in T$ of length $k$ corresponds to an isolated point

$$y_\sigma = 0^{\sigma(0)}10^{x(1)}1\dots 0^{x(k-1)}10^\infty \in [S].$$

**Theorem 4.5.3.** *Let $T \subset \mathbb{N}^*$ be a computable tree with no infinite paths and let $S = \Phi[T]$ be the image of $T$ in $\{0,1\}^*$ and $Q = [S]$. Then for any $\sigma \in T$, $rk_Q(y_\sigma) \leq ht_T(\sigma)$.*

*Proof.* It is easy to see that $Q = \{y_\sigma : \sigma \in T$. The inequality is proved by induction on $\alpha = ht_T(\sigma)$. If $ht_T(\sigma) = 0$, then $\sigma$ has no immediate extensions in $T$ and hence $y_\sigma$ is isolated in $Q$ so that $rk_Q(y_\sigma) = 0$. Now let $\sigma = (\sigma(0),\dots,\sigma(k-1))$ and suppose that $ht_T(\sigma) = \alpha$, so that for any proper extension $\tau \in T$ of $\sigma$, $ht(T) < \alpha$. Let $y \in Q$ be any extension of $\sigma^* = 0^{\sigma(0)}10^{x(1)}1\dots 0^{x(k-1)}1$ different from $y_\sigma$. Then $y = y_\tau$ for some extension $\tau \in T$ of $\sigma$, so that $ht_T(\tau) < \alpha$ and hence by induction $rk(y) < \alpha$. It follows that $rk_Q(y_\sigma) \leq \alpha$. □

**Exercises**

4.5.1. Show that the reverse inequality in Theorem 4.5.3 does not hold. In fact, for any computable ordinal $\alpha$, there is a computable tree $T$ with no infinite path and $\sigma \in T$ such that the rank of $y_\sigma$ in $\Phi[T]$ is 0 but $ht_T(\sigma) = \alpha + 1$.

4.5.2. Combine Theorem 4.5.3 and Corollary 4.5.2 to show that for any computable ordinal $\alpha$, there is a real $x \in \{0,1\}^{\mathbb{N}}$ of degree $\mathbf{0}^{\alpha\cdot 2}$ and a $\Pi^0_1$ class $Q$ such that $rk_Q(x) \leq \omega\cdot\alpha$.

## 4.6 Logical Theories revisited

In this section, we apply the basis results for countable $\Pi^0_1$ classes to axiomatizable theories.

**Theorem 4.6.1.** *Let $\Gamma$ be an axiomatizable first order theory with only countably many complete consistent extensions. Then*

*(a)* *$\Gamma$ has a decidable complete consistent extension.*

(b) *If $\Gamma$ has only finitely many complete consistent extensions, then every complete consistent extension is decidable.*

(c) *Every complete consistent extension of $\Gamma$ is hyperarithmetic.*

*Proof.* Let $\Gamma$ be an axiomatizable theory with countably many complete consistent extensions. By Theorem 2.2.9.1, the set of complete consistent extensions of $\Gamma$ may be represented as a $\Pi^0_1$ class $P$. It follows from Theorem 4.2.3 that $P$ has a computable member, which represents a decidable complete consistent extension of $\Gamma$. Similarly, it follows from Theorem 4.4.4 that every complete consistent extension of $\Gamma$ is hyperarithmetic. If $\Gamma$ has only finitely many complete consistent extensions, then $P$ has only finitely many elements and therefore all of the elements of $P$ are computable by Theorem 4.2.2 and thus all of the complete consistent extensions of $\Gamma$ are decidable. □

Given an axiomatizable theory $\Gamma$, let us say that an extension $\Delta$ of $\Gamma$ is a *finite extension* of $\Gamma$ if there is a finite set $F$ of sentences such that $\Delta$ is logically equivalent to $\Gamma \cup F$. Then it is easy to see that a complete consistent extension $\Delta$ of $\Gamma$ is a finite extension if and only if $\Delta$ is isolated in the $\Pi^0_1$ class of complete consistent extensions of $\Gamma$. In particular, any complete consistent finite extension of $\Gamma$ must be decidable. Thus if $\Delta$ is an undecidable complete consistent extension of $\Gamma$, then $\Delta$ is not a finite extension. We now want to focus on complete consistent extensions of rank one.

**Theorem 4.6.2.** *Let $\Gamma$ be an axiomatizable first-order theory which has a unique complete consistent, non-finite extension $\Delta$. Then $\Delta \leq_T \mathbf{0}''$ and if $\Gamma$ is decidable, then $\Delta \leq_T \mathbf{0}'$.*

*Proof.* This follows from Theorem 4.4.3. □

Here is an existence result.

**Theorem 4.6.3.** (a) *For any degree $\mathbf{b} \leq 0'$, there is a decidable theory $\Gamma$ with unique complete consistent, non-finite extension $\Delta$ and $\Delta$ has degree $\mathbf{b}$.*

(b) *For any degree $\mathbf{b}$ such that $\mathbf{0}' \leq \mathbf{b} \leq \mathbf{0}'$, there is an axiomatizable theory $\Gamma$ with unique complete consistent, non-finite extension $\Delta$ of degree $\mathbf{b}$.*

*Proof.* (a) Let $\mathbf{b} \leq 0'$. Then by Theorem 4.4.5, there is a decidable $\Pi^0_1$ class $P$ with unique nonisolated element $B$ of degree $\mathbf{b}$. By Theorem 2.2.9.3, there is a decidable theory $\Gamma$ such that $P$ represents the set of complete consistent extensions of $\Gamma$ and thus $B$ represents the unique non-finite complete consistent extension $\Delta$ of $\Gamma$.

(b) This follows from Corollary 4.5.2 and Theorem 2.2.9.3 as in (a). □

# Chapter 5

# Index Sets

The notions of *enumeration* and of an *index set* are fundamental in the study of the computable functions and computably enumerable sets. The complexity (in the arithemetic hierarchy) of many properties can be measured using index sets. For example, the index set $Inf = \{a : W_a \text{ is infinite}\}$ is $\Pi^0_2$ complete, so that from this point of view, the property of being infinite is $\Pi^0_2$. The chapter begins with a brief list of such results for c.e. index sets, together with their complexity.

We present an enumeration of the $\Pi^0_1$ classes and then classify several index sets for $\Pi^0_1$ classes. In particular, we study index sets for properties related to cardinality, computable cardinality, measure and category. We then indicate how these index sets will play an important role in the application of $\Pi^0_1$ classes to various mathematical problems in Part 2.

A set $A$ is said to be an *index set* (for c. e. sets) if for any $a, b$, $a \in A$ and $\phi_a = \phi_b$ imply that $b \in A$. We can also define a *co-index set* to be a set $A$ such that for any $a, b$, $(a \in A \;\&\; b \in A \;\&\; \phi_a = \phi_b)$ implies that $a = b$. Thus in particular, $\emptyset$ and $\omega$ are index sets. Rice's Theorem ([181], p. 21) states that these are the only two computable index sets. We have defined the index sets $K$ and $K_0$ in Section 1.7. In fact, it is the case that if $A$ is an index set other than $\emptyset, \omega$, then $K \leq_1 A$ or $K \leq_1 \bar{A}$ where $K = \{a : a \in W_a\}$. Here are some other examples of index sets which we will employ:

- $K_1 = \{a : W_a \neq \emptyset\}$;
- $Fin = \{a : W_a \text{ is finite}\}$;
- $Inf = \{a : W_a \text{ is infinite}\}$;
- $Cof = \{a : \omega \setminus W_a \text{ is finite}\}$;
- $Coinf = \{a : \omega \setminus W_a \text{ is infinite}\}$;
- $Rec = \{a : W_a \text{ is a computable set}\}$;
- $Tot = \{a : \phi_a \text{ is total}\}$;

- $Ext = \{a : \phi_a$ is extendible to a total computable function$\}$;
- $Ext_2 = \{a : \phi_a$ is extendible to a total $\{0,1\}$-valued computable function$\}$;
- $Comp = \{e : W_e \equiv_T K\}$;
- $U_1^1 = \{a : (\exists x)(\forall n) < x\lceil n >\notin W_a\}$.

Following Soare [181], p. 66, we define $(\Sigma_n^m, \Pi_n^m) \leq_m (B, C)$ for a disjoint pair of sets $B$ and $C$ if for some $\Sigma_n^m$ complete set $A$, there is a computable function $f$ such that, for any $a$, $a \in A \iff f(a) \in B$ and $a \notin A \iff f(a) \in C$. If $B$ is $\Sigma_n^m$, $C$ is $\Pi_n^m$ and $(\Sigma_n^m, \Pi_n^m) \leq_m (B, C)$, then we will say that the pair $(B, C)$ is $(\Sigma_n^m, \Pi_n^m)$ *complete*.

The index sets described above all turn out to be complete for some level of the arithmetical hierarchy. Here is a brief list of such complexity results, most taken from Soare [181], where the reader can find a further discussion of index sets. We give some details of the proofs as preparation for the work on index sets for $\Pi_1^0$ classes.

**Theorem 5.0.4.** *(i) $K$, $K_0$ and $K_1$ are $\Sigma_1^0$ complete sets;*

*(ii) $Tot$ is a $\Pi_2^0$ complete set;*

*(iii) $(Fin, Inf)$ is $(\Sigma_2^0, \Pi_2^0)$ complete.*

*(iv) $(Cof, Coinf)$ is $(\Sigma_3^0, \Pi_3^0)$ complete;*

*(v) $Ext$, $Ext_2$, and $Rec$ are $\Sigma_3^0$ complete sets;*

*(vi) $Comp$ is $\Sigma_4^0$ complete;*

*(vii) $U_1^1$ is a $\Sigma_1^1$ complete set.*

*Sketch.* (i) It is clear that these are all $\Sigma_1^0$ sets and that $K_0$ is complete. Let $W = Dom(\phi)$ be any c. e. set and define the partial recursive function $\phi_e$ so that $\phi_e(m, i) = \phi(m)$. Let $f(m) = S_1^1(e, m)$ so that $\phi_{f(m)}(i) = \phi_e(m, i) = \phi(m)$. Then

$$m \in W \iff \phi(m)\downarrow \iff W_{f(m)} = \omega$$

and

$$m \notin W \iff \phi(m)\downarrow \iff W_{f(m)} = \emptyset.$$

Thus $m \in W \iff f(m) \in W_{f(m)} \iff W_{f(m)} \neq \emptyset$.

(ii,iii) $a \in Tot \iff (\forall m)(\exists s) m \in W_{a,s}$ and $a \in Fin \iff (\exists m)(\forall n > m)(\forall s) n \notin W_{a,s}$.

For the completeness, let $B$ be a $\Pi_2^0$ set and let $R$ be a computable relatio such that

$$i \in B \iff (\forall m)(\exists n) R(i, m, n).$$

It is an important observation that we may assume that $R(i, m, n) \to R(i, m, n+1)$ and that $R(i, m+1, n) \to R(i, m, n)$. That is, let $R'(i, m, n) \iff (\forall m' \leq m)(\exists n' \leq n) R(i, m', n')$. Then $R'$ has the desired properties.

Now let $\phi_a(i,m) = (\mu n)R(i,m,n)$ and let $g(i) = S^1_1(a,i)$, so that $\phi_{g(i)}(m) = (\mu n)R(i,m,n)$. If $i \in B$, then $W_{g(i)} = \omega$, so that $g(i) \in Tot$ and also $g(i) \in Inf$. If $i \notin B$, then $W_{g(i)}$ is finite, so that $g(i) \in Fin$ and $g(i) \notin Tot$.

(iv,v) Let $A$ be a $\Sigma^0_3$ set and let $R$ be a computable relation so that, for all $a$,

$$a \in A \iff (\exists m)(\forall n)(\exists k)R(a,m,n,k)$$

We will define a primitive recursive function $f$ such that $a \in A$ if and only if $W_{f(a)}$ is cofinite, which will be if and only if $W_{f(a)}$ is computable. Let the standard noncomputable c. e. set $K$ have a computable enumeration $K = \cup_s K_s$. The c. e. set $W_{f(a)}$ in stages $W_{f(a),s}$ so that $\omega \setminus W_{f(a),s} = \{\{b^s_{a,0} < b^s_{a,1} < \dots\}$.

**Stage 0**: $W_{f(a),s} = \emptyset$.

**Stage** $s+1$: For each $m \le s$ such that either $m \in K_{s+1} \setminus K_s$ or such that there is some $n \le s$ such that $(\forall n' \le n)(\exists k \le s)R(a,m,n',k)$ but $\neg(\forall n' \le n)(\exists k < s)R(a,m,n',k)$, enumerate $b^s_{a,m}$ into $W_{f(a),s+1}$.

If $a \in A$, then for some $m$, $b^s_{a,m}$ is put into $W_{f(a),s+1}$ infinitely often, so that $\lim_s b^s_{a,m} = \infty$ and hence $W_{f(a)}$ is cofinite. If $a \notin A$, then for every $m$, $\lim_s b^s_{a,m} = b_{a,m} < \infty$. Thus $W_{f(a)}$ is coinfinite. Furthermore, $K \le W_{f(a)}$ (so that $W_{f(a)}$ is not computable), since $m \in K \iff m \in K_{b_{a,m}}$ and $b_{a,m}$ can be uniformly computed from $W_{f(a)}$.

This argument can be modified to show that $Ext$ and $Ext_2$ are both $\Sigma^0_3$ complete, as follows. Here we want to say that $a \in A$ if and only if $\phi_a$ is extendible. As before, put $m \in W_{f(a)}$ at stage $s+1$ (by defining $\phi_{f(a)}(m) = 0$) if the list of $n$ such that $(\exists k)R(a,m,n,k)$ becomes longer at stage $s$. Also, replace the action associated with the set $K$ with the following. If $\phi_{m,s}(b^s_{a,m}) = j$ and

$$(\forall i < b^s_{a,m})\phi_{f(a),s}(i) = \phi_{m,s}(i),$$

then define $\phi_{f(a)}(b^s_{a,m}) = 1 - j$, thus putting $b^s_{a,m} \in W_{f(a)}$. (By the usual convention, $a - b = 0$ if $a < b$.) If $a \in A$, then $W_{f(a)}$ is cofinite as before, so that $\phi_{f(a)}$ is extendible. If $a \notin A$, then for each $m$, $\phi_{f(a)}(b_{a,m})$ is either undefined or not equal to $\phi_m(b_{a,m})$, so that $\phi_{f(a)}$ is not extendible.

(vi) A proof is given in Soare [181, Ch. XII].

(vii) A proof can be found in Hinman [80, p. 84]. □

Each of the results above can be relativized. That is, let $W^x_e = \{n : \Phi^x_e(n)\downarrow\}$. Then for example, $Fin^x = \{a : W^x_a \text{ is finite}\}$ is $\Sigma^{0,x}_2$ complete. In particular, if we let $x = \emptyset^{(n)}$ denote the n-th jump of the emptyset, then Post's theorem (Theorem 1.10.7) implies that a set is $\Sigma^{0,x}_k$ if and only if it is $\Sigma^0_{n+k}$, see Soare [181]. It follows that, for example, $Fin^K$ is $\Sigma^0_3$ complete.

## 5.1 Index sets for $\Pi^0_1$ classes

There are several different ways to define index sets to $\Pi^0_1$ classes. We use here an approach from [36] based on primitive recursive trees.

Let $\sigma_n$ denote the string $\sigma \in \mathbb{N}^*$ such that $\langle\sigma\rangle = n$. Then $\sigma_0, \sigma_1, \dots$ enumerate $\mathbb{N}^*$, and furthermore, whenever $\sigma_i \prec \sigma_j$, it must be the case that $i < j$.

Then a tree $T$ is primitive recursive, computable, etc. if the corresponding set $\{i : \sigma_i \in T\}$ is itself primitive recursive, computable, etc..

**Definition 5.1.1.** *Let $\pi_e$ denote the eth primitive recursive function and let $\sigma \in T_e \iff (\forall \tau \preceq \sigma)\pi_e(\langle\tau\rangle) = 1$; let $P_e = [T_e]$.*

**Lemma 5.1.2.** *(a) For each $e$, $\Pi_e$ is a $\Pi^0_1$ class;*

*(b) For each $\Pi^0_1$ class $P$, there are infinitely many $e$ such that $P = P_e$.*

*Proof.* Part (a) is clear. Part (b) follows from Proposition 2.3.1 and the observation that every primitive recursive function has infinitely many indices. □

There are several other approaches to defining and enumerating the $\Pi^0_1$ classes. Some of these were studied in [18].

For example, one can first define the $\Sigma^0_1$ classes (or c. e. open sets, and then obtain the $\Pi^0_1$ classes as compements.

For any set $W \subseteq \omega^*$, we define the open set generated by $W$ to be

$$\mathcal{O}(W) = \bigcup\{I(\sigma) : \langle\sigma\rangle \in W\}.$$

We say that a subset $U$ of $\mathbb{N}^{\mathbb{N}}$ is a $\Sigma^0_1$ class, or c. e. open set, if $U = \mathcal{O}(W)$ for some c. e. set $W$. Thus the $\Sigma^0_1$ classes may be enumerated in the form $U_e = \mathcal{O}(W_e)$ and the $\Pi^0_1$ classes in the form $\Psi(e) = \mathbb{N}^{\mathbb{N}} - U_e$.

**Proposition 5.1.3.** *There are primitive recursive functions $\phi$ and $\psi$ such that $P_e = \Psi(\phi(e))$ and $\Psi(e) = P_{\psi(e)}$ for all $e$.*

*Proof.* For each $e$, recall that $W_{e,s}$ is the set of elements enumerated into the $e$th c.e. set $W_e$ by stage $s$ and define $\psi$ so that

$$\sigma \in T_{\psi(e)} \iff (\forall \tau \sqsubseteq \sigma)\ \langle\tau\rangle \notin W_{e,|\sigma|}.$$

It is easy to seee that $P_{\psi(e)} = [T_{\psi(e)}] = \Psi(e)$.

Given the primitive recursive tree $T_e$, define $\phi$ so that $W_{\phi(e)} = \mathbb{N}^* - T_e$. Then it is easy to check that $P_e = [T_e] = \mathbb{N}^{\mathbb{N}} - \mathcal{O}(W_e) = \Psi(\phi(e))$. □

Other numberings are given in the exercises.

An enumeration of the strong $\Pi^0_n$ classes can be given based on the enumeration of the $\Sigma^0_n$ sets. Let $W^n_e$ be the $e$th $\Sigma^0_n$ set. To be more precise, $W^n_e$ is the domain of the function $\phi^n_e$ where $\phi^n_e(m) = \Phi_e(m, \emptyset^{(n)})$. Then the $e$th strong $\Pi^0_{n+1}$ class is defined as follows, where we identify $\sigma \in \mathbb{N}^*$ with $\langle\sigma\rangle$ as usual for simplicity of expression.

**Definition 5.1.4.** $T^{n+1}_e = \{\sigma : (\forall \tau \preceq \sigma)\tau \in W^n_e\}$*;* $P^{n+1}_e = [T^{n+1}_e = \{x : (\forall m) x\lceil m \in W^n_e\}$*.*

Next we look at the complexity of the various notions of boundedness.

**Theorem 5.1.5.** *Let $g \geq 2$ be a computable function.*

*(a)* $\{e : T_e$ *is g-bounded*$\}$ *is* $\Pi^0_1$ *complete;*

*(b)* $\{e : T_e$ *is almost g-bounded*$\}$ *is* $\Sigma^0_2$ *complete.*

*Proof.* (a) Let $g : \mathbb{N}^* \to \mathbb{N}$ be an arbitrary function such that $g(\sigma) \geq 2$ for all $\sigma$.

The set is $\Pi^0_1$ since $T_e$ is $g$-bounded if and only if

$$(forall\sigma)(\forall i)[\sigma^\frown i \in T_e \to i < g(\sigma)].$$

For the completeness, we will define a primitive recursive function $h$ such that $T_{h(e)}$ is $g$-bounded if and only if $e \notin K$. Let

$$\sigma \in T_{h(e)} \iff (\forall t < |\sigma|)[\phi_{e,t}(e) \uparrow \to \sigma(t) = 0].$$

It follows from the Master Enumeration Theorem 1.6.5 and the s-m-n Theorem 1.6.7 that $h$ is primitive recursive. If $e \notin K$, then $T_{h(e)} = \{0^t : t \in \mathbb{N}\}$ and is clearly $g$-bounded. If $e \in K$ and $\phi_{e,t}(e) \downarrow$, then $0^{t\frown} i \in T_{h(e)}$ for all $i$, so that $T_{h(e)}$ is not $g$-bounded.

(b) This set is $\Sigma^0_2$, since if $g = \phi_a$, then $T_e$ is almost $g$-bounded if and only if

$$(\exists k)(\forall i)(\forall \sigma)[(|\sigma| \geq k \;\&\; \sigma^\frown i \in T_e) \to i < \phi_a(\sigma)].$$

For the completeness, we define a reduction of $Fin$ as follows. For each $e$ and $s$, recall that $W_{e,s} = \{i : \phi_{e,s}(i) \downarrow\}$ and that $\phi_{e,s}(i) \downarrow$ implies that $i \leq s$. Thus $e \in Fin$ if and only if $\{s : W_{e,s+1} \setminus W_{e,s} \neq \emptyset\}$ is finite. For $|\sigma| = s$, let

$$\sigma \in T_{h(e)} \iff (\forall n < s)[W_{e,n+1} \setminus W_{e,n} = \emptyset \to \sigma(n+1) < g(\sigma)].$$

If $e \in Fin$ and $k$ satisfies $W_{e,k} = W_e$, then $T_{h(e)}$ is $g$-bounded above $k$. If $e \notin Fin$, then for each $n$ such that $W_{e,n+1} \setminus W_{e,n} \neq \emptyset$, we have $0^{n\frown} i \in T_{h(e)}$ for every $i$, so that $T_{h(e)}$ is not almost bounded by $g$. □

**Theorem 5.1.6.** *(a)* $\{e : T_e$ *is c. b.*$\}$ *is* $\Sigma^0_3$ *complete.*

*(b)* $\{e : T_e$ *is almost c. b.*$\}$ *is* $\Sigma^0_3$ *complete.*

*Proof.* (a) The first set is $\Sigma^0_3$, since $T_e$ is c. b. if and only if $T_e$ is $\phi_a$-bounded for some total computable function $\phi_a$.

For the completeness, we define a reduction $f$ of $Rec$ to our set. This will be done so that $[T_{f(e)}]$ is empty if $W_e$ is finite and $[T_{f(e)}]$ has a single element if $W_e$ is infinite. The primitive recursive tree $T_{f(e)}$ is defined as follows: Put $\sigma = (s_0, s_1, \dots, s_{k-1}) \in T_{f(e)}$ if and only if $s_0 < s_1 < \dots < s_{k-1}$ and there exists a sequence $m_0 < m_1 < \dots < m_{k-1}$ such that, for each $i < k$, $m_i \in W_{e,s_i} \setminus W_{e,s_i-1}$ and $m_i$ is the least element of $W_{e,s_{k-1}} \setminus \{m_0, \dots, m_{i-1}\}$. We observe that if $W_e$ is finite, then $T_{f(e)}$ is also finite and therefore recursively bounded. Now fix $e$ and suppose that $W_e$ is infinite. Then we may define canonical sequences $n_0 < n_1 < \dots$ of elements of $W_e$ and corresponding stages $t_0 < t_1 < \dots$ such that, for each $i$, $n_i \in W_{e,t_i} \setminus W_{e,t_i-1}$ and $(t_0, t_1, \dots, t_i) \in T_{f(e)}$ as follows. Let $n_0$ be the least element of $W_e$ and, for each $k$, let $n_{k+1}$ be the least element of

$W_e \setminus W_{e,t_k}$. Then for each $k$, $(t_0, \ldots, t_k) \in T_{f(e)}$ and $n_k \in W_{e,t_k}$. Furthermore, we see by induction on $k$ that

$$k \in W_e \to k \in W_{e,t_k}.$$

For $s = 0$, this is because $n_0 = 0$ if $0 \in W_e$. Assuming the statement to be true for all $i < k$, we see that if $k \in W_e$, then either $k \in W_{e,t_{k-1}}$, or else $n_k = k$. In either case, we have $k \in W_{e,t_k}$. The key fact here is that for any $(s_0, \ldots, s_k) \in T_{f(e)}$, $s_k \leq t_k$. To see this, let $(s_0, \ldots, s_k) \in T_{f(e)}$, let $(m_0, \ldots, m_k)$ be the associated sequence of elements of $W_e$, and suppose by way of contradiction that $s_k \geq t_k$. It follows from the definitions of $T_{f(e)}$ and of $t_0, \ldots, t_k$ that in fact $s_i = t_i$ and $m_i = n_i$ for all $i \leq k$. Thus $T_{f(e)}$ has the sequence $(t_0 + 1, t_1 + 1, \ldots)$ as a bounding function.

Suppose now that $W_e$ is computable. Then the sequence $t_0 < t_1 < \ldots$ is also computable and thus $T_{f(e)}$ is computably bounded by. Now suppose that $T_{f(e)}$ is bounded by some computable function $h$. Then we must have $t_k < h(k)$ for each $k$. It follows that $k \in W_e \iff k \in W_{e,h(k)}$, so that $W_e$ is computable.

(b) This set is $\Sigma^0_3$, since $T_e$ is a. c. b. if and only if $T_e$ is $\phi_a$-almost bounded for some total computable function $\phi_a$.

For the completeness, use the argument given in (3) above. We may assume that $W_e$ is infinite, since otherwise the argument goes through trivially. Clearly, if $W_e$ is computable, then $T_{f(e)}$ is computably bounded and therefore a. c. b. as well. If $T_{f(e)}$ is almost bounded by the computable function $g$, let $k$ be large enough so that for $|\sigma| > k$, $\sigma^i \in T_{f(e)} \to i < g(\sigma)$ and let $\tau = (t_0, t_1, \ldots, t_k)$. Then we can recursively define a bounding function $h(i) \geq t(i)$ by letting $h(i) = t(i)$ for $i \leq k$ and, for each $j \geq k$,

$$h(j+1) = max\{g(\langle \tau^\frown(s_{k+1}, \ldots, s_j)\rangle) : s_i \leq h(i) \text{ for each } i \text{ with } k < i \leq j\}.$$

It follows as above that $W_e$ is computable. □

**Theorem 5.1.7.** *(a)* $\{e : T_e$ *is bounded*$\}$ *is* $\Pi^0_3$ *complete;*

*(b)* $\{e : T_e$ *is almost bounded*$\}$ *is* $\Sigma^0_4$ *complete.*

*Proof.* (a) This set is $\Pi^0_3$, since

$$T_e \text{is bounded} \iff (\forall \sigma)(\exists n)(\forall m > n)(\sigma^\frown m \notin T_e).$$

For the completeness, we define a reduction of $\omega \backslash Cof$ as follows. Let $\phi(e, m, s) =$ (least $n > m)(n \notin W_{e,s})$. This is a primitive recursive definition since $n \in W_{e,s} \to n \leq s$. Then, using the s-m-n Theorem, define the tree $T_{f(e)}$ by

$$T_{f(e)} = \{0^m : m \in \mathbb{N}\} \cup \{0^{m\frown}(s+1) : \phi(e, m, s+1) > \phi(e, m, s)\}.$$

Then $T_{f(e)}$ will be a finite-branching tree if and only if, for each $m$, there are only finitely many $s$ such that $0^{m\frown}(s+1) \in T_{f(e)}$. Now if $W_e$ is not cofinite, then for each $m$ there is a minimal $n > m$ such that $n \notin W_e$. It follows that $\lim_s \phi(e, m, s) = n$, so that $\phi(e, m, s+1) > \phi(e, m, s)$ for only finitely many

$s$, which will make $T_{f(e)}$ finite-branching. On the other hand, if $W_e$ is cofinite and we choose $m$ so that $n \in W_e$ for all $m > n$, then it is clear that there will be infinitely many $s$ such that $\phi(e, m, s+1) > \phi(e, m, s)$, so that $0^m$ will have infinitely many successors and $T_{f(e)}$ will not be finite-branching. Thus we have

$$e \notin Cof \iff T_{f(e)} \text{ is bounded.}$$

(b) This set is $\Sigma^0_4$, since

$$T_e \text{ is almost bounded } \iff (\exists k)(\forall \sigma)(\exists n)(\forall m > n)(|\sigma| > k \to \sigma^\frown m \notin T_e).$$

For the completeness, first modify the proof of part (a) by letting $T_{g(e)}$ contain $0^m$ for each $m$ together with $0^{m\frown}(s+1)$ if $m$ is the least such that $\phi(e, m, s+1) > \phi(e, m, s)\}$ This modification ensures that $T_{g(e)}$ is always almost bounded, since only for the largest $m \notin W_e$ will there be infinitely many $s$ with $0^{m\frown}s+1 \in T_{g(e)}$. By the previous argument, $T_{g(e)}$ will be bounded if and only if $e \notin Cof$. Now $S$ be an arbitrary $\Sigma^0_4$ set and suppose that $a \in S \iff (\exists k)R(a, k)$, where $R$ is $\Pi^0_3$. By the usual quantifier methods, we may assume that $R(a, k)$ implies that $R(a, j)$ for all $j > k$. By the argument above, there is a computable function $h$ such that $R(a, k)$ if and only if $T_{h(a,k)}$ is bounded and such that $T_{h(a,k)}$ is almost bounded for every $a$ and $k$. Now simply define

$$T_{\phi(a)} = \{0^n : n < \omega\} \cup \{(0^k 1)^\frown \sigma : \sigma \in T_{h(a,k)}\}.$$

If $a \in S$, then $T_{h(a,k)}$ is bounded for all but finitely many $k$ and is almost bounded for the remainder. Thus $T_{\phi(a)}$ is almost bounded. If $a \notin S$, then, for every $k$, $T_{h(a,k)}$ is not bounded, so that $T_{\phi(a)}$ is not almost bounded. □

Index sets for decidable $\Pi^0_1$ classes will use the alternate definition that $P = [T]$ for some computable tree $T$ with no dead ends.

**Theorem 5.1.8.** *(i) $\{e : P_e$ is decidable is $\Pi^0_2$ complete.*

*(ii) For any recursive $g \geq 2$, $\{e : P_e$ is decidable and $g$-bounded is $\Pi^0_1$ complete.*

*(iii) For any computable $g \geq 2$, $\{e : P_e$ is decidable and $g$-a.b. is $D^0_2$ complete.*

*(iv) $\{e : P_e$ is decidable and c. b. is $\Sigma^0_3$ complete.*

*(v) $\{e : P_e$ is decidable and almost c. b. is $\Sigma^0_3$ complete.*

*(vi) $\{e : P_e$ is decidable and bounded is $\Pi^0_3$ complete.*

*(vii) $\{e : P_e$ is decidable and almost bounded is $\Sigma^0_4$ complete.*

*Proof.* (i) This set is $\Pi^0_2$ since $T_e$ has no dead ends if and only if

$$(\forall \sigma \in T_e)(\exists i)(\sigma^\frown i \in T_e).$$

For the completeness, let $C$ be a $\Pi^0_2$ set and $R$ be a computable relation so that

$$e \in C \iff (\forall m)(\exists n)R(e,m,n).$$

Put $\emptyset$ and $(m)$ in $T_{f(e)}$ for all $m$ and, for any $k$, put $(m,n)^\frown 0^k \in T_{f(e)}$ if and only if $R(e,m,n)$. Thus $T_{f(e)}$ has no dead ends if and only if $e \in C$. Note that $T_{f(e)}$ is $g$-a.b. for any $g$.

(ii) This index set is $\Pi^0_1$, since a $g$-bounded tree $T_e$ has no dead ends if

$$(\forall \sigma)(\sigma \in T_e \rightarrow (\exists i \leq g(\sigma)(\sigma \frown i \in T_e).$$

For the completeness, observe that the proof given in Theorem 5.1.5 in fact defines a tree with no dead ends.

(iii) This index set is $D^0_2$ since the property of being $g$-bounded is $\Sigma^0_2$ and for any tree $U_e$, $T_e$ has no dead ends if and only if

$$(\forall \sigma \in T_e)(\exists i)\sigma^\frown i \in T_e.$$

For the completeness, let $A = B \cap C$, where $B$ is a $\Sigma^0_2$ set and $C$ is a $\Pi^0_2$ set. The tree $T_{j(e)}$ is constructed in two parts. First, modify the construction of part (a) by putting $1^\frown\sigma \in T_{j(e)} \iff \sigma \in T_{f(e)}$. Then $T_{f(e)} \cap I((1))$ is always $g$-a.b. and has no dead ends if and only if $e \in C$. Then we use the function $h$ defined in Theorem 5.1.7 which has the property that $e \in B$ if and only if $T_{h(e)}$ is $g$-a.b.. Note that $T_{h(e)}$ always has no dead ends. Then put $0^\frown\sigma \in T_{j(e)}$ if and only if $0^\frown\sigma \in T_{h(e)}$.

(iv) For this and the remaining cases, the upper bound on the complexity follows from part (0) above and complexity of the corresponding parts of Theorems 5.1.6 and 5.1.7. The completeness of the remaining cases follows from a simple modification of the reductions used to prove the corresponding theorems above. That is, one needs only ensure that corresponding trees used in the reductions have no dead ends. This is easily accomplished by modifying any given recursive tree $T$ to construct a new computable tree $T'$ such that (i) $0^k \in T'$ for all $k \geq 0$ and (ii) for any $n \geq 1$, $(\sigma_1, \ldots, \sigma_n) \in T$ iff $(\sigma_1 + 1, \ldots, \sigma_n + 1) \frown 0^k \in T'$ for all $k \geq 0$. $\square$

Notions of boundedness for strong $\Pi^0_2$ classes are considered in the exercises.

**Exercises**

5.1.1. The following numbering is essentially taken from Jockusch and Soare [90]. Let $\Phi_e$ be the $e$'th functional mapping $\mathbb{N}^{\mathbb{N}} \times \mathbb{N}$ to $\mathbb{N}$ and let $\Psi_2(e) = \{X \in \mathbb{N}^{\mathbb{N}} : \Phi_e(e,X) \uparrow\}$. Define transfer functions $\phi$ and $\psi$ as in Proposition 5.1.3 so that $\Psi_2(e) = P_{\psi(e)}$ and $P_e = \Psi_2(\phi(e))$. As an alternative, let $\Psi_3(e) = \{X : \Phi_e(0,X) \uparrow\}$.

5.1.2. Show that $\{e : T^2_e$ is $g$-bounded$\}$ is $\Pi^0_1$ complete and $\{e : T^2_e$ is computably bounded$\}$ is $\Sigma^0_3$ complete.

5.1.3. Show that for any $\Delta_2^0$ function $g \geq 2$, $\{e : T_e^2 \text{ is } g\text{-bounded}\}$ is $\Pi_2^0$ complete and $\{e : T_e^2 \text{ is highly bounded}\}$ is $\Sigma_4^0$ complete. (Hint: relativize from Theorems 5.1.5.

5.1.4. Give the details for the proofs of Theorem 5.1.8(4-7).

## 5.2 Cardinality

In this section we classify index sets corresponding to cardinality properties of $\Pi_1^0$ classes.

**Theorem 5.2.1.** *Let $g \geq 2$ be a computable function from $\mathbb{N}^*$ to $\mathbb{N}$.*

*(a) $\{e : P_e$ is $g$-bounded and nonempty$\}$ is $\Pi_1^0$ complete;*

*(b) $\{e : P_e$ is $g$-bounded and empty$\}$ is $D_1^0$ complete;*

*(c) $(\{e : P_e$ is $g$-bounded and nonempty$\}, \{e : P_e$ is $g$-bounded and empty$\})$ is $(\Pi_1^0, \Sigma_1^0)$ complete.*

*Proof.* We observe first that the relation $\sigma \in Ext(T_e)$ has a $\Pi_1^0$ characterization. That is,

$$\sigma \in Ext(T_e) \iff (\forall n)(\exists \tau)[|\tau| = n \;\&\; \sigma \prec \tau \;\&\; \tau \in T_e],$$

where the quantifier "$(\exists \tau)$" is bounded by $g$ in the following sense. Let $h(0) = g(\emptyset)$ and for each $n$, let $h(n+1) = max\{g(\sigma) : \sigma \in \{0, 1, \ldots, h(n)\}^n\}$. Then $\sigma \in T_e$ implies $\sigma(n) \leq h(n)$ for all $n$, so that the quantifier "$\exists \tau$" above may be replaced by "$(\exists \tau \in \{0, 1, \ldots, h(n-1)\}^n)$".

(a,c) Now $P_e$ is nonempty if and only if $\emptyset \in Ext(T_e)$.

For the double completeness, define a reduction $f$ for a given $\Pi_1^0$ set $A$ so that $P_{h(e)}$ is always $g$-bounded and is nonempty if and only if $e \in A$. Let $R$ be a computable relation so that $e \in A \iff (\forall n) R(e, n)$. Then the map may be defined by putting $0^n \in T_{f(e)} \iff R(e, n)$ and putting no other strings in $T_{h(e)}$.

(b) We see that this set is $D_1^0$ by part (a) and Theorem 5.1.5.

For the completeness, let $C = B \setminus A$, where $A$ and $B$ are $\Pi_1^0$ sets and let $R$ and $S$ be computable relations so that

$$e \in A \iff (\forall n) R(e, n) \;\text{ and } e \in B \iff (\forall n) S(e, n).$$

Then a reduction $f$ of $C$ to our set is given by putting $\sigma \in T_{f(e)}$ if and only if either

**(i)** $(\forall i < |\sigma|)[R(e, i) \;\&\; \sigma(i+1) < g(\sigma \lceil i))]$ or

**(ii)** $\sigma = (1 + g(\emptyset) + n)$ where not $S(e, n)$ and $(\forall i < n)(S(e, i))$.

Clearly $T_{f(e)}$ is $g$-bounded if and only if if $e \in B$. Similarly $T_{f(e)}$ is nonempty if and only if $e \in A$. Thus $T_{f(e)}$ is $g$-bounded and empty if and only if $e \in B - A$. $\square$

**Theorem 5.2.2.** *For any computable $g \geq 2$,*

*(a) $\{e : P_e$ is almost $g$-bounded and nonempty$\}$ is $\Sigma^0_2$ complete;*

*(b) $\{e : P_e$ is almost $g$-bounded and empty$\}$ is $D^0_2$ complete.*

*Proof.* It follows from the proof of Theorem 5.1.5 that the relation "$T_e$ is $g$-bounded above $k$" is $\Pi^0_1$. Now modify the proof of Theorem 5.2.1 to define $h$ for $\sigma \in T_e$ with $|\sigma| = k$ by $h(\sigma, k) = max\{\sigma(i) : i < k\}$ and for all $n$, $h(\sigma, k+n+1) = max\{g(\sigma) : \sigma \in \{0, 1, \ldots, h(k+n)\}^{k+n}\}$. Then for $\sigma| = k$,

$$\sigma \in Ext(T_e) \iff (\forall n)(\exists \tau \in \{0, 1, \ldots, h(\sigma, k+n)\}^{k+n})(\sigma \prec \tau \;\&\; \tau \in T_e).$$

Thus the relation "$\sigma \in T_e$" is $\Pi^0_1$ when restricted to $\sigma$ of length $k$ when $T_e$ is $g$-bounded above $k$. Then for $g$-bounded $T_e$, $P_e$ is nonempty if and only if there exists $k$ such that $T_e$ is $g$-bounded above $k$ and there exists $\sigma \in T_e$ such that $\sigma \in Ext(T_e)$. This gives the upper bound on the complexity for both parts.

(a) For the completeness, use the same reduction as in the proof of Theorem 5.1.5.

(b) For the completeness, let $A = B \cap C$, where $B$ is a $\Sigma^0_2$ set and $C$ is a $\Pi^0_2$ set. Suppose that $b \in B \iff (\exists m)(\forall n)R(b, m, n)$ and $c \in C \iff (\forall m)(\exists n)S(c, m, n)$, where $R$ and $S$ are computable. Define the function $\phi(b, n)$ to be the least $m < n$ such that $R(b, m, n)$ for all $n' < n$ (or $\phi(b, n) = n$ if there is no such $m$), so that $b \in B$ if and only if $\phi(b, n)$ is eventually constant. Define the tree $T_{f(b)}$ recursively as follows. Every string $(m)$ of length 1 is in $T_{f(b)}$. If $\sigma \in T_{f(b)}$ is of odd length $2s+1$, then $\sigma^\frown i \in T_{f(b)}$ if either $i < g(\sigma)$ or $\phi(b, s+1) > \phi(b, s)$. If $\sigma \in T_{f(b)}$ is of even length $2s+2$ and $\sigma(0) = m$, then $\sigma^\frown i \in T_{f(e)}$ if $i = 0$ and either $s < m$ or, for all $n \leq s$, $\neg S(b, m, n)$. Observe that allowing an extension when $s < m$ in the second part of the definition of $T_{f(b)}$ means that we have always have arbitrarily long strings in $T_{f(b)}$.

Suppose first that $b \in B$. Then there is some $k$ such that $\phi(b, s+1) = \phi(b, s)$ for all $s \geq k$. It follows that $T_{f(b)}$ is $g$-bounded above $k$. Next suppose that $b \in C$. Then, for any $m$, choose $n_m$ such that $S(b, m, n)$. It follows that there is no $\sigma$ of length $2n_m + 3$ beginning with $\sigma(0) = m$ in $T_{f(b)}$. It follows that $P_{f(b)}$ is empty in this case. Thus if $b \in A$, then $P_{f(b)}$ is $g$ almost bounded and is empty. If $b \notin B$, then, since $T_{f(b)}$ has arbitrarily long strings, it will not be almost bounded by $g$. If $b \notin C$, then $P_{f(b)}$ will be nonempty, since for any $m$ such that $\neg S(b, m, n)$ for all $n$, we will have $m^\frown 0^\omega \in P_{f(b)}$. $\square$

**Theorem 5.2.3.** *(a) $\{e : P_e$ is c. b. and empty$\}$ is $\Sigma^0_2$ complete;*

*(b) $\{e : P_e$ is c. b. and nonempty$\}$ is $\Sigma^0_3$ complete.*

*Proof.* (a) The case of c. b. empty classes is equivalent to bounded empty classes and is treated in Theorem 5.2.10 below.

(b) This set is $\Sigma^0_3$, since $P_e$ is c. b. and nonempty if and only if

$$(\exists a)[a \in Tot \ \&\ e \in P_e \text{ is } \phi_a\text{-bounded and nonempty})].$$

For the completeness, modify the reduction $f$ from the proof of Theorem 5.1.6 as follows. For any $\sigma = (s_0, s_1, \dots, s_{k-1}) \in T_{f(e)}$, add $\sigma^\frown 0^k$ to $T_{f'(e)}$ whenever there is no $s < k$ such that $\sigma^\frown s \in T_{f(e)}$. It is clear that $P_{f'(e)}$ will contain exactly one element for each $e$. □

**Theorem 5.2.4.** *(a) $\{e : P_e$ is bounded and empty$\}$ is $\Sigma^0_2$ complete;*

*(b) $\{e : P_e$ is bounded and nonempty$\}$ is $\Pi^0_3$ complete.*

*Proof.* (a) The case of bounded empty classes is a special one, since $[T]$ is bounded and empty if and only if $T$ is finite, that is, if and only if

$$(\exists n)(\forall \sigma)[\sigma \in T_e \rightarrow < \sigma > < n].$$

For the completeness, define a reduction $f$ of $Fin$ to by letting

$$T_{f(e)} = \{\emptyset\} \cup \{(\langle n, s\rangle) : n \in W_{e,s+1} \setminus W_{e,s}\}.$$

(b) Recall that $\{e : T_e$ is finite-branching$\}$ is $\Pi^0_3$ complete. Now if $T_e$ is finite-branching, then for any $\sigma$,

$$\sigma \in Ext(T_e) \iff (\forall i)(\exists \tau)[\sigma \prec \tau \ \&\ \tau \in T_e \ \&\ |\tau| \geq i].$$

Thus our set is $\Pi^0_3$. For the completeness, use the same reduction $f$ as given in the proof of Theorem 5.1.7, since $P_{f(e)} = \{0^\omega\}$ for every $e$. □

**Theorem 5.2.5.** *$\{e : P_e$ is a. b. and empty$\}$ and $\{e : P_e$ is a. b. and nonempty$\}$ are both $\Sigma^0_4$ complete.*

*Proof.* For the "nonempty" case, the set is $\Sigma^0_4$, since $P_e$ is a. b. and nonempty if and only if

$$(\exists k)(\exists \sigma)[B(k, e) \ \&\ |\sigma| \geq k \ \&\ \sigma \in Ext(T_e)].$$

For the completeness, use the same reduction as given in Theorem 5.1.7(b).

For the "empty" case, the set is $\Sigma^0_4$, since $P_e$ is a. b. and empty if and only if

$$(\exists k)[B(k, e) \ \&\ (\forall \sigma)(|\sigma| = k \rightarrow \sigma \notin Ext(T_e))].$$

For the completeness, modify the proof of Theorem 5.1.7(b). First define

$$T_{g'(e,k)} = \{m^i : i \leq m + k\} \cup \{(m^{m+k})^\frown s + 1) : (0^m)^\frown s + 1 \in T_{g(e)}\}.$$

where $g$ is the function defined in Theorem 5.1.7(b). Note that $[T_{g'(e,k)}]$ is always empty and that $T_{g'(e,k)}$ is always almost bounded. $T_{g'(e,k)}$ is never actually bounded because the empty string has infinitely many successors $(m)$ for each $m$. However, $T_{g'(e,k)}$ clearly has the following properties.

(i) If $e \notin Cof$, then every node except $\emptyset$ has finitely many successors.

(ii) If $e \in Cof$, then for some $m$, $m^{m+k}$ has infinitely many successors.

Now let $S$ be any $\Sigma^0_4$ set and suppose that $a \in S \iff (\exists k)R(a,k)$, where $R$ is $\Pi^0_3$. By the usual quantifier methods, we may assume that $R(a,k)$ implies that $R(a,j)$ for all $j > k$. Since $Cof$ is $\Sigma^0_3$ complete set, it follows from the above discussion above that there is a recursive function $h'$ such that $T_{h'(a,k)}$ is almost bounded for all $a, k$, $[T_{h'(a,k)}]$ is empty for all $a, k$ and

(iii) if $R(a,k)$, then every node in $T_{h'(a,k)}$ except $\emptyset$ has finitely many successors and

(iv) if $\neg R(a,k)$, then for some $m$, $m^{m+k}$ has infinitely many successors.

Now define $T_{\psi(a)} = \{(k)^\frown\sigma : \sigma \in T_{h'(a,k)}\}$. $[T_{\psi(a)}]$ is empty since each $[T_{h'(a,k)}]$ is empty. If $a \in S$, then for all but finitely many $k$, every node in $T_{h'(a,k)}$ except $\emptyset$ has finitely many successors, and for the remainder, $T_{h'(a,k)}$ is almost bounded. Thus $T_{\psi(a)}$ is almost bounded. If $a \notin S$, then, for every $k$, $T_{h'(a,k)}$ has a string of length $\geq k$ with infinitely many successors, so that $T_{\psi(a)}$ is not almost bounded. □

**Theorem 5.2.6.** *$(\{e : P_e = \emptyset\}, \{e : P_e \neq \emptyset\})$ is $(\Sigma^1_1, \Pi^1_1)$ complete.*

*Proof.* The upper bounds on the complexity follow from the fact that

$$P_e \neq \emptyset \iff (\exists x)(\forall n) x\lceil n \in T_e.$$

For the completeness, let $A$ be a $\Sigma^1_1$ set, so that, by the normal form theorem (see Hinman [80, p. 84]), there is a primitive recursive relation $R$ such that, for all $a$,

$$a \in A \iff (\exists x)(\forall n) R(a, x|n).$$

Then we may define $T_{f(a)} = \{\sigma : R(a,\sigma)\}$ by the s-m-n Theorem. Then $a \in A \iff P_{f(a)} \neq \emptyset$, as desired. □

Next we consider index sets for the cardinality of strong $\Pi^0_2$ classes.

**Theorem 5.2.7.**

*(i)* $(\{e : P^2_e$ *is g-bounded & empty*$\}), \{e : P^2_e$ *is g-bounded & nonempty*$\})$ *is* $(\Sigma^0_2, \Pi^0_2)$ *complete for any computable* $g \geq 2$.

*(ii)* $\{e : P^2_e$ *is c. b. and nonempty*$\}$ *is* $\Sigma^0_3$ *complete and* $\{e : P^2_e$ *is c. b. and empty*$\}$ *is* $\Sigma^0_2$ *complete.*

*(iii)* $\{e : P^2_e$ *is bounded and nonempty*$\}$ *is* $\Pi^0_3$ *complete and* $\{e : P^2_e$ *is bounded and empty*$\}$ *is* $\Sigma^0_2$ *complete.*

*(iv)* $\{e : P^2_e \neq \emptyset\}, \{e : P^2_e = \emptyset\})$ *is* $(\Sigma^1_1, \Pi^1_1)$ *complete.*

(v) $(\{e : P^2_e$ *is g-bounded and empty*$\}, \{e : P^2_e$ *is g-bounded and non empty*$\})$ *is* $(\Sigma^0_2, \Pi^0_2)$ *complete for any* $g \geq 2$ *which is computable in* $\mathbf{0}'$.

(vi) $\{e : P^2_e$ *is highly bounded and nonempty*$\}$ *is* $\Sigma^0_4$ *complete and* $\{e : P^2_e$ *is highly bounded and empty*$\}$ *is is* $\Sigma^0_2$ *complete.*

*Proof.* The upper bounds on the complexity are routine to check.

(i) For the completeness, we define a reduction $f$ such that $P_{2,f(e)}$ is always a class of sets and such that $e \in Inf$ if and only if $P_{2,f(e)}$ is nonempty. Simply let $0^n \in T_{2,f(e)}$ if and only if there exist $a_0 < \cdots < a_{n-1}$ each in $W_e$.

(ii,iii,iv) In each case, the completeness follows exactly as in Theorems 5.2.3, 5.2.4 .

(v.vi) These are simply relativizations of Theorems 5.2.1 and 5.2.4. □

Next we consider finite cardinality. Results related to almost boundedness are relegated to the exercises.

**Theorem 5.2.8.** *For any positive integer c and any computable function* $g \geq 2$,

(a) $(\{e : P_e$ *is g-bounded &* $|P_e| > c\}, \{e : P_e$ *is g-bounded &* $|P_e| \leq c\})$ *is* $(\Sigma^0_2, \Pi^0_2)$ *complete;*

(b) $\{e : P_e$ *is g-bounded and* $Card(P_e) = c+1\}$ *is* $D^0_2$ *complete;*

c) $\{e : P_e$ *is g-bounded and* $Card(P_e) = 1)$ *is* $\Pi^0_2$ *complete.*

*Proof.* $\{e : P_e$ is $g$-bounded and $Card(P_e) > c\}$ is $\Sigma^0_2$, since if $P_e$ is $g$-bounded, then $Card(P_e) > c)$ if and only if there exist $k$ and incomparable $\sigma_1, \sigma_2, \ldots, \sigma_{c+1} \in \omega^k$ such that each $\sigma_i \in Ext(T_e)$. For $c = 0$, this set is in fact $\Pi^0_1$ by Theorem 5.2.1. These facts imply the upper bounds on the complexity.

To prove the $\Sigma^0_2$ completeness for cardinality $> c$, we define a reduction $f$ of $\omega \setminus Tot$, as follows. For each $e$, let $\sigma = 0^{m_0}1^r0^{m_1}1^r \ldots 0^{m_{k-1}}1^r0^{m_k}1^t \in T_{f_c(e)}$ if and only if the following conditions are satisfied.

(i) $1 \leq r \leq c$ and $t \leq r$.

(ii) for each $i < k$, if $\phi_{e,|\sigma|}(i)\downarrow$, then $\phi_{e,|\sigma|}(i) = m_i$.

(iii) if $\phi_{e,|\sigma|}(k)\downarrow$, then $\phi_{e,|\sigma|}(k) \geq m_k$.

Thus if $\phi_e$ is total, then $P_{f(e)}$ has exactly $c$ elements, $0^{\phi_e(0)}1^r0^{\phi_e(1)}1^r \ldots$ for $1 \leq r \leq c$. On the other hand, if $\phi_e$ is not total, then $P_{f_c(e)}$ will be infinite. Note that the tree $T_{f_c(e)}$ is always $g$-bounded, since it is a binary tree. This reduction shows both the double completeness result as well as the completeness for cardinality $= 1$. Note that since $Tot$ is $|Pi^0_2$ complete, it follows that for any $\Pi^0_2$ set $C$, there is a reduction $h_c$ of $C$ so that $card(P_{h_c(e)}) = c$ if $e \in C$ and $P_{h_c(e)}$ is infinite otherwise.

To prove the $D^0_2$ completeness for cardinal $= c+1$, let $A = B \cap C$ where $B$ is $\Sigma^0_2$ and $C$ is $\Pi^0_2$, let $h_1$ be a reduction of $\omega - B$ (as above) so that $P_{h_1(e)}$ is infinite if $e \in B$ and $card(P_{f(e)}) = 1$ otherwise. Let $h_{c+1}$ be the reduction of $C$ described above. Then a reduction $\phi$ of $A$ to $\{e : P_e$ is $g$-bounded $= c+1\}$ may be given by defining $T_{\phi(e)} = T_{h_1(e)} \oplus T_{h_{c+1}(e)}$. □

**Remark.** It follows from Theorem 5.2.8 that $\{e : card(P_e \cap \{0,1\}^\omega) > c\}$ is $\Sigma^0_2$ complete, that $\{e : card(P_e \cap \{0,1\}^\omega) = 1\}$ and $\{e : card(P_e \cap \{0,1\}^\omega) \leq c\}$ are both $\Pi^0_2$ complete, and that $\{e : card(P_e \cap \{0,1\}^\omega) = c+1\}$ is $D^0_2$ complete.

**Theorem 5.2.9.** *For any positive integer $c$, $\{e : P_e$ is c. b. and $Card(P_e) > c\}$, $\{e : P_e$ is c. b. and $Card(P_e) \leq c\}$, and $\{e : P_e$ is c. b. and $Card(P_e) = c\}$, are all $\Sigma^0_3$ complete.*

*Proof.* The $g$-bounded case above is uniformly $\Sigma^0_2$. Then

$$P_e \text{ is c. b. } \& \ |P_e| > c \iff (\exists a)[a \in Tot \ \& \ P_e \text{is } \phi_a\text{-bounded } \& \ |P_e| > c.$$

A similar argument gives the upper bound for cardinality $\leq c$.

For the $\Sigma^0_3$ completeness of cardinality $> c$, recall the modified function $g$ from the proof of Theorem 5.2.3 such that $card(P_{g(e)}) = 1$ for all $e$ and such that $P_{g(e)}$ is c. b. if and only if $e \in Rec$. Fix $c$ and let $T$ be a binary tree such that $[T]$ has exactly $c+1$ elements. Then let $T_{k(e)} = T_{g(e)} \otimes T$. Then $k$ is a reduction of $Rec$ to $\{e : P_e$ is c. b. and $Card(P_e) > c\}$. This same reduction $k$ also works for cardinality $= c+1$ and cardinality $\leq c+1$. Note here that since $T$ is binary, $T_{g(e)} \otimes T$ will be r. b. if $T_{g(e)}$ is r. b. and since $[T]$ is nonempty, $T_{g(e)} \otimes T$ will be not r. b. if $T_{g(e)}$ is not computably bounded. □

**Theorem 5.2.10.** *For any positive integer $c$,*

*(a) $\{e : P_e$ is bounded & $|P_e| \leq c\}$ and $\{e : P_e$ is bounded & $|P_e| = 1\}$ are both $\Pi^0_3$ complete;*

*(b) $\{e : P_e$ is bounded & $|P_e| > c\}$ and $\{e : P_e$ is bounded & $|P_e| = c+1\}$ are both $D^0_3$ complete.*

*Proof.* Let us define here a $\Sigma^0_3$ relation $C(c,k,e)$ such that $C(c,k,e)$ holds iff there is a $j \geq k$ such that there exist distinct $\sigma_1, \ldots, \sigma_{c+1} \in \omega^j \cap T_e$ such that for all $n > j$ there exists $\tau_1, \ldots, \tau_{c+1} \in \omega^n \cap T_e$ which extend $\sigma_1, \ldots, \sigma_{c+1}$ respectively. Note that if $T_e$ is bounded above $k$, then $C(c,k,e)$ implies that $card(P_e) \geq c$.
$\{e : P_e$ is bounded and $Card(P_e) \leq c\}$ is $\Pi^0_3$, since

$$(P_e \text{ is bounded } \& \ |P_e| \leq c) \iff P_e \text{ is bounded } \& \ \neg C(c,0,e)].$$

The upper bounds on the complexity for the other cases follows easily.

For the completeness results, let $A = B \cap C$, where $B$ is a $\Pi^0_3$ set and $C$ is a $\Sigma^0_3$ set. It follows from Theorem 5.1.7 that there is a reduction $f$ such that $card(P_{f(e)}) = 1$ for all $e$ and such that $T_{f(e)}$ is bounded if and only if $e \in B$. This gives the $\Pi^0_3$ completeness in the cases of cardinality $\leq c$ and cardinality $= 1$.

Suppose now that $e \in C \iff (\exists m)(\forall n)(\exists k)R(e,m,n,k)$, where $R$ is computable. We will define, uniformly in $e$, a computable tree $T_{g(e)}$ such that $T_{g(e)}$ is bounded for all $e$ and such that $P_{g(e)}$ has exactly 2 elements if $e \in C$ and exactly one element ($0^\omega$) otherwise. Let $T_{g(e)}$ consist of all strings $0^m$ together with all strings $(0^m)^\frown(r+1, k_1, k_2, \dots k_n)$ such that

(i) either $m = r = 0$ or $m > 0$ and $\neg R(e, m-1, r, k)$ for all $k < n$;

(ii) for all $i \le n$, $R(e, m, i, k_i)$ and $\neg R(e, m, i, j)$ for all $j < k_i$.

Clearly $T_{g(e)}$ is always bounded and has at least one element $0^\omega$. There will be another element $(0^m)^\frown(r+1, k_1, k_2), \dots)$ when $m$ is the least number such that $(\forall n)(\exists k) R(e, m, n, k)$. Thus $e \in C$ if and only if $P_{g(e)}$ contains exactly two elements and $e \notin C$ if and only if $P_{g(e)}$ contains exactly one element. By taking a disjoint union with a fixed set containing exactly $c$ elements, we may obtain a recursive function $g_c$ such that $e \in C$ if and only if $P_{g_c(e)}$ contains exactly $c+1$ elements and $e \notin C$ if and only if $P_{g_c(e)}$ contains exactly $c$ elements.

The reduction of $A$ for cardinality $> c$ is then given by $T_{h(e)} = T_{f(e)} \otimes T_{g_c(e)}$; this also works for the case of cardinality $= c+1$. $\square$

**Theorem 5.2.11.** *For any positive integer $c$,*

*(a) $(\{e : Card(P_e) > c\}), (\{e : Card(P_e) \le c\})$ $(\Sigma^1_1, \Pi^1_1)$ complete;*

*(b) $\{e : Card(P_e) = c\}$ is $\Pi^1_1$ complete.*

*Proof.* $I_P(> c)$ is $\Sigma^1_1$ uniformly in $c$ since $e \in I_P(\ge c)$ if and only if there exist distinct $x_1, \dots, x_c \in P_e$. It then immediately follows that $I_P(\le c)$ is $\Pi^1_1$ uniformly in $c$.

For $I_P(= c)$, we recall from Theorem 1.3 that any countable $\Pi^0_1$ class contains a hyperarithmetic member. Thus we have

$$e \in I_P(= c) \iff e \in I_P(\le c) \; \& \; (\exists x_1, \dots, x_c \in HYP)(x_c \in P_e).$$

It then follows from the Spector-Gandy Theorem 1.14.5 that $I_P(= c)$ is $\Pi^1_1$.

For the completeness, let $A$ be a $\Sigma^1_1$ set and let $f$ be the function from Theorem 5.2.6 which reduces $A$ to $\{e : P_e \ne \emptyset\}$ and its complement to $\{e : P_e = \emptyset\}$. Let $T$ be a primitive recursive tree such that $card([T]) = c$. Then a reduction of $A$ to $\{e : Card(P_e) > c\}$ may be defined by $T_{g(e)} = T_{f(e)} \oplus T$ and this simultaneously reduces $\mathbb{N} \setminus A$ to $\{e : Card(P_e) \le c\}$ and in fact reduces $\mathbb{N} \setminus A$ to $\{e : Card(P_e) = c\}$. $\square$

**Theorem 5.2.12.** *Let $c$ be a positive integer.*

*(a) For any computable function $g \ge 2$,*

*(i) $(\{e : P_e$ is decidable, $g$-bounded and $Card(P_e) > c\}$ is $D^0_1$ complete;*

*(ii) $\{e : P_e$ is decidable, $g$-bounded and $Card(P_e) \le c\}$ is $\Pi^0_1$ complete;*

*(iii) $\{e : P_e$ is decidable, $g$-bounded and $Card(P_e) = c+1\}$ is $D^0_1$ complete.*

*(b) $\{e : P_e$ is decidable & $|P_e| \le c\}$, $\{e : P_e$ is decidable & $|P_e| > c\}$ and $\{e : P_e$ is decidable & $|P_e| = c\}$ are all $\Pi^0_2$ complete.*

*Proof.* For a decidable tree $T_e$, $P_e$ has $> c$ members if and only if there $T_e$ contains incomparable $\sigma_1, \ldots, \sigma_{c+1}$. The upper bounds on the complexity now follow from Theorem 5.1.8.

(a) (i) For the $D^0_1$ completeness of this set as well as the set in (iii), let $A = B \cap C$, where $B$ is a $\Pi^0_1$ set and $C$ is a $\Sigma^0_1$ set. Let $R$ be a computable relation so that $e \in C \iff (\exists n)R(e,n)$. It follows from the proof of Theorem 5.1.5 that there is a computable function $g$ such that if $e \in B$, then $P_{g(e)} = \{0^\omega\}$ and is 2-bounded and if $e \notin B$, then $P_{g(e)}$ is not $g$-bounded. Let $P_{h(e)} = P_{g(e)} \cup \{0^n 1^{i+1} 0^\omega : i < c \ \& \ R(e,n) \ \& \ (\forall m < n)\neg R(e,m)\}$.

(ii) Let $Q$ be a fixed decidable $\Pi^0_1$ class having exactly $c$ elements. For the $\Pi^0_1$ completeness of this set, just let $P_{g(e)} = P_{h(e)} \otimes Q$.

(iii) For the $\Pi^0_1$ completeness, use the reduction $h$ given in the proof of Theorem 5.1.5.

(b) For the $\Pi^0_2$ completeness when cardinality equals 1, modify the reduction given for the $\Pi^0_2$ set $C$ in the proof of part (0) of Theorem 5.1.8 by putting $\emptyset$ in $T_{f(e)}$ and by putting $(n_0, n_1, \ldots, n_{k-1}) \in T_{f(e)}$ if and only if, for all $i < k$, $n_i$ is the least such that $R(e,i,n)$. Then if $e \in C$, $T_{f(e)}$ will be a decidable tree with exactly one element and if $e \notin C$, then $T_{f(e)}$ will be a finite tree and hence will not be decidable.

For the $\Pi^0_2$ completeness in the other two sets, let $P_{g(e)} = P_{f(e)} \otimes Q$ as in (ii) above. □

There are five cases to consider when we examine the possible infinite cardinality of a $\Pi^0_1$ class $P$. $P$ may be finite, countably infinite, or uncountable. Negations of two of these adds the notions of infinite and of countable. Our first result deals with the problem of finite versus infinite sets.

**Theorem 5.2.13.**

*(a) For any computable function $g \geq 2$,*
*$(\{e : P_e$ is $g$-bounded and infinite$\}, \{e : P_e$ is $g$-bounded and finite$\})$ is $(\Pi^0_3, \Sigma^0_3)$ complete.*

*(b) $\{e : P_e$ is c. b. and infinite$\}$ is $D^0_3$ complete and $\{e : P_e$ is c. b. and finite$\}$ is $\Sigma^0_3$ complete.*

*(c) $(\{e : P_e$ is bounded and infinite$\}, \{e : P_e$ is bounded and finite$\})$ is $(\Pi^0_4, \Sigma^0_4)$ complete.*

*(d) $(\{e : P_e$ is infinite$\}, \{e : P_e$ is finite$\})$ is $(\Sigma^1_1, \Pi^1_1)$ complete.*

*Proof.* In each case, the upper bound on these complexities follows from the uniformity of previous results (Theorems 5.2.8, 5.2.9, 5.2.10 and 5.2.11, since $P_e$ is infinite if and only if $Card(P_e) > c$ for all $c$.

(a) For the completeness, we define a reduction of $Cof$ to $\{e : P_e$ is $g$-bounded and finite$\}$ which simultaneously reduces $\omega \backslash Cof$ to $\{e : P_e$ is $g$-bounded and infinite$\}$. Let

$$T_{f(e)} = \{0^n : n \in \omega\} \cup \{0^n 1 0^k : n \notin W_{e,k}\}.$$

Then $T_{f(e)}$ is always a binary tree and it is easy to see that $P_{f(e)} = \{0^\omega\} \cup \{0^n10^\omega : n \notin W_e\}$, so that $f(e) \in I_P(g\text{-bounded} < \aleph_0) \iff e \in Cof\}$.
(b) For the $\Sigma^0_3$ completeness in the finite case, use the same reduction $f$ given in part (a) above.

For the $D^0_3$ completeness in the infinite case, let $A = B \cap C$ where $B$ is a $\Sigma^0_3$ set and $C$ is a $\Pi^0_3$ set and let $f$ be the reduction given above applied to $\omega \setminus C$, so that $T_{f(a)}$ is always a binary tree and such that $P_{f(a)}$ is finite if and only if $a \notin C$. It follows from the proof of Theorem 5.2.3 that there is a recursive function $g$ such that $P_{g(a)}$ is always a singleton and is r. b. if and only if $a \in B$. Then the reduction of $A$ to $\{e : P_e \text{ is c. b. and infinite}\}$ is given by $T_{h(a)} = T_{f(a)} \otimes T_{g(a)}$.

(c) For the double completeness, let $A$ be a $\Pi^0_4$ set and let $C$ be a $\Sigma^0_3$ set so that $e \in A \iff (\forall m)\langle e, m\rangle \in C$. We may assume that if $e \notin A$, then $\langle e, m\rangle \in C$ for only finitely many $m$. Let the reduction $f$ be given by the proof of Theorem 5.2.13, so that $T_{f(e,m)}$ is always bounded and $P_{f(e,m)}$ has one element if $\langle e, m\rangle \notin C$ and has two elements if $\langle e, m\rangle \in C$. Define the reduction $h$ by $T_{h(e)} = \otimes_m T_{f(e,m)}$. Then $T_{h(e)}$ is always bounded and $card(P_{h(e)} = \prod_m card(P_{f(e,m)})$, so that if $e \notin C$, then $T_{h(e)}$ is finite, and, if $e \in C$, then $T_{h(e)}$ is uncountable.

(d) For the completeness, just let $f$ be the reduction of Theorem 5.2.6 and let $T_{h(a)} = T_{f(a)} \otimes T$, where $T$ is a primitive recursive tree such that $[T]$ is infinite. $\square$

**Remark.** It follows from part (a) that $\{e : P_e \cap \{0,1\}^\omega \text{ is infinite}\}$ is $\Pi^0_3$ complete and $\{e : P_e \cap \{0,1\}^\omega \text{ is finite}\}$ is $\Sigma^0_3$ complete.

Again we give only two cases for decidable classes.

**Theorem 5.2.14.**

*(a) For any computable function $g \geq 2$, $(\{e : P_e \text{ is } g\text{-bounded, decidable}\}, \{e : P_e \text{ is } g\text{-bounded, decidable and finite}\})$ is $(\Pi^0_2, \Sigma^0_2)$ complete.*

*(b) $\{e : P_e \text{ is decidable and infinite}\}$ is $\Pi^0_2$ complete and $\{e : P_e \text{ is decidable and finite}\}$ is $D^0_2$ complete.*

*Proof.* The upper bounds on the complexities of the classes in both parts easily follow from the uniformity of the proof of Theorem 5.2.12 as in the previous result.

For the $\Sigma^0_2$ completeness in the $g$-bounded infinite case, we define a reduction $f$ so that $P_{f(e)}$ is always 2-bounded and decidable and is finite if and only if $W_e$ is finite. Let $P_{f(e)}$ contain $0^\omega$ and, for each $m$, contain $0^m10^\omega$ if $m = [k, s]$ where $k \in W_{e,s+1} \setminus W_{e,s}$. This also gives the $\Pi^0_2$ completeness in the (unbounded) infinite case.

For the $D^0_2$ completeness in the finite case, let $A = B \cap C$ where $B$ is $\Sigma^0_2$ and $C$ is $\Pi^0_2$. Using the reduction $f$, it follows that there is a reduction $k$ of $B$ such that $P_{k(e)}$ is always 2-bounded and decidable and is finite if and only if $e \in B$. Let $h$ be the reduction from Theorem 5.2.12 so that $P_{h(e)}$ is decidable and has cardinality 1 if $e \in C$ and otherwise $P_{h(e)}$ is not decidable. Then $P_{j(e)} = P_{k(e)} \otimes P_{h(e)}$ defines a reduction of $A$ to $\{e : P_e \text{ is finite and decidable}\}$. $\square$

The other three notions of infinity produce the same level of complexity independent of the level of boundedness and of the decidability.

**Theorem 5.2.15.** *$(\{e : P_e$ is uncountable$\}, \{e : P_e$ is countable$\})$ is $(\Sigma^1_1, \Pi^1_1)$ complete and $\{e : P_e$ is countably infinite$\}$ is $\Pi^1_1$ complete and the same result holds for bounded classes, for c. b. classes, and for g-bounded classes, and also for decidable classes and for strong $Pi^0_2$ class, under each possible notion of boundedness.*

*Proof.* Recall from Theorem 5.1.7 that in each case the underlying set of $e$ such that $P_e$ is suitably bounded is a $\Sigma^0_4$ set. (For strong $\Pi^0_2$ classes, this also holds.) Then the property of being uncountable is $\Sigma^1_1$, since for any tree $T_e$, $P_e$ is uncountable if and only if $P_e$ has a perfect subset, i.e. if and only if there exists an embedding $f$ from $\{0,1\}^{<\omega}$ into $T_e$ which preserves the partial order $\prec$. It follows that the property of being countable is $\Pi^1_1$ and, by Theorem 5.2.13, that the property of being countably infinite is also $\Pi^1_1$.

For the completeness of $\{e : P_e$ is uncountable$\}$, we define a reduction of $\{e : P_e \neq \emptyset\}$ to $\{e : P_e$ is 2-bounded and uncountable$\}$ as follows. Define the binary tree $T_{f(e)}$ to consist of all strings $0^{n_0} \frown \tau_0 \frown 0^{n_1} \frown \tau_1 \frown \cdots \frown 0^{n_{k-1}} \frown \tau_{k-1} 0^n$, where $(n_0, \dots, n_{k-1}) \in U_e$ and for $i < k$, $\tau_i = (1)$ or $\tau_i = (1,1)$. Then for any path $x \in [T_e]$, $T_{f(e)}$ will contain uncountably many paths, so that if $P_e$ is nonempty, then $P_{f(e)}$ will be uncountable. If $P_e$ is empty, then every path in $P_{f(e)}$ will end in $0^\omega$, so that $P_{f(e)}$ will be countable. Note that $f$ also reduces $\{e : P_e = \emptyset\}$ to $\{e : P_e$ is 2-bounded and countable$\}$. A reduction $g$ of $\{e : P_e = \emptyset\}$ to $\{e : P_e$ is 2-bounded and countably infinite$\}$ is then given by $T_{g(e)} = T_{f(e)} \oplus T$, where $T$ is some primitive recursive binary tree with $[T]$ countably infinite.

It is clear that these reductions work for each of the notions of boundedness and also for decidable classes, since the trees constructed have no dead ends. $\square$

**Exercises**

5.2.1. A $\Pi^0_1$class $P$ is said to be *intrinsically bounded* by $g$ if the tree $T_P = \{\sigma : P \cap \sigma \neq \emptyset\}$ is $g$-bounded. Show that $\{e : P_e$ is c. b.$\}$ is $\Pi^1_1$ complete, and similarly for all other notions of boundedness.

5.2.2. Show that $\{e : P_e$ is a. c. b. and empty$\}$ and $\{e : P_e$ is a. c. b. and nonempty$\}$ are both $\Sigma^0_3$ complete. Hint: For the completeness in the "empty" case, modify the reduction $f$ from the proof of Theorem 5.1.7 by letting $T_{f'(e)}$ contain strings $(n, s_0, \dots, s_{k-1})$ such that $(s_0, \dots, s_{k-1}) \in T_{f(e)}$ and $k < n$, so that $T_{f'(e)}$ is a. r. b. if and only if $T_{f(e)}$ is a. c. b. and $P_{f'(e)}$ is always empty.

5.2.3. Let $c$ be a positive integer and $g \geq 2$ a computable function. Show that $\{e : P_e$ is $g$-almost bounded and $Card(P_e) > c\}$ is $\Sigma^0_2$ complete and both $\{e : P_e$ is $g$-almost bounded and $Card(P_e) \leq c\}$ and $\{e : P_e$ is $g$-almost bounded and $Card(P_e) = c\}$ are $D^0_2$ complete.

5.2.4. For any positive integer $c$, show that $\{e : P_e$ is a. c. b. and $Card(P_e) > c\}, \{e : P_e$ is a. c. b. and $Card(P_e) \leq c\}$, and $\{e : P_e$ is a. c. b. and $Card(P_e) = c\}$, are all $\Sigma^0_3$ complete.

5.2.5. For any positive integer $c$, show that $\{e : P_e$ is a. b. and $Card(P_e) > c\}$, $\{e : P_e$ is a. b. and $Card(P_e) \leq c\}$ and $\{e : P_e$ is a. b. and $Card(P_e) = c\}$ are all $\Sigma^0_4$ complete.

5.2.6. Show that $(\Sigma^0_1, \Pi^0_1) \leq_m (\{e : P_e$ is decidable, $g$-bounded and $Card(P_e) > c\}, \{e : P_e$ is decidable, $g$-bounded and $Card(P_e) \leq c\})$.

5.2.7. For any positive integer $c$ and any computable function $g \geq 2$,

(a) $(\{e : P^2_e$ is $g$-bounded and $Card(P_e) \leq c\})$ is $\Pi^0_3$ complete.

(b) $\{e : P^2_e$ is $g$-bounded and $Card(P^2_e) = c + 1\}$ is $D^0_3$ complete.

c) $\{e : P^2_e$ is $g$-bounded and $Card(P^2_e) = 1)$ is $\Pi^0_3$ complete.

(a) $(\{e : P^2_e$ is $g$-bounded and finite$\})$ is $\Sigma^0_4$ complete

## 5.3 Computable Cardinality

The *computable cardinality* of a class $P$ is the cardinality of the set of computable members of $P$. Also, we say that $P$ is *computably nonempty* if it has a computable member and *computably empty* otherwise. In this section, we classify the various index sets of classes with given computable cardinality conditions. The first theorem extends the result of Gasarch and Martin [72] that the property of being computably nonempty is $\Sigma^0_3$ complete for c. b. $\Pi^0_1$ classes.

**Theorem 5.3.1.** *For any computable $g \geq 2$,*

*(a) $(\{e : P_e$ is g-bounded and computably nonempty$\}, \{e : P_e$ is g-bounded and computably empty$\})$ is $(\Sigma^0_3, \Pi^0_3)$ complete*

*(b) $\{e : P_e$ is g-bounded, nonempty, and computably empty$\}$ is $\Pi^0_3$ complete.*

*Proof.* The upper bounds on the complexity follow from Theorem 5.2.1 and the fact that $P_e$ has a computable member if and only if

$$(\exists a)[a \in Tot \;\&\; (\forall n)(\phi_a | n \in U_e)].$$

For the completeness of the first two sets, we define a reduction of $Ext_2$ by letting $P_{f(a)}$ equal

$$\{x \in \{0,1\}^\omega : \phi_a \prec x\} = \{x : (\forall m)(\forall s)(\forall i)[\phi_{a,s}(m) = i \rightarrow x(m) = i]\}.$$

For the other completeness, let $Q$ be a nonempty, binary $\Pi^0_1$ class with no computable members and let $P_{h(a)} = P_{f(a)} \oplus Q$. $\square$

**Theorem 5.3.2.** *(a) $\{e : P_e$ is c. b. and computably nonempty$\}$ is $\Sigma^0_3$ complete;*

*(b)* $\{e : P_e$ *is c. b. and computably empty*$\}$ *is* $D^0_3$ *complete;*

*(c)* $\{e : P_e$ *is c. b., nonempty and computably empty*$\}$ *is* $D^0_3$ *complete.*

*Proof.* The upper bounds on the complexity are easily checked.

For the completeness in (a) use the reduction from Theorem 5.3.1. For the completeness of the other two sets, let $A = B \cap C$, where $B$ is $\Pi^0_3$ and $C$ is $\Sigma^0_3$. It follow from the proof of Theorem 5.3.1 that there is a computable function $f$ such that $P_{f(a)}$ is always c. b. nonempty and has a computable member if and only if $a \notin B$. It follows from the proof of Theorem 5.1.6 that there is a computable function $g$ such that $P_{g(a)}$ is c. b. if and only if $a \in C$. Then a reduction $h$ of $A$ to the sets in (b) and (c) may be given by $P_{h(a)} = P_{f(a)} \otimes (P_{g(a)} \cup \{0^\omega\})$. □

Proofs of the next two theorems are left for the exercises.

**Theorem 5.3.3.** *(a)* $\{e : P_e$ *is bounded and computably nonempty*$\}$ *is* $D^0_3$ *complete;*

*(b)* $\{e : P_e$ *is bounded and computably empty*$\}$ *is* $\Pi^0_3$ *complete;*

*(c)* $\{e : P_e$ *is bounded, nonempty and computably empty*$\}$ *is* $\Pi^0_3$ *complete.* □

**Theorem 5.3.4.** *(a)* $\{e : P_e$ *is and computably nonempty*$\}$ *is* $\Sigma^0_3$ *complete;*

*(b)* $\{e : P_e$ *is computably empty*$\}$ *is* $\Pi^0_3$ *complete;*

*(c)* $\{e : P_e$ *is nonempty and computably empty*$\}$ *is* $\Sigma^1_1$ *complete.* □

**Theorem 5.3.5.** *Let $c$ be a positive integer and let $g \geq 2$ be a computable function.*

*(a)* $(\{e : P_e$ *is $g$-bounded and has computable cardinality* $> c\}$,
$\{e : P_e$ *is $g$-bounded and computable cardinality* $\leq c\})$ *is* $(\Sigma^0_3, \Pi^0_3)$ *complete;*

*(b)* $\{e : P_e$ *is $g$-bounded, nonempty, and has computable cardinality* $= c\}$ *is* $D^0_3$ *complete.*

*Proof.* The upper bounds on the complexity are easily checked.

(a) For the completeness, let $f$ be the reduction from Theorem 5.3.1 such that $T_{f(a)}$ is always a binary tree and such that $P_{f(a)}$ has a computable member if and only if $a \in Ext_2$. Let $T_b$ be a fixed binary tree such that $P_b$ consists of exactly $c + 1$ computable elements. Then let $P_{h(a)} = P_{f(a)} \otimes P_b$.

(b) We begin with a construction for unbounded classes using the $\Sigma^0_3$ completeness of *Rec*.

Define the *modulus function* $\mu_a$ for the c. e. set $W_a$ by

$$\mu_a(i) = (\text{least } s)[W_a \cap \{0, 1, \ldots, i\} = W_{a,s} \cap \{0, 1, \ldots, i\}].$$

It is easy to see that $W_a$ is computable if and only if $\mu_a$ is computable. We shall define a tree $T_{f(a)}$ so that $P_{f(a)} = \{\mu_a\}$ and hence $P_a$ has exactly one computable element if and only if $a \in Rec$.

The tree $T_{f(a)}$ is defined so that a string $\sigma$ of length $n$ is in $T_{f(a)}$ if and only if each of the following three conditions is satisfied.

1. $(\forall i < n)(i \in W_{a,n} \iff i \in W_{a,\sigma(i)})$,
2. $\sigma_0 > 0 \to 0 \in W_{a,\sigma(0)} \setminus W_{a,\sigma(0)-1}$, and
3. $(\forall 0 < m < n)[\sigma(m) > \sigma(m-1) \to m \in W_{a,\sigma(m)} \setminus W_{a,\sigma(m)-1})]$.

Now let $A = B \cap C$, where $B$ is $\Sigma^0_3$ and $C$ is $\Pi^0_3$. It follows from the completeness of *Rec* and the above construction that $P_{f(a)}$ is a singleton for each $a$ and has a (unique) computable member if and only if $a \in B$. Let $h$ be the reduction from (a) such that $P_{h(a)}$ has no computable members if $a \in C$ and has at least 2 computable members if $a \notin C$. Let $S$ be a fixed class with exactly $c$ computable members. Define the computable function $\psi$ so that

$$P_{\psi(a)} = S \otimes (P_{f(a)} \oplus P_{h(a)}).$$

This gives a reduction of $A$ to $\{e : P_e \text{ has exactly } c \text{ computable members}\}$.

Finally, let $k$ be the primitive recursive function given in Theorem 2.7.7 so that for any $e$, $P_{k(e)}$ is a $\Pi^0_1$ class of sets such that there is a one-to-one correspondence between the members of $P_e$ and the computable members of $P_{k(e)}$. Then the composite function $k(\psi(a))$ gives a desired reduction of $A$ to

$$\{e : P_e \text{ is } g\text{-bounded, nonempty, and has computable cardinality} = c\},$$

so that $A$ is $D^0_3$ complete. □

The next three theorems essentially follow from the proof of Theorem 5.3.5. Details are left for the exercises.

**Theorem 5.3.6.** *Let $c$ be a positive integer.*

*(a) $\{e : P_e$ is c. b. and has computable cardinality $> c\}$ is $\Sigma^0_3$ complete;*

*(b) $\{e : P_e$ is c. b. and has computable cardinality $\leq c\}$ is $D^0_3$ complete;*

*(c) $\{e : P_e$ is c. b. and has computable cardinality $= c\}$ is $D^0_3$ complete.* □

**Theorem 5.3.7.** *Let $c$ be a positive integer.*

*(a) $\{e : P_e$ is bounded and has computable cardinality $\leq c\}$ is $\Pi^0_3$ complete;*

*(b) $\{e : P_e$ is bounded and has computable cardinality $> c\}$ is $Dp_3$ complete;*

*(c) $\{e : P_e$ is bounded and has computable cardinality $= c\}$ is $D^0_3$ complete.* □

**Theorem 5.3.8.** *Let $c$ be a positive integer.*

*(a)* $(\{e : P_e$ *has computable cardinality* $> c\}, \{e : P_e$ *has computable cardinality* $\leq c\})$ *is* $(\Sigma^0_3, \Pi^0_3)$ *complete;*

*(b)* $\{e : P_e$ *has computable cardinality* $= c\}$ *is* $D^0_3$ *complete.* □

For finite versus infinite computable cardinality, all versions of boundedness produce index sets of the same complexity (excluding almost boundedness).

**Theorem 5.3.9.** *$(\{e : P_e$ has finite computable cardinality$\}, \{e : P_e$ has finite computable cardinality$\})$ is $(\Sigma^0_4, \Pi^0_4)$ complete and the similar result is true for $g$-bounded, c. b. and bounded classes and also for strong $\Pi^0_2$ classes.*

*Proof.* The upper bounds follow from the uniformity of Theorems 5.3.5, 5.3.6, 5.3.7 and 5.3.8.

For the completeness results, let $A$ be a $\Pi^0_4$ set, so that for some $\Sigma^0_3$ relation $R$,

$$a \in A \iff (\forall i)R(i, a).$$

As usual, $R$ may be defined so that if $a \notin A$, then $R(i, a)$ for only finitely many values of $i$. By the proof of part (i) of Theorem 5.3.1, there is a computable function $f$ so that for each $a$ and $i$, $R(i, a)$ if and only if $P_{f(i,a)}$ has a computable member and $P_{f(i,a)}$ is a binary class. Now let

$$T_{\phi(a)} = \{0^n : n \geq 0\} \cup \{0^i 1 \frown \sigma : \sigma \in U_{f(i,a)}.$$

Then it is clear that $a \in A$ if and only if $P_{\phi(a)}$ has infinitely many computable members and $P_{\phi(a)}$ is always a binary class. □

Next we consider the problem of whether a $\Pi^0_1$ class has a member computable in $\mathbf{0}'$, or equivalently whether it has an element in $\Delta^0_2$. We omit the almost computably bounded classes, since by Exercise 4.1.3, an a. c. b. $\Pi^0_1$ class has a member computable in $\mathbf{0}'$ if and only if it is nonempty.

**Theorem 5.3.10.** *(a)* $(\{e : P_e$ *has a* $\Delta^0_2$ *member*$\}, \{e : P_e$ *has no* $\Delta^0_2$ *member*$\})$ *is* $(\Sigma^0_4, \Pi^0_4)$ *complete and* $\{e : P_e$ *is nonempty but has no* $\Delta^0_2$ *member*$\}$ *is* $\Sigma^1_1$ *complete.*

*(b)* $\{e : P_e$ *is bounded and has a* $\Delta^0_2$ *member*$\}$ *is* $\Sigma^0_4$ *complete and*

$$\{e : P_e \text{ is bounded and has no } \Delta^0_2 \text{ member}\}$$

*and*

$$\{e : P_e \text{ is bounded and nonempty but has no } \Delta^0_2 \text{ member}\}$$

*are both* $\Pi^0_4$ *complete.*

*Proof.* (a) By relativization of Theorem 5.0.4, the set $Tot(\mathbf{0}') = \{e : \phi^{\mathbf{0}'}_e$ is total is a $\Pi^0_3$ complete set and

$$P_e \text{ has a } \Delta^0_2 \text{ member} \iff (\exists a)[a \in Tot(0') \ \& \ (\forall n)(\phi'_a\lceil n \in T_e)].$$

The upper bounds on the complexity follow easily.

For the completeness of the first two sets, let $S$ be an arbitrary $\Sigma^0_4$ set and suppose that

$$a \in S \iff (\exists m)(\forall n)(\exists i)(\forall j)R(a,i,j,m,n),$$

for some recursive relation $R$. Define the reduction $f$ so that

$$(m, i_0, i_1, \ldots) \in P_{f(a)} \iff (\forall n)(\forall j)R(a, i_n, j, m, n).$$

It is clear that if $a \notin A$, then $P_{f(a)}$ is empty and therefore has no member computable in $\mathbf{0}'$. On the other hand, if $a \in A$, then we may choose $m$ so that $(\forall n)(\exists i)(\forall j)R(a,i,j,m,n)$ and define an infinite path $(m, i_0, i_1, \ldots) \in P_{f(a)}$ which computable in $\mathbf{0}'$ by letting $i_n$ be the least $i$ such that $(\forall j)R(a,i,j,m,n)$.

For the $\Sigma^1_1$ completeness result, let $A$ be a complete $\Sigma^1_1$ set and let $f$ be the reduction given in Theorem thm:iue so that $P_{f(a)}$ is nonempty iff $a \in A$. Then let $g$ be the computable function such that $P_{g(a)} = P_{f(a)} \oplus Q$, where $Q$ is a $\Pi^0_1$ class with no members computable in $0'$.

(b) The upper bounds on the complexity of the three sets follows as in part (a). For the completeness of the first two sets, let $h$ be the function given in Theorem 2.7.7. Then $P_e$ has a (respectively, no) $\Delta^0_2$ member if and only if $P_{h(e)}$ is bounded and has a (resp. no) $\Delta^0_2$ member.

For the $\Pi^0_4$ completeness of the third set, let $Q$ be a nonempty bounded $\Pi^0_1$ class $Q$ with no $\Delta^0_2$ members and let $P_{k(e)} = Q \oplus P_{h(e)}$. □

Finally, we consider the computable and $\Delta^0_2$ cardinality of strong $\Pi^0_2$ classes.

**Theorem 5.3.11.** *(i) For any $g \geq 2$ which is computable in $\mathbf{0}'$,*

$$(\{e : P^2_e \text{ is } g\text{-bounded and comp. empty}\}, \{e : P^2_e \text{ is } g\text{-bounded and comp. nonempty}\})$$

*is $(\Sigma^0_3, \Pi^0_3)$ complete and*

$$\{e : P^2_e \text{ is } g\text{-bounded, nonempty, and computably empty}\}$$

*is $\Pi^0_3$ complete.*

*(ii) $\{e : P^2_e$ is c. b. and computably nonempty is $\Sigma^0_3$ complete and*

$$\{e : P^2_e \text{ is c. b. and computably empty}\}$$

*and*

$$\{e : P^2_e \text{ is c. b., nonempty, and computably empty}\}$$

*are $D^0_3$ complete.*

*(iii) $\{e : P^2_e$ is bounded and computably nonempty is $D^0_3$ complete and*

$$\{e : P^2_e \text{ is bounded and computably empty}\}$$

*and*

$$\{e : P^2_e \text{ is bounded, nonempty, and computably empty}\}$$

*are $\Pi^0_3$ complete.*

*(iv)* $(\{e : P^2_e$ *is computably nonempty*$\}, \{e : P^2_e$ *is computably empty*$\})$ *is* $(\Sigma^0_3, \Pi^0_3)$ *complete and* $\{e : P^2_e$ *is nonempty but computably empty*$\}$ *is* $\Pi^0_3$ *complete.*

*(v)* $\{e : P^2_e$ *is highly bounded and computably nonempty*$\}$ *is* $\Sigma^0_4$ *complete,*

$$\{e : P^2_e \text{ is highly bounded and computably empty}\},$$

*and*

$$\{e : P^2_e \text{ is highly bounded, nonempty and computably empty}\}$$

*are all* $\Sigma^0_4$ *complete.*

*Proof.* The upper bounds on the complexity are routine. The completeness of parts (i)–(iv) follow from previous results on $\Pi^0_1$ classes. The completeness in part (v) follows easily from the $\Sigma^0_4$ completeness of the property of being highly bounded. □

**Exercises**

5.3.1. Prove Theorem 5.3.3. Hint: Combine the reductions from Theorems 5.1.7 and 5.3.1.

5.3.2. Prove Theorem 5.3.4. Hint: Combine the reductions from Theorems 5.2.6 and 5.3.1.

5.3.3. Prove Theorems 5.3.6, 5.3.7 and 5.3.8.

5.3.4. Show that the computable cardinality of a decidable class $P$ is the same as the cardinality of $P$, except when $P$ is uncountable but has a countable infinite number of computable elements. Then formulate and prove decidable versions of Theorems 5.3.5, 5.3.6, 5.3.7 and 5.3.8.

5.3.5. Prove Theorem 5.3.11(v).

5.3.6. Show that $(\{e : P^2_e$ has a $\Delta^0_2$ member$, \{e : P^2_e$ has no $\Delta^0_2$ member$\})$ is $(\Sigma^0_4, \Pi^0_4)$ complete and likewise for $g$-bounded, c. b. and bounded classes.

5.3.7. Show that $\{e : P^2_e$ is bounded and nonempty but has no $\Delta^0_2$ member is $\Pi^0_4$ complete and likewise for $g$-bounded and c. b. classes. However, for unbounded classes the corresponding set is $\Sigma^1_1$ complete.

## 5.4 Index Sets and Lattice Properties

In this section, we consider in particular the complexity of the inclusion relation and the lattice operations on the family of $\Pi^0_1$ classes.

**Lemma 5.4.1.** *There are primitive recursive functions* $\psi_i$, $\psi_u$, $\psi_s$ *and* $\psi_p$ *such that, for all a and b, (a)* $P_{\psi_i(a,b)} = P_a \cap P_b$*; (b)* $P_{\psi_u(a,b)} = P_a \cup P_b$*; (c)* $P_{\psi_s(a,b)} = P_a \oplus P_b$*; (d)* $P_{\psi_p(a,b)} = P_a \otimes P_b$

*Proof.* (a) Here we define $\psi_i$ so that $T_{\psi_i(a,b)} = T_a \cap T_b$, that is, $\pi_{\psi_i(a,b)}(\sigma) = \pi_a(\sigma) \cdot \pi_b(\sigma)$.

(b) Similarly, $T_{\psi_i(a,b)} = T_a \cup T_b$.

(c) Here $T_{\psi_i(a,b)} = \{\emptyset\} \cup \{0^\frown\tau : \tau \in T_a\} \cup \{1^\frown\tau : \tau \in T_b\}$, so that $\pi = \pi_{\psi_u(a,b)}$ is defined by $\pi(\emptyset) = 1$ and

$$\pi(\sigma) = \begin{cases} \pi_a(\sigma(1), \dots, \sigma(|\sigma| - 1)), & \text{if } \sigma(0) = 0 \\ \pi_b(\sigma(1), \dots, \sigma(|\sigma| - 1)), & \text{if } \sigma(0) = 1 \end{cases}$$

A formal definition of $\psi_u$ can now be obtained by the s-m-n Theorem.

(d) Here $T = T_{\psi_p(a,b)}$ is defined to contain $\sigma$ if and only if $(\sigma(0), \sigma(2), \dots) \in T_a$ and $\sigma(1), \sigma(3), \dots \in T_b$. Details are left to the reader. $\square$

Next we consider some aspects of the *Verification Problem* for $\Pi^0_1$ classes, that is, $\{\langle i, j\rangle : P_i \subseteq P_j\}$. This problem has been studied for various families of $\omega$-languages by Klarlund [96], Staiger [188] and others. More generally, Cenzer and Remmel [39] investigated index set problems concerning the size of the difference of two classes.

**Theorem 5.4.2.** *(Staiger) Let $g \geq 2$ be a computable function.*

*(i) $\{\langle a, b\rangle : P_a, P_b$ are $g$-bounded and $P_a \subseteq P_b\}$ and $\{\langle a, b\rangle : P_a, P_b$ are $g$-bounded and $P_a = P_b\}$ are $\Pi^0_2$ complete.*

*(ii) $\{\langle a, b\rangle : P_a, P_b$ are $g$-bounded and $P^2_a \subseteq P^2_b\}$ and $\{\langle a, b\rangle : P^2_a, P^2_b$ are $g$-bounded and $P^2_a = P^2_b\}$ are $\Pi^0_3$ complete.*

*Proof.* (i) The first set is $\Pi^0_2$ (and hence also the second set), since

$$P_a \subseteq P_b \iff (\forall\sigma)(\sigma \in Ext(T_a) \to \sigma \in T_b).$$

For the completeness, let $b$ be given so that $P_b = \{0^\omega\}$. Let $A$ be a $\Pi^0_2$ set and let $R$ be a computable relation such that, for all $i$

$$i \in A \iff (\forall m)(\exists n)R(i, m, n)$$

Define the tree $T_{f(i)}$ to contain $0^m$ for all $m$ and also

$$0^m 1^n \in T_{f(i)} \iff (\forall j < n)\neg R(i, m, j)$$

Then it is clear that

$$P_{f(i)} \subseteq P_b \iff P_{f(i)} = P_b \iff i \in A.$$

(ii) Recall that $\sigma \in T^2_e \iff (\forall\tau \preceq \sigma)\tau \in W_e)$, which is a $\Sigma^0_1$ relation. Now let $G_n = \{\sigma : |\sigma| = n \ \& \ (\forall m < n)\sigma(m) < g(m)\}$. Then

$$\sigma \in Ext(T^2_e) \iff (\forall n)(\exists\tau \in G_n)(\sigma \preceq \tau \& \ \tau \in T^2_e),$$

and this relation is $\Pi^0_2$, since the quantifier $(\exists\tau \in G_n)$ is bounded. It follows as in (i) that the two sets are $\Pi^0_3$. For the completeness, simply relativize the argument from (i). That is, let $R$ now be a $\Pi^0_1$ relation and note that the corresponding class $P^2_{f(i)}$ will indeed by a strong $\Pi^0_2$ class. $\square$

The complexity of being "almost equal" or "almost a subset" is covered by considering differences of classes. The following results are taken from [39].

**Theorem 5.4.3.** *For any computable function $g$ and any finite $k \geq 1$:*

*(i) $\{\langle a,b\rangle : P_a$ and $P_b$ are $g$-bounded and $card(P_a - P_b) \leq k\}$ is $\Pi^0_2$ complete. [(ii)] $\{\langle a,b\rangle : P^2_a$ and $P^2_b$ are $g$-bounded and $card(P^2_a - P^2_b) \leq k\}$ is $\Pi^0_2$ complete.*

*Proof.* The completeness follows from Theorem 5.2.8 in (i) and from Exercise 7 in (ii). Thus we need only see that index sets have the appropriate complexity.

(i) To see that (i) is $\Pi^0_2$, we claim that

$$card(P_a - P_b) \leq k \iff (\forall e)[P_b \cap P_e = \emptyset \rightarrow card(P_a \cap P_e) \leq k].$$

Certainly if the condition is false, then $card(P_a - P_b) > k$. On the other hand, suppose that $card(P_a - P_b) > k$. Then there are $k+1$ elements $x_0, x_1, \ldots, x_k$ in $P_a - P_b$. For each $i$, there is a basic open set $U_i$ such that $x_i \in U_i$ and $U_i \cap P_b = \emptyset$. Then $P_e = U_0 \cup \cdots \cup U_k$ contradicts the condition.

(ii) The argument is similar to (i). □

**Theorem 5.4.4.** *For any computable function $g \geq 2$:*

*(i) $\{\langle a,b\rangle : P_a$ and $P_b$ are $g$-bounded and $P_a - P_b$ is finite$\}$ is $\Sigma^0_3$ complete. [(ii)] $\{\langle a,b\rangle : P^2_a$ and $P^2_b$ are $g$-bounded and $P^2_a - P^2_b$ is finite$\}$ is $\Pi^0_4$ complete.*

*Proof.* In each case, the upper bound on the complexity follows from the uniformity of Theorem 5.4.3. The completeness follows from Theorem 5.2.13 and Exercise 3.7. □

**Theorem 5.4.5.** *For any computable function $g \geq 2$ and any finite $k \geq 0$:*

*(i) $\{\langle a,b\rangle : P_a$ and $P_b$ are $g$-bounded and $P_a - P_b$ has $\leq k$ computable members$\}$ and $\{\langle a,b\rangle : P^2_a$ and $P^2_b$ are $g$-bounded and $P^2_a - P^2_b$ has $\leq k$ computable members$\}$ are $\Pi^0_3$ complete.*

*(ii) $\{\langle a,b\rangle : P_a,\ P_b$ are $g$-bounded and $P_a - P_b$ has $< \aleph_0$ computable members$\}$ and $\{\langle a,b\rangle : P^2_a,\ P^2_b$ are $g$-bounded and $P^2_a - P^2_b$ has $< \aleph_0$ computable members$\}$ are $\Sigma^0_4$ complete.*

*Proof.* (i) The completeness follows from Theorem 5.3.5. For the upper bounds on the complexity, we claim that

$$P_a - P_b \text{ has } \leq k \text{ computable members} \iff (foralle)[P_b \cap P_e = \emptyset \rightarrow card(P_a \cap P_e) \leq k].$$

The key here is that if there are $k+1$ computable elements $x_0, x_1, \ldots, x_k$ in the difference, then $\{x_0, \ldots, x_k\}$ is a $\Pi^0_1$ class. Details are left to the reader. A similar argument covers strong $\Pi^0_2$ classes.

(ii) The upper bounds follow from the uniformity of (i) and completeness follows from Theorem 5.3.9. □

**Theorem 5.4.6.** $\{e : P_e \subseteq \{0,1\}^{\mathbb{N}}$ & $P_e$ *is thin*$\}$ *and* $\{e : P_e \subseteq \{0,1\}^{\mathbb{N}}$ & $P_e$ *is thin*$\}$ *are* $\Pi^0_4$ *complete sets.*

*Proof.* First observe that $P_e$ is minimal if and only if, for all $a$, either $P_e \cap P_a$ is finite or $P_e - P_a$ is finite. Thus the property of being minimal is $\Pi^0_4$ by Theorem 5.4.4. $P_e$ is thin if and only if, for every $a$, there exist $\sigma_1, \ldots, \sigma_k$ such that $P_e \cap P_a = P_e \cap (I(\sigma_1) \cup \cdots \cup I(\sigma_k))$. Thus the property of being thin is $\Pi^0_4$ by Theorem 5.4.2.

For the completeness, the proof of Theorem 2.8.3 may be modified for a given $\Pi^0_4$ set $C$ to define a reduction $f$ so that $P_{f(c)}$ is thin and minimal if $c \in C$ and otherwise is neither. The modification uses a uniform version of Theorem 5.2.13 that $\{e : P_e$ is finite$\}$ is $\Sigma^0_3$ complete to add a new limit point to $P$ if $c \notin C$ and otherwise to add only isolated points. That is, let $c \in C$ if and only if $P_{g(c,e)}$ is finite for all $e$. Now define the computable function $f$ so that, at each stage $s$ of the construction of $T_{f(c)}$, there is a copy of $T^s_{f(c,e)}$ below $\tau^s_e$ but not below $\tau^s_{e+1}$. If $\tau^s_e$ is abandoned, we just extend the finitely many branches with 0's. Now if $c \in C$, then only finitely many new points have been added below any $\tau_e$, so that no new limit point has been added. Then $P_{f(c)}$ will be a minimal thin class as before. If $c \notin C$, then for some $e$, we have attached an infinite $\Pi^0_1$ class, a copy of $P_{g(c,e)}$ below $\tau_e$. Thus there is a second limit point below $\tau_e$. It follows that $P_{f(c)}$ is not minimal or thin. □

**Exercises**

5.4.1. Show that $\{e : \{0,1\}^{\mathbb{N}} \subseteq P_e\}$ is $\Pi^0_1$ complete.

5.4.2. Give the details in the proof of Lemma 5.4.1.

5.4.3. Show that for any cardinal $c$, $\{\langle a,b\rangle : card(P_a \cap P_b) \leq c\}$ has the same complexity as $\{a : card(P_a) \leq c\}$ and similarly for cardinality $= c$.

5.4.4. Show that $\{\langle a,b\rangle : P_a, P_b$ are $g$-bounded and $P_a \cap P_b \neq \emptyset\}$ is $\Pi^0_1$ complete and $\{\langle a,b\rangle : P^2_a, P^2_b$ are $g$-bounded and $P^2_a \cap P^2_b \neq \emptyset\}$ is $\Pi^0_2$ complete.

5.4.5. Show that $\{\langle e,\langle\sigma\rangle\rangle : I(\sigma) \subseteq P_e\}$ and $\{\langle e,\langle\sigma\rangle\rangle : P_e$ is $g$-bounded and $I(\sigma) \subseteq P_e\}$ are $\Pi^0_1$ complete (for any computable $g$).

5.4.6. Show that $\{\langle e,\langle\sigma\rangle\rangle : I(\sigma) \subseteq P^2_e\}$ and $\{\langle e,\langle\sigma\rangle\rangle : P^2_e$ is $g$-bounded and $I(\sigma) \subseteq P_e\}$ are $\Pi^0_2$ complete (for any computable $g$).

5.4.7. Show that $\{\langle a,b\rangle : P_a \subseteq P_b\}$ is $\Pi^1_1$ complete.

5.4.8. For any computable function $g$ and any finite $k$:

(i) $\{\langle a,b\rangle : P_a$ and $P_b$ are $g$-bounded and $card(P_a - P_b) = 1\}$ is $\Pi^0_2$ complete. [(ii)] $\{\langle a,b\rangle : P_a$ and $P_b$ are $g$-bounded and $card(P_a - P_b) = k+1\}$ is $D^0_2$ complete.

5.4.9. For any computable function $g$ and any finite $k$:

(i) $\{\langle a, b\rangle : P_a^2$ and $P_b^2$ are $g$-bounded and $card(P_a^2 - P_b^2) = 1\}$ is $\Pi_3^0$ complete. [(ii)] $\{\langle a, b\rangle : P_a^2$ and $P_b^2$ are $g$-bounded and $card(P_a^2 - P_b^2) = k+1\}$ is $D_3^0$ complete.

5.4.10. For any computable function $g \geq 2$, $\{\langle a, b\rangle : P_a$ and $P_b$ are $g$-bounded and $P_a - P_b$ is countable $\}$ and $\{\langle a, b\rangle : P_a^2$ and $P_b^2$ are $g$-bounded and $P_a^2 - P_b^2$ is countable$\}$ are $\Pi_1^1$ complete.

## 5.5 Separating Classes

Recall that, for any two sets $A$ and $B$, the class $S(A, B)$ contains those *separating* sets $C$ such that $A \subset C$ and $B \cap C = \emptyset$. When $A$ and $B$ are c. e. sets, $S(A, B)$ is a $\Pi_1^0$ class of sets. There are twp special cases here. The class of supersets of $W_e$ is $S(W_e, \emptyset)$ and the class of sets disjoint from $W_e$, is $S(\emptyset, W_e)$. Note that the class $S(A, B)$ of separating sets has the following property, which we shall refer to as being *closed under between-ness*, that, for any sets $X, Y, Z$, if $X \subset Y \subset Z$ and $X, Z \in P$, then $Y \in P$.

**Lemma 5.5.1.**

1. *For any nonempty $\Pi_1^0$ class $P \subseteq \{0,1\}^{\mathbb{N}}$, the following are equivalent.*
   (a) *$P$ is the class of subsets of a $\Pi_1^0$ set $A$.*
   (b) *$P$ is the class of subsets of some set $A$.*
   (c) *$P$ is closed under subsets and under union.*
2. *For any nonempty $\Pi_1^0$ class $P \subseteq \{0,1\}^{\mathbb{N}}$, the following are equivalent.*
   (a) *$P$ is the class of supersets of a $\Sigma_1^0$ set $A$*
   (b) *$P$ is the class of supersets of some set $A$*
   (c) *$P$ is closed under supersets and under intersection.*
3. *For any $\Pi_1^0$ class $P \subseteq \{0,1\}^{\mathbb{N}}$, the following are equivalent.*
   (a) *$P$ is the class of separating sets of some pair $A, B$*
   (b) *$P$ is the class of separating sets of some pair $A, B$ of r. e. sets.*
   (c) *$P$ is closed under union, intersection and between-ness.*

*Proof.* (i) Certainly (a) implies (b) and (b) implies (c). To show that (c) implies (a), suppose that $P$ is closed under subsets and under union and let

$$A = \{n : (\exists x)[x \in A \ \& \ x(n) = 1]\}.$$

We claim that $P = \mathcal{P}(A)$. First we show by induction that $A \cap \{0, 1 \ldots, n-1\} \in P$ for all $n$. For $n = 0$, this follows from the subset property and the fact that $A$ is nonempty. Now suppose that $A \cap \{0, \ldots, n-1\} \in P$. If $n \notin A$, then

$A \cap \{0, \dots, n\} = A \cap \{0, \dots, n-1\} \in P$ by assumption. If $n \in A$, then by definition there is some $B \in P$ with $n \in B$ and then, by closure under union, $A \cup B \in P$ and by closure under subset, $A \cap \{0, \dots, n\} \in P$. Finally $P$ is a closed set and $lim_{n\to\infty} A \cap \{0, \dots, n\} = A$, so that $A \in P$ as desired.

Now let $P = \mathcal{P}(A) = [T]$ for some computable tree and recall that $Ext(T)$ is a $\Pi^0_1$ set. Then observe that $A$ may be defined by:

$$n \in A \iff \iff (\exists \sigma \in \{0,1\}^n + 1)[\sigma \in Ext(T) \ \& \ \sigma(n) = 1].$$

(ii) This is left as an exercise.

(iii) Observe that $S = S[A, B]$ if and only if $S$ is the intersection of the class of supersets of $A$ with the class of subsets of $\{0,1\}^{\mathbb{N}} - B$. Details are left as an exercise. □

**Lemma 5.5.2.** *Suppose that $P = [T]$ where $T$ is a tree with no dead ends. Then*

1. *$P$ is closed under subsets if and only if for every $\sigma \subset \tau$, if $\tau \in T$, then $\sigma \in T$.*
2. *$P$ is closed under supersets if and only if for every $\sigma \subset \tau$, if $\sigma \in T$, then $\tau \in T$.*
3. *$P$ is closed under union if and only if, for every $\sigma$ and $\tau$ in $T$, $\sigma \cup \tau \in T$.*
4. *$P$ is closed under intersection if and only if, for every $\sigma$ and $\tau$ in $T$, $\sigma \cap \tau \in T$.*

*Proof.* The proof is left as an exercise. □

**Theorem 5.5.3.** 1. *$Sub = \{e : P_e = S(\emptyset, W_b)$ for some $b\}$ is $\Pi^0_2$ complete.*

2. *$Sup = \{e : P_e = S(W_a, \emptyset)$ for some $a\}$ is $\Pi^0_2$ complete.*
3. *$Sep = \{e : P_e = S(W_a, W_b)$ for some $a, b\}$ is a $\Pi^0_2$ complete set.*

*Proof.* (i) By Lemma 5.5.1, $e \in Sub$ if and only if $P_e$ is closed under subsets and under intersection. Thus, by Lemma 5.5.2, $e \in Sub$ if and only if $P_e$ is 2-bounded and

$$(\forall \sigma, \tau \in \{0,1\}^*)[[(\sigma \subset \tau \ \& \ \tau \in Ext(T_e)) \to \sigma \in Ext(T_e)] \ \& \\ [(\sigma \in Ext(T_e) \ \& \ \tau \in Ext(T_e)) \to \sigma \cap \tau \in Ext(T_e)]].$$

For the completeness, let $A$ be a $\Pi^0_2$ set and $R$ a recursive relation so that

$$a \in A \iff (\forall m)(\exists n) R(a, m, n).$$

Define the $\Pi^0_1$ class $P_{f(a)}$ as follows:

$$x \in P_{f(a)} \iff (\forall m)[x(2m) = x(2m{+}1) = 0 \lor (x(2m) = x(2m{+}1) = 1 \ \& \ (\forall n) \neg R(a, m, n).$$

Now if $a \in A$, then $P_{f(a)} = \{0^\omega\}$ and $f(a) \in \mathit{Sub}$. If $a \notin A$, then choose $m$ such that $(\forall n)\neg R(a,m,n)$. Then $\{2m, 2m+1\} \in P_{f(e)}$, but $\{2m\} \notin P_{f(e)}$, so $f(e) \notin \mathit{Sub}$.

(ii) For any sets $B, C$,

$$C \in S(\emptyset, B) \iff C \cap B = \emptyset \iff B \subset \omega \setminus C.$$

Thus $e \in \mathit{Sub} \iff f(e) \in \mathit{Sup}$, where $\phi_{f(e)}(n) = 1 - \phi_e(n)$ and similarly $e \in \mathit{Sup} \iff f(e) \in \mathit{Sub}$. The result now follows from (i).

(iii) It follows easily from Lemma 5.5.1 that $\mathit{Sep}$ is a $\Pi^0_2$ set and the completeness follows from part (i). □

**Theorem 5.5.4.** *(i) $\{e \in \mathit{Sep} : P_e \neq \emptyset\}$ is $\Pi^0_2$ complete.*

*(ii) $\{e \in \mathit{Sep} : P_e$ is nonempty but has no computable members$\}$ is $\Pi^0_3$ complete.*

*(iii) $\{e \in \mathit{Sep} : P_e$ has a computable member$\}$ is $\Sigma^0_3$ complete.*

*Proof.* (i) This set is $\Pi^0_2$ by Theorem 5.5.3 and Theorem 5.2.1. The completeness follows by the proof of part (i) of Theorem 5.5.3.

(ii) This set is $\Pi^0_3$ by Theorem 5.5.3 and Theorem 5.3.1. For the completeness, we define a reduction of $\mathit{Rec}$ to $\mathit{Sep}$. This is done by uniformizing the proof from Odifreddi [151] of Shoenfield's theorem that every noncomputable c. e. Turing degree contains a recursively inseparable pair of c. e. sets. That is, define computable functions $f(e)$ and $g(e)$ so that

$n \in W_{f(e)} \iff (\exists s)(n)_1 \in W_{e,s+1} \setminus W_{e,s}$ & $\phi_{(n)_2,s}(n) \simeq 0$, and

$n \in W_{g(e)} \iff (\exists s)(n)_1 \in W_{e,s+1} \setminus W_{e,s}$ & $\phi_{(n)_2,s}(n) \simeq 1$.

Then $W_{f(e)}$ and $W_{g(e)}$ are a disjoint pair of c. e. sets with the following two properties:

(a) $W_{f(e)}$ and $W_{g(e)}$ have the same Turing degree as $W_e$;

(b) For any separating set $D$ such that $W_{f(e)} \subset D$ and $W_{g(e)} \cap D = \emptyset$, we have $W_e$ computable in $D$.

It follows from (a) that if $W_e$ is computable, then the pair $W_{f(e)}$ and $W_{g(e)}$ have the computable separating set $W_{f(e)}$. It follows from (b) that if $W_e$ is not computable, then there is no computable separating set for $W_{f(e)}, W_{g(e)}$. Finally, define the computable function $h$ by letting $\phi_{h(e)}(\sigma) = 1$ if and only if

$$(\forall i < |\sigma|)[(i \in W_{f(e),|\sigma|} \to \sigma(i) = 1) \;\&\; (i \in W_{g(e),|\sigma|} \to \sigma(i) = 0)]/.$$

Then we have $P_{h(e)} = S(W_{f(e)}, W_{g(e)})$. It then follows from the discussion above that

$$e \in \mathit{Rec} \iff e \in \mathit{Sep} \;\&\; P_{h(e)} \text{ has a computable member.}$$

(iii) This follows from the proof of (ii), since $P_{h(e)}$ is always a nonempty class of separating sets. □

For many applications of $\Pi^0_1$ classes, one demonstrates the difficulty of finding a solution to a certain type of computable problem by constructing c. e. sets $W_a$ and $W_b$ and a corresponding separating class $P_e = S(W_a, W_b)$ such that the set of solutions to the given problem corresponds to the class $P_e$. Thus we want to consider for a given property $\mathcal{R}$ of classes, such as the property of being finite, $\{e \in Sep : P_e \text{ has property } \mathcal{R}\}$. We note that there is a primitive recursive function $\psi$ such that $S(W_a, W_b) = P_{\psi(a,b)}$ for each $a$ and $b$ and conversely, there is a partial computable function $\phi$ such that for all $e \in Sep$, $P_e = S(W_{(\phi(e))_0}, W_{(\phi(e))_1})$.

**Theorem 5.5.5.**

*(i)* $(\{\langle a,b\rangle : S(W_a,W_b) = \emptyset\}, \{\langle a,b\rangle : S(W_a,W_b) \neq \emptyset\})$ *is* $(\Sigma^0_1, \Pi^0_1)$ *complete.*

*(ii)* *For any positive integer c,* $(\{\langle a,b\rangle : card(S(W_a,W_b)) > c\}, \{\langle a,b\rangle : card(S(W_a,W_b) \leq c\})$ *is* $(\Sigma^0_2, \Pi^0_2)$ *complete,* $\{\langle a,b\rangle : card(S(W_a,W_b)) = c+1\}$ *is* $D^0_2$ *complete and* $\{\langle a,b\rangle : card(S(W_a,W_b)) = 1\}$ *is* $\Pi^0_2$ *complete.*

*(iii)* $(\{\langle a,b\rangle : S(W_a,W_b) \textit{ is finite}\}, \{\langle a,b\rangle : S(W_a,W_b) \textit{ is infinite}\})$ *is* $(\Sigma^0_3, \Pi^0_3)$ *complete.*

*Proof.* In each case, an upper bound on the complexity is given by the reduction $\psi$ noted above together with previous results, Theorems 5.2.1, 5.2.8, 5.2.13 and 5.3.1. For the rest of the proof, we set $W_b = \emptyset$.

(i) Observe that $S(W_e, W_b)$ is empty if and only if $W_e$ is empty.

(ii) Observe that $card(S(W_e,W_b)) = 2^c$ if and only if $card(\mathbb{N} \setminus W_e) = c$. Thus only powers of 2 need to be considered. Now $e \in Tot \iff \mathbb{N} \setminus W_e = \emptyset$, which is $\iff Card(S(W_e,W_b)) = 1$ and also $\iff Card(S(W_e,W_b)) \leq 1$, which gives the completeness for cardinality $> 1$ as well. If we let $W_{\phi(e,c)} = \{n + c : n \in W_e\}$, then $card(\mathbb{N} \setminus W_e) = c + card(\mathbb{N} \setminus W_e)$. Thus $card(S(W_e, W_b) \leq 1 \iff card(W_\phi(e,c), b) \leq 2^c$ and similarly for $SS(> 2^c)$.

Next we show the $D^0_2$ completeness for cardinality $2^{c+1}$. It follows from the reduction above that, for a given $\Pi^0_2$ set $A$, there is a computable function $f$ such that if $a \in A$, then $card(\mathbb{N} \setminus W_a) = c$ and if $a \notin A$, then $card(\mathbb{N} \setminus W_a) > c$. Let $B$ be a $\Sigma^0_2$ set. We will obtain a reduction $g$ such that if $e \in B$, then $card(\mathbb{N} \setminus W_e) = 0$ and if $e \notin B$, then $card(\mathbb{N} \setminus W_e) = 1$. Of course it suffices to define such a reduction for the $\Sigma^0_2$ complete set $Fin$, which we do as follows. Given an index $e$, construct the c. e. set $W_{g(e)}$ in stages $W_{g(e),s}$ along with a number $x_s$ which is intended to be the unique member of $\mathbb{N} \setminus W_e$, if any. We assume as usual that at most one element comes into $W_e$ at any stage $s$. The construction begins with $W_{g(e),0} = \emptyset$ and $x_0 = 0$. At stage $s+1$, there are two cases.

(Case 1) If no element comes into $W_e$, or if an element $x < x_s$ comes into $W_e$, then we let $x_{s+1} = x_s$ and we put $s + 1 \in W_{g(e),s+1}$. In this case, $W_{g(e),s+1} = \{0, 1, \ldots, s+1\} \setminus \{x_s\}$.

(Case 2) If an element $x \geq x_s$ comes into $W_e$, then we put $x_s \in W_{g(e),s+1}$ and let $x_{s+1} = s+1$; in this case $W_{g(e),s+1} = \{0, 1, \ldots, s\}$.

If $W_e$ is finite, then at some stage, we obtain $x_s$ greater than every element of $W_e$, so that Case 1 applies at every later stage $t$. Thus $x_t = x_s$ for all $t > s$ and $\omega \setminus W_{g(e)} = \{x_s\}$. If $W_e$ is infinite, then Case 2 applies infinitely often and $W_{g(e)} = \omega$. Finally, we define a reduction of the $D^0_2$ set $A \cap B$ by letting $W_{h(e)} = W_{f(e)} \oplus W_{g(e)}$.

(iii) Observe that $S(W_a, W_b)$ is finite if and only if $W_a$ is cofinite and apply Theorem 5.0.4. □

**Exercises**

5.5.1. Prove Lemma 5.5.2.

5.5.2. Give the details in the proof of Lemma 5.5.1.

5.5.3. Show that that there is a primitive recursive function $\psi$ such that $S(W_a, W_b) = P_{\psi(a,b)}$ for each $a$ and $b$ and there is a partial computable function $\phi$ such that for all $e \in Sep$, $P_e = S(W_{(\phi(e))_0}, W_{(\phi(e))_1})$.

5.5.4. Show that $\{e \in Sep : card(P_e) = c + 1\}$ is $D^0_2$ complete and similarly for the $Sep$ versions of cardinality 1, finite or infinite from Theorem 5.5.5.

## 5.6 Measure and Category

In this section, we consider properties such as being perfect, being meager, and having measure $> r$ or $\geq r$ for some fixed real $r$.

Recall that a closed set $C$ is *perfect* if every element of $C$ is a limit point of $C$, that is, if $D(C) = C$. In particular, $\omega^\omega$, $\{0, 1, \ldots, k\}^\omega$ (for any $k$) and $\emptyset$ are all perfect; some authors exclude the empty set. We can use the method of Theorem 5.2.15 to classify index sets of perfect classes.

**Theorem 5.6.1.** *(i) For any computable function $g \geq 2$, $\{e : P_e$ is g-bounded and perfect$\}$ and $\{e : P_e$ is g-bounded, nonempty and perfect$\}$ are $\Pi^0_3$ complete.*

*(ii) $\{e : P_e$ is c. b. and perfect$\}$ and $\{e : P_e$ is c. b. , nonempty and perfect$\}$ are $D^0_3$ complete.*

*(iii) $\{e : P_e$ is bounded and perfect$\}$ and $\{e : P_e$ is bounded, nonempty and perfect$\}$ are $\Pi^0_4$ complete.*

*(iv) $\{e : P_e$ is perfect$\}$ and $\{e : P_e$ is nonempty and perfect$\}$ are $\Sigma^1_1$ complete.*

*Proof.* We first observe that if $P = [T]$ where $U$ is a tree with no dead ends, then $P$ is perfect if and only if $d(T) = T$. Thus $P_e$ is perfect if and only if

$$(*)(\forall \sigma)[\sigma \in Ext(T_e) \to (\exists \tau)(\exists i, j)(\sigma \prec \tau \ \&\ \tau^\frown i \in Ext(T_e) \ \&\ \tau^\frown j \in Ext(T_e))].$$

We also observe that there is a $\Pi^0_2$ relation $B \subset \mathbb{N} \times \mathbb{N}^*$ such that if $T_e$ is finite branching, then $\sigma \in Ext(T_e) \iff B(e, \sigma)$. (This is left as an exercise.)

(i) The upper bounds now follow easily from (*) and Theorem 5.2.1.

For the completeness of both index sets, modify the proof of part (i) of Theorem 5.2.13 by letting $T_{h(e)}$ contain $\{0^n : n \in \omega\}$ together with all strings $0^n 1^\frown \sigma_1 \dots \sigma_k$ where $n \notin W_{e,k}$ and each $\sigma_i$ is either (010) or (011). It is then easy to see that $T_{h(e)}$ is perfect if and only if $e \notin Cof$.

(ii) It follows from Theorem 5.1.6 and the uniform proof of part (i) that $\{e : P_e$ is c. b. and not perfect$\}$ is $\Sigma^0_3$. The upper bounds on the complexity now follow from (*).

For the completeness of each set, let $A = B \cap C$ be a $D^0_3$ set where $B$ is $\Sigma^0_3$ and $C$ is $\Pi^0_3$. It follows from the proof of part (iii) of Theorem 5.2.3 that there is a computable function $g'$ such that $P_{g'(e)}$ is always a singleton and is c. b. if and only if $e \in B$.

It follows from our proof of part (i) above that there is a comptable function $h'$ such that $P_{h'(e)}$ is always c. b. and $e \in C$ if and only if $P_{h'(e)}$ is nonempty perfect. For each $e$, let

$$P_{\phi(e)} = P_{g'(e)} \otimes P_{h'(e)}.$$

Then the desired reduction is given by

$$e \in A \iff P_{\phi(e)} \text{ is c. b. and perfect.}$$

and this also works for nonempty perfect.

(iii) The upper bound on the complexity follows from (*) and Theorem 5.2.4 as above.

For the completeness results, let $A$ be an arbitrary $\Pi^0_4$ set and let $R$ be a computable relation such that for all $a$,

$$a \in A \iff (\forall m)(\exists n)(\forall j)(\exists k) R(a, m, n, j, k).$$

We assume as usual that $R(a, m, n, j, k) \to R(a, m, n+1, j, k)$. The desired reduction of $A$ is defined as follows. First, for each $m$, $n$, and $a$, let $T_{f(m,n,a)}$ consist of all strings $(k_0 + 1, k_1 + 1, \dots, k_t + 1)$, where for each $j \le t$, $k_j$ is the least $k$ such that $R(a, m, n, j, k)$. Then let $T_{f(a)}$ contain all strings of the form $0^n$ together with all strings of the form $0^{n_0} * \sigma_0^\frown 0 * \sigma_1^\frown 0 * \dots * \sigma_r$, where for each $m \le r$, $\sigma_m \in U_{f(m,n_m,a)}$ and $n_{m+1} = |\sigma_m|$. Each $T_{f(m,n,a)}$ is finite-branching, so that $T_{f(a)}$ is always finite-branching. $0^\omega \in U_{f(a)}$, so that $P_{f(a)}$ is always nonempty. Elements of $P_{f(a)}$, other than $0^\omega$, have one of two forms:

(a) $0^{n_0} * \sigma_0^\frown 0 * \sigma_1^\frown 0 * \dots \sigma_t^\frown 0 * x$, where for each $m \le t$, $\sigma_m \in T_{f(m,n_m,a)}$ and $n_{m+1} = |\sigma_m|$ and $x \in P_{f(m+1,n_{m+1},a)}$.

(b) $0^{n_0} * \sigma_0^\frown 0 * \sigma_1^\frown 0 * \dots$, where for each $m$, $\sigma_m \in T_{f(m,n_m,a)}$ and $n_{m+1} = |\sigma_m|$.

Suppose that $a \in A$. Then for infinitely many $n$, there exists $x_n \in P_{f(0,n,a)}$ and we have $0^n * x_n \in P_{f(a)}$. Thus $0^\omega$ is not isolated. Similarly any string $\sigma = 0^{n_0} * \sigma_0^\frown 0 * \sigma_1^\frown 0 * \dots * \sigma_r \in Ext(U_{f(a)})$, will have infinitely many extensions $0^{n_0} * \sigma_0^\frown 0 * \sigma_1^\frown 0 * \dots * \sigma_r^\frown 0 * x_n$ in $P_{f(a)}$.

On the other hand, suppose that $a \notin A$ and let $M$ be the least $m$ such that $\neg(\exists n)(\forall j)(\exists k)R(a,m,n,j,k)$. Then there will be an isolated path $0^{n_0} * \sigma_0^\frown 0 * \sigma_1^\frown 0 * \cdots * \sigma_{M-2}^\frown 0 * x$ in $P_{f(a)}$, where $x \in P_{f(M-1,|\sigma_{M-2}|,a)}$.

Thus we have

$$a \in A \iff P_{f(a)} \text{ is bounded, nonempty perfect.}$$

The same reduction applies for bounded, perfect.

(iv) First define the $\Pi^1_1$ relation $Isol(x,e)$ which says that $x$ is isolated in $P_e$ by

$$Isol(x,e) \iff x \in P_e \ \& \ (\exists n)(\forall y)[(x\lceil n = y\lceil n \ \& \ x \neq y) \to y \notin P_e].$$

Next recall from Theorem 4.2.2 that every isolated point in $P_e$ must be hyperarithmetic. Thus $\{e : P_e$ is perfect is seen to be $\Sigma^1_1$ by the Spector-Gandy Theorem 1.14.5, since

$$P_e \text{ is perfect} \iff (\forall^{HYP} x)\neg Isol(x,e).$$

It follows from Theorem 5.2.6 that $\{e : P_e$ is nonempty perfect$\}$ is also $\Sigma^1_1$.

For the completeness in the nonempty perfect case, let $f$ be the reduction given in Theorem 5.2.6 so that, for an arbitrary $\Sigma^1_1$ set $A$, $P_{f(a)}$ is nonempty if and only if $a \in A$, and let $T_{g(a)} = T_{f(a)} \otimes \{0,1\}^{<\omega}$. Then $P_{g(a)}$ is nonempty perfect if $a \in A$ and is empty otherwise. For the other case, let $g$ be as above and let

$$T_{h(a)} = \{0^{n\frown}(\sigma(0)+1, \ldots, \sigma(k-1)+1) : n \in \omega \ \& \ \sigma \in T_{g(a)}\}.$$

Thus

$$P_{h(a)} = \{0^\omega\} \cup \{0^{n\frown}(x(0)+1, x(1)+1, \ldots) : n \in \mathbb{N} \ \& \ x \in P_{g(a)}\}.$$

For $a \in A$, $P_{h(a)}$ is clearly a perfect set, and for $a \notin A$, $P_{h(a)} = \{0^\omega\}$. $\square$

Next we consider the notions of category. We begin with a few definitions. A set $K \subset \mathbb{N}^{\mathbb{N}}$ is said to be *dense in* another set $M$ if $M \subset Cl(K)$. For a closed set $K$, $K$ is dense in $M$ if and only if $M \subset K$. $K$ is said to be *nowhere dense* in $\omega^\omega$ if there is no string $\sigma$ such that $K$ is dense in the interval $I(\sigma)$. Similarly, $K \subset \{0,1\}^\omega$ is nowhere dense if there is no $\sigma \in \{0,1\}^{<\omega}$ such that $K$ is dense in $I(\sigma) \cap \{0,1\}^\omega$. Thus a closed set $K$ is nowhere dense if and only if it includes no interval. Note that a nonempty open set can never be nowhere dense. A set is said to be *meager* or *first category* if it is the countable union of nowhere dense sets. A meager set includes no interval, by the Baire Category Theorem, and thus a closed meager set is itself nowhere dense. Thus a closed set $K$ is meager if and only if it includes no interval. A set is said to be *non-meager* or *second category* if it is not meager. Thus a closed set $K$ is second category if and only if it includes an interval. Note that a nonempty open set always contains an interval and thus is always non-meager. Finally, a set is said to be *comeager* if it is the complement of a meager set. It follows that a closed set $K$ is comeager if and only if $K = \omega^\omega$ (or $\{0,1\}^\omega$).

**Theorem 5.6.2.**

*(i) For all $\sigma \in \omega^{<\omega}$, $\{e : I(\sigma) \subset P_e\}$ is $\Pi^0_1$ complete and for all $\sigma \in \{0,1\}^{<\omega}$, $\{e : I(\sigma) \cap \{0,1\}^\omega \subset P_e \cap \{0,1\}^\omega\}$ is $\Pi^0_1$ complete.*

*(ii) $\{e : P_e$ is meager$\}$ and $\{e : P_e$ is meager in $\{0,1\}^{\mathbb{N}}\}$ are both $\Pi^0_2$ complete.*

*Proof.* (i) is left as an exercise.

(ii) The upper bound on the complexity follows from the fact that $P_e$ is non-meager if and only if $I(\sigma) \subset P_e$ for some $\sigma$.

For the completeness, let $A$ be a $\Sigma^0_2$ set and let $R$ be a computable relation so that

$$a \in A \iff (\exists m)(\forall n)R(m,n,a).$$

Then a reduction of $A$ to $\{e : P_e$ is meager$\}$ given by

$T_{f(a)} = \{0^m : m \in \omega\} \cup \{(0^m)^\frown 1^\frown \tau : \tau \in \{0,1\}^{<\omega}\ \&\ (\forall n < |\tau|)R(m,n,a)\}$. □

It follows that, for example, $\{e : P_e \neq \mathbb{N}^{\mathbb{N}}\}$ is $\Sigma^0_1$ complete and $\{e : P_e$ is non-meager$\}$ is $\Sigma^0_2$ complete. Also note that
$\{e : P_e$ is co-meager$\} = \{e : \mathbb{N}^{\mathbb{N}} = P_e\}$ and is $\Pi^0_1$ complete.

Next we consider the complexity of index sets associated with measure. Recall that the measure on $\{0,1\}^{\mathbb{N}}$ is defined by setting $\mu(I(\sigma) = 2^{-|\sigma|}$ and the measure on $\mathbb{N}^{\mathbb{N}}$ is defined (with $\lambda(\mathbb{N}^{\mathbb{N}}) = 1$) by setting the measure of $\{x : x(m) = n\}$ to be $2^{-n-1}$, so that $I(\sigma)$ has measure $2^{-(m_0+m_1+\cdots+m_{k-1}+k)}$. Recall from section II.1.8 that a real number $r$ is said to be $\Pi^0_1$ (respectively, $\Sigma^0_1$, etc.) if $\{q \in \mathbb{Q} : q < r\}$ is a $\Pi^0_1$ (resp. $\Sigma^0_1$, etc.) set. We note that the ordered ring $\mathbb{Q}$ of rationals is a computable structure and can be coded into $\mathbb{N}$ for computability purposes.

**Lemma 5.6.3.**

*(a) For any $\Pi^0_1$ class $P$, $\mu(P)$ is a $\Pi^0_1$ real number.*

*Proof.* Let $T$ be a computable tree such that $P = [T]$, let $\tau_0, \tau_1, \ldots$ be a computable enumeration of $\mathbb{N}^* - T$. For each $n \in \mathbb{N}$, let $K_n = \mathbb{N}^{\mathbb{N}} - \bigcup_{i \le n} I(\tau_i)$. The result now follows from the fact that $\mu(P)$ is the decreasing limit of the computable sequence $\langle \mu(K_m)\rangle_{m\in\omega}$ of dyadic rationals. □

It is an exercise to show that $\mu(P)$ need not be computable.

**Theorem 5.6.4.** *(i) For any $\Sigma^0_1$ real $r \in (0,1]$, $(\{e : \mu(P_e) < r\}, \{e : \mu(P_e) \geq r\})$ is $(\Sigma^0_1, \Pi^0_1)$ complete if $r$ is not computable, then $\{e : \mu(P_e) \leq r)$ is $\Sigma^0_1$ complete.*

*(ii) For any $\Pi^0_1$ real $r < 1$, $(\{e : \mu(P_e) > r\}, \{e : \mu(P_e) \leq r\})$ is $(\Sigma^0_2, \Pi^0_2)$ complete and $\{e : \mu(P_e) = r)$ is $\Pi^0_2$ complete. If $r$ is $\Pi^0_1$ complete, then $(\{e : \mu(P_e) < r\}, \{e : \mu(P_e) \geq r\})$ $(\Sigma^0_2, \Pi^0_2)$ complete.*

(iii) *For any $\Sigma^0_1$ real $r \in (0,1]$, $(\{e : \mu(P_e \cap \{0,1\}^{\mathbb{N}}) < r\}, \{e : \mu(P_e \cap \{0,1\}^{\mathbb{N}}) \geq r\})$ is $(\Sigma^0_1, \Pi^0_1)$ complete if $r$ is not computable, then $\{e : \mu(P_e \cap \{0,1\}^{\mathbb{N}}) \leq r)$ is $\Sigma^0_1$ complete.*

(iv) *For any $\Pi^0_1$ real $r < 1$, $(\{e : \mu(P_e \cap \{0,1\}^{\mathbb{N}}) > r\}, \{e : \mu(P_e \cap \{0,1\}^{\mathbb{N}}) \leq r\})$ is $(\Sigma^0_2, \Pi^0_2)$ complete and $\{e : \mu(P_e \cap \{0,1\}^{\mathbb{N}}) = r)$ is $\Pi^0_2$ complete. If $r$ is $\Pi^0_1$ complete, then $(\{e : \mu(P_e \cap \{0,1\}^{\mathbb{N}}) < r\}, \{e : \mu(P_e \cap \{0,1\}^{\mathbb{N}}) \geq r\})$ is $(\Sigma^0_2, \Pi^0_2)$ complete.*

*Proof.* (i) Let $\sigma_0, \sigma_1, \dots$ enumerate $\mathbb{N}^*$ and let

$$P_{e,n} = \mathbb{N}^{\mathbb{N}} - \bigcup\{I(\sigma_i) : i < n \ \&\ \sigma_i \notin T_e\}.$$

Then the function $\mu(P_{e,n})$ is computable and we have for any rational $q$:

$$\mu(P_e) \geq q \iff (\forall n)\mu(P_{e,n}) \geq q.$$

If $r$ is $\Sigma^0_1$ and *not* rational, then

$$\mu(P_e) \geq r \iff \mu(P_e) > r \iff (\forall q \in \mathbb{Q})[q < r \to \mu(P_{e,n}) \geq q.$$

For the completeness, let $A$ be a $\Pi^0_1$ set and $R$ a computable relation such that

$$a \in A \iff (\forall n)R(n, a).$$

The necessary reduction $f$ of $A$ is defined so that $P_{f(a)}) = \{0,1\}^{\mathbb{N}}$ when $a \in A$ and $P_{f(a)} = \emptyset$ if $a \notin A$. Just let The reduction $f$ is defined by $T_{f(a)} = \{\sigma : (\forall n < |\sigma|)R(n,a)\}$.

(ii) Let $r$ be a $\Pi^0_1$ real. Then we have

$$\mu(P_e) \leq r \iff (\forall q \in \mathbb{Q})(q \leq \mu(P_e) \to q \leq r)$$

and similarly

$$\mu(P_e) \geq r \iff (\forall q \in \mathbb{Q})(q \leq r \to q \leq \mu(P_e)).$$

It follows that $\{e : \mu(P_e) \leq r\}$, $\{e : \mu(P_e) \geq r\}$, and $\{e : \mu(P_e) = r\}$ are all $\Pi^0_2$ sets. Next we show the completeness of the latter two sets. Let $B$ be a $\Pi^0_1$ set so that $r = \sum_{i \in B} 2^{-i-1}$ and let $P_B = \{0^\omega\} \cup \bigcup_{i \in B} I(0^i 1)$, so that $\mu(P_B) = r$. Since $r \neq 1$, we may assume that $B$ is co-infinite. Let $A$ be a $\Pi^0_2$ set and $R$ a computable relation so that

$$a \in A \iff (\forall m)(\exists n)R(m,n,a).$$

Here we assume as usual that if $a \notin A$, then $(\exists n)R(m,n,a)$ for only finitely many $m$. Now define the reduction $g$ by

$$P_{g(a)} = \{0^\omega\} \cup \bigcup\{I(0^m 1) : i \in B \ \text{ or } \ (\forall n)\neg R(m,n,a)\}.$$

If $a \in A$, then clearly $P_{g(a)} = P_B$ so that $\mu(P_{g(a)}) = r$. If $a \notin A$, then $P_{g(a)}$ includes $P_B$ together with cofinitely many intervals $I((m))$, so that $\mu(P_{g(a)}) > r$.

For the completeness of measure $\geq r$ when $r$ is $\Pi^0_1$ complete, let $B$ be a $\Pi^0_1$ set such that $\mu(P_B) = r$. Let $A$ be a $\Pi^0_2$ set and, by the completeness, let $f$ be a computable functions such that, for any $a$,

$$a \in A \iff (\forall m) f(a,m) \notin B.$$

Define the uniformly $\Pi^0_1$ set $C_a = B \setminus \{f(a,m) : m \in \mathbb{N}\}$, so that for any $a$, we have $a \in A \iff C(a) = B$ and otherwise, $C_a$ is a proper subset of $B$. Then define

$$P_{g(a)} = \{0^\omega\} \cup \bigcup \{I((0^n 1)) : n \in C(a)\}.$$

If $a \in A$, then $P_{g(a)} = P_B$, so that $\mu(P_{g(a)}) = r$ and if $a \notin A$, then $P_{g(a)}$ is a subset of $P_B - I(0^n 1)$ for some $n \in B$ and thus $\mu(P_{g(a)}) < r$.

Parts (iii) and (iv) follow immediately. □

**Exercises**

5.6.1. Define a $\Pi^0_2$ relation $B \subset \mathbb{N} \times \mathbb{N}^*$ such that if $T_e$ is finite branching, then $\sigma \in Ext(T_e) \iff B(e,\sigma)$.

5.6.2. For any computable function $g \geq 2$, show that $\{e : P_e$ is $g$-a.b. and perfect$\}$ and $\{e : P_e$ is $g$-a.b., nonempty and perfect$\}$ are $\Pi^0_3$ complete.

5.6.3. $\{e : P_e$ is a.c.b. and perfect$\}$ and $\{e : P_e$ is a.c.b., nonempty and perfect$\}$ are $D^0_3$ complete.

5.6.4. $\{e : P_e$ is a.b. and perfect$\}$ and $\{e : P_e$ is a.b., nonempty and perfect$\}$ are $D^0_4$ complete.

5.6.5. Prove part (i) of Theorem 5.6.2.

5.6.6. Define a $\Pi^0_1$ class $P$ such that $\mu(P)$ is a $\Pi^0_1$ complete real.

## 5.7 Derivatives

In this section we consider the uniform (arithmetic) complexity of $D^\alpha(P_e)$ and the complexity of various cardinality properties of $D^\alpha(P_e)$. These problems were first studied in the context of Polish spaces by Kuratowski, see [109], where the Cantor-Bendixson derivative is viewed as a mapping from the space of compact subsets of $\{0,1\}^\omega$ to itself. Kuratowski showed that the derivative is a Borel map of class exactly two. In particular, he showed that the family $D^{-1}(\{\emptyset\})$ of finite closed sets is a universal $\mathbf{\Sigma}^0_2$ class and posed the problem of determining the exact Borel class of the iterated operator $D^\alpha$. Cenzer and Mauldin showed in [26, 27] and that the iterated operator $D^n$ is of Borel class exactly $2n$ for finite $n$ and that for any limit ordinal $\lambda$ and any finite $n$, $D^{\lambda+n}$ is of Borel class exactly $\lambda + 2n + 1$. In particular it is shown that for any $\alpha$, the family $T_\alpha$ of closed sets $K$ such that $D^\alpha(K) = \emptyset$ is a universal $\mathbf{\Sigma}^0_{2\alpha}$ set. Lempp [115] gives an effective version of this result.

We first observe that the basic results on the cardinality of $\Pi^0_1$ classes can be relativized. For any fixed set $X$, let $P_e^X$ enumerate the binary classes which are $\Pi^0_1$ in $X$. That is, let $P_e^X = [T_e^X]$, where

$$T_e^X = \{\sigma : (\forall \tau \preceq \sigma)(\langle e, \tau\rangle \notin W_e^X\}.$$

**Theorem 5.7.1.** *For any set $X$,*

1. *$(\{e : P_e^X \text{ is empty}\}, \{e : P_e^X \text{ is nonempty}\})$ is $({\Sigma^0_1}^X, {\Pi^0_1}^X)$ complete,*
2. *$\{e : card(P_e^X) = 1\}$ is ${\Pi^0_2}^X$ complete.*
3. *For any positive integer $c$, $(\{e : card(P_e^X) > c\}, \{e : card(P_e^X) \leq c\})$ is $({\Sigma^0_2}^X, {\Pi^0_2}^X)$ complete and $\{e : card(P_e^X) = c+1\}$ is ${D^0_2}^X$ complete.*
4. *$(\{e : P_e^X \text{is finite}\}, \{e : P_e^X \text{is infinite}\})$ is $({\Sigma^0_3}^X, {\Pi^0_3}^X)$ complete.*

*Proof.* This follows from the proofs of Theorems 5.2.1, 5.2.8 and 5.2.13. □

The strong $\Pi^0_n$ classes were defined in Section III.2.3. Here we need a uniform definition of the strong $\Pi^0_\beta$ classes for any computable ordinal $\beta$. Let

$$T_{e,\alpha} = \{\sigma : (\forall \tau \preceq \sigma)(\langle e, \tau\rangle \notin \mathbf{O}^\alpha\}.$$

and let

$$P_{e,\alpha} = [T_{e,\alpha}].$$

Then a closed set $P$ is said to be $\Pi^0_\alpha$ if it equals $P_{e,\alpha}$ for some index $e$. It follows that $P$ is $\Pi^0_{\alpha+1}$ if and only if $P$ is $\Pi^0_1$ in $\mathbf{O}^\alpha$. Furthermore, for any ordinal $\beta$, $P$ is a strong $\Pi^0_\beta$ class if and only if $T_P$ is a $\Pi^0_\beta$ set. (See the exercises.)

The following result from [35] is now immediate for successor ordinals.

**Theorem 5.7.2.** *For any computable ordinal $\alpha$,*

1. *$(\{e : P_{e,\alpha+1} \text{ is empty}\}, \{e : P_{e,\alpha+1} \text{ is nonempty}\})$ is $(\Sigma^0_{\alpha+1}, \Pi^0_{\alpha+1})$ complete,*
2. *$\{e : card(P_{e,\alpha+1}) = 1\}$ is $\Pi^0_{\alpha+2}$ complete.*
3. *For any positive integer $c$, $(\{e : card(P_{e,\alpha+1}) > c\}, \{e : card(P_{e,\alpha+1}) \leq c\})$ is $(\Sigma^0_{\alpha+2}, \Pi^0_{\alpha+2}$ complete and $\{e : card(P_{e,\alpha+1}) = c+1\}$ is $D^0_{\alpha+2}$ complete.*
4. *$(\{e : P_{e,\alpha+1} \text{is finite}\}, \{e : P_{e,\alpha+1} \text{is infinite}\})$ is $(\Sigma^0_{\alpha+3}, \Pi^0_{\alpha+3})$ complete.*

We need a uniform version of Lemma V.4.4.1.

**Lemma 5.7.3.** *[[35]] For any computable limit ordinal $\lambda$ and any finite $n > 0$,*

*(a) $\{\langle e, \sigma\rangle : \sigma \in d^n(T_e \cap \{0,1\}^*)\}$ is $\Sigma^0_{2n}$;*

*(b)* $\{\langle e,\sigma\rangle : \sigma \in d^{\lambda}(T_e \cap \{0,1\}^*)\}$ *is* $\Pi^0_{\lambda+1}$*;*

*(c)* $\{\langle e,\sigma\rangle : \sigma \in d^{\lambda+n}(T_e \cap \{0,1\}^*)\}$ *is* $\Sigma^0_{\lambda+2n}$*.*

By the uniform proof of Theorem V.4.4.8, we have the following.

**Theorem 5.7.4.** *There is a primitive recursive function $\phi$ such that, for any computable ordinal $\alpha$, if $Q$ is the $\Pi^0_{2\alpha+1}$ class with index $e$, then $P_{\phi(e)}$ is the index of a $\Pi^0_1$ class $P$ of sets such that there is a homeomorphism $H$ from $Q$ onto $D^{\alpha}(P)$ with $x \leq_T H(x) \leq x \oplus 0^{2\alpha-1}$ for all $x \in Q$.*

**Theorem 5.7.5.** *For any recursive ordinal $\alpha$,*

1. *$(\{e : D^{\alpha}(P_e)$ is empty$\}, \{e : D^{\alpha}(P_e)$ is nonempty$\})$ is $(\Sigma^0_{2\alpha+1}, \Pi^0_{2\alpha+1})$ complete.*

2. *$\{e : card(D^{\alpha}(P_e)) = 1)\}$ is $\Pi^0_{2\alpha+2}$ complete.*

3. *For any positive integer c, $(\{e : card(D^{\alpha}(P_e)) \leq c), \{e : card(D^{\alpha}(P_e)) > c\})$ is $(\Sigma^0_{2\alpha+2}, \Pi^0_{2\alpha+2})$ complete and $\{e : card(D^{\alpha}(P_e)) = c+1)$ is $D^0_{2\alpha+2}$ complete.*

4. *$(\{e : D^{\alpha}(P_e)$ is finite$\}, \{e : D^{\alpha}(P_e)$ is infinite$\})$ is $(\Sigma^0_{2\alpha+3}, \Pi^0_{2\alpha+3})$ complete.*

*Proof.* The upper bound on the complexity follows from Lemma 5.7.3 and Theorem 5.7.2. That is, for example, fix $\alpha = \lambda + n$, where $\lambda$ is a limit and $n > 0$. Then $D^{\alpha}(P_e) = [d^{\lambda+n}(T_e)]$ and it follows from Lemma 5.7.3 that this equals $P_{f(e),\lambda+2n+1}]$ for some computable function $f$. Since $\lambda + 2n + 1 = 2\alpha + 1$, the complexity follows from Theorem 5.7.2.

The completeness follows from Theorems 5.7.2 and 5.7.5. That is, for example, $P_{e,\lambda+1}$ is finite if and only if $D^{\lambda}(P_{f(e)})$ is finite, and $\{e : P_{e,\lambda+1}$ is finite$\}$ is $\Sigma^0_{\lambda+3}$ complete, therefore $\{e : D^{\lambda}(P_{f(e)})$ is finite$\}$ is also $\Sigma^0_{\lambda+3}$ complete. $\square$

Lempp used different methods in [115] to prove parts (i) and (iv). He gave weaker versions of parts (ii) and (iii), showing that
$(\Sigma^0_{2\alpha+1}, \Pi^0_{2\alpha+1}) \leq (I^{(\alpha)}_P(\text{empty}), I^{(\alpha)}_P(=1))$.

We now consider the complexity of the perfect kernel $K(P_e)$. It follows from Theorem 5.2.15 that $\{e : K(P_e) = \emptyset\}$ is $\Pi^1_1$ complete. It follows from Theorem 5.7.5 that, for every computable ordinal $\alpha$, there exists $e$ such that $D^{\alpha}(P_e)$ is nonempty but $D^{\alpha+1}(P_e) = \emptyset$. This gives us the following.

**Theorem 5.7.6.** *There is a $\Pi^0_1$ class $P \subseteq \{0,1\}^{\mathbb{N}}$ such that*

*(i)* $rk(P) = \omega_1^{C-K}$.

*(ii)* $T_{K(P)}$ *is* $\Sigma^1_1$ *complete, that is,* $\{\sigma : K(P) \cap I(\sigma) \neq \emptyset\}$ *is* $\Sigma^1_1$ *complete.*

*Proof.* Let

$$P = \bigcup_e \{0^e 1x : x \in P_e\}.$$

Then

$$K(P_e) = \{0^\omega \cup \bigcup_e \{0^e 1x : x \in K(P_e)\}$$

and, for each $\alpha$,

$$D^\alpha(P) = \{0^\omega \cup \bigcup_e \{0^e 1x : x \in D^\alpha(P_e)\}.$$

We know that $rk(P) \leq \omega_1^{C-K}$ by Theorem V.4.1.4 and it now follows from Theorem 5.7.5 that $rk(P) = \omega_1^{C-K}$. It also follows from TheoremV.4.1.4 that $K(P)$ is a $\Sigma_1^1$ class, so that $T_{K(P)}$ is a $\Sigma_1^1$ set. For the completeness, observe that $K(P_e) \neq \emptyset$ if and only if $P_e$ is uncountable and that $\{e : P_e \text{ is uncountable}\}$ is $\Sigma_1^1$ complete by Theorem 5.2.15. Then we have

$$K(P_e) \neq \emptyset \iff K(P) \cap I(0^e 1) \neq \emptyset,$$

which shows that $K(P)$ is $\Sigma_1^1$ complete. □

**Exercises**

5.7.1. Give a careful proof of Lemma 5.7.3.

5.7.2. Show that, for any ordinal $\beta$, $P$ is a strong $\Pi_\beta^0$ class if and only if $T_P$ is a $\Pi_\beta^0$ set.

## 5.8 Index Sets for Logical Theories

In this section we define index sets for (propositional) logical theories and consider the complexity of properties associated with the consistency and completeness of such theories.

Let *Sent* denote the set of sentences $\{\gamma_0, \gamma_1, \dots\}$ of the propositional language with variables $\{A_0, A_1, \dots\}$, enumerated first by length and then lexicographically. The $e$'th axiomatizable theory $\Gamma_e \subseteq Sent$ may be defined as the set of consequences of $\{\gamma_i : i \in W_e\}$. The following lemma is left as an exercise.

**Lemma 5.8.1.** *$\{i : \gamma_i \in \Gamma_e\}$ is a c. e. set and in fact there is a computable function $f$ such that $\{i : \gamma_i \in \Gamma_e\} = W_{f(e)}$.*

As in section III.2.9, a $\Pi_1^0$ class $P \subseteq \{0,1\}^{\mathbb{N}}$ represents a class $G$ of subsets of *Sent* if, for any $x \in TN$, $x \in P$ if and only if $\{A_i : x(i) = 1\} \in G$.

The next result now follows easily from the uniformity of the proof of Theorem 2.9.1.

**Lemma 5.8.2.** *There is a primitive recursive function $f$ such that, for all $e$, $P_{f(e)}$ represents the set of complete consistent extensions of $\Gamma_e$. Furthermore, if $\Gamma_e$ is a decidable theory, then $P_{f(e)}$ is a decidable $\Pi^0_1$ class, that is, $\{\sigma : P_{f(e)} \cap I(\sigma) \neq \emptyset\}$ is a computable set.*

Note here that when $\Gamma_e$ is decidable, we do not necessarily have $T_{f(e)}$ to be a tree without dead ends; there simply *exists* a tree $T$ without dead ends such that $P_{f(e)} = [T]$.

On the other hand, Theorem 2.9.3 may be uniformized as follows.

**Lemma 5.8.3.** *There is a primitive recursive function $g$ such that, for all $e$, $P_e$ represents the set of complete consistent extensions of $\Gamma_{g(e)}$. Furthermore, if $P_e$ is a decidable $\Pi^0_1$ class, then $\Gamma_{g(e)}$ is a decidable theory.*

Note again that when $P_e$ is decidable (which is true whenever $T_e$ has no dead ends), then $\Gamma_{g(e)}$ is a decidable theory but it not necessarily true that $W_{g(e)}$ is a computable set.

We can now apply the index set result sets of this chapter to obtain some complexity results for axiomatizable theories.

**Theorem 5.8.4.** *1. $\{e : \Gamma_e$ is consistent$\}$ is $\Pi^0_1$ complete.*

*2. $\{e : \Gamma_e$ is consistent and complete$\}$ is $\Pi^0_2$ complete.*

*3. $\{e : \Gamma_e$ is essentially undecidable$\}$ is $\Pi^0_3$ complete.*

*Proof.* (1) Using the function $f$ from Lemma 5.8.2, $\Gamma_e$ is consistent if and only if $P_{f(e)}$ is nonempty, and this is a $\Pi^0_1$ condition by Theorem 5.2.1. For the completeness, $P_e$ is nonempty if and only if $\Gamma_{g(e)}$ is consistent, where $g$ is the function from Lemma 5.8.3. The completeness now follows from Theorem 5.2.1.

(2) $\Gamma_e$ is consistent and complete if and only if it has a unique complete consistent extension, that is, if and only if $card(P_{f(e)}) = 1$, which is a $\Pi^0_2$ condition by Theorem 5.2.8. The completeness follows from Theorem 5.2.8 since $card(P_e) = 1$ if and only if $\Gamma_{g(e)}$ is consistent and complete.

(3) $\Gamma_e$ is essentially undecidable if and only if it has no computable complete consistent extension, that is, if and only if $P_{f(e)}$ has no computable element, which is a $\Pi^0_3$ complete condition by Theorem 5.3.1. The completeness follows from Theorem 5.3.1 since $P_e$ has no computable element if and only if $\Gamma_{g(e)}$ is essentially undecidable. $\square$

We can also classify the index sets of theories with a given number of complete consistent extensions (and similarly for computable complete consistent extensions). The next theorem follows from Theorems 5.2.8, 5.2.13 and 5.2.15 as above. Let us abbreviate "computable consistent extensions" by CCEs.

**Theorem 5.8.5.** *Let $c > 0$ be finite.*

1. *$(\{e : \Gamma_e$ has $> c$ CCEs$\}, \{e : \Gamma_e$ has $\leq c$ CCEs$\}$ is $(\Sigma^0_2, \Pi^0_2)$ complete.*

2. *$\{e : \Gamma_e$ has exactly $c$ CCEs$\}$ is $D^0_2$ complete.*

3. *($\{e : \Gamma_e$ has finitely many CCEs$\}, \{e : \Gamma_e$ has infinitely many CCEs$\}$ is $(\Sigma^0_3, \Pi^0_3)$ complete.*

4. *$\{e : \Gamma_e$ has exactly $\aleph_0$ CCEs$\}$ is $\Pi^1_1$ complete.*

5. *($\{e : \Gamma_e$ has uncountably many CCEs$\}, \{e : \Gamma_e$ has countably many CCEs$\}$ is $(\Sigma^1_1, \Pi^1_1)$ complete.*

For a given number of computable complete consistent extensions, we apply Theorems 5.3.5 and 5.3.9.

**Theorem 5.8.6.** *Let $c > 0$ be finite.*

1. *($\{e : \Gamma_e$ has $> c$ computable CCEs$\}, \{e : \Gamma_e$ has $\leq c$ computable CCEs$\}$ is $(\Sigma^0_3, \Pi^0_3)$ complete.*

2. *$\{e : \Gamma_e$ has exactly $c$ computable CCEs$\}$ is $D^0_3$ complete.*

3. *($\{e : \Gamma_e$ has $< \aleph_0$ computable CCEs$\}, \{e : \Gamma_e$ has $\geq \aleph_0$ computable CCEs$\}$) is $(\Sigma^0_4, \Pi^0_4)$ complete.*

Finally, we consider thin Martin–Pour-El (MPE) theories.

**Theorem 5.8.7.** *$\{e : \Gamma_e$ is $MPE\}$ is a $\Pi^0_4$ complete set.*

*Proof.* Let $f$ be the function from Lemma 5.8.2 so that $P_{f(e)}$ represents the set of complete consistent extensions of $\Gamma_e$. Then $\Gamma_e$ is $MPE$ if and only if $P_{f(e)}$ is thin, and this is a $\Pi^0_4$ condition by Theorem 5.4.6. For the completeness, let $g$ be the function from Lemma 5.8.3 such that $P_e$ represents the class of complete consistent extensions of $\Gamma_{g(e)}$. Then $P_e$ is thin if and only if $\Gamma_e$ is Martin–Pour-El and the $\Pi^0_4$ completeness now follows from Theorem 5.4.6. □

.

**Exercises**

5.8.1. Prove Lemma 5.8.1

5.8.2. Prove Lemma 5.8.3.

5.8.3. Prove Lemma 5.8.2.

5.8.4. Let $f$ be the function from Lemma 5.8.1. Show that $\{e : W_{f(e)} = W_e\}$ is $\Pi^0_2$ complete. That is, the property of being a logical theory is $\Pi^0_2$ complete for sets of sentences.

# Chapter 6

# Reverse Mathematics

There is a close connection between $\Pi^0_1$ classes and certain subsystems of second order arithmetic which are used in the so-called Reverse Mathematics developed by Friedman and Simpson (see [176]). In particular, the system $WKL_0$ (Weak Konig's Lemma) corresponds roughly to the statement that every infinite tree in $\{0,1\}^*$ has an infinite. The system $ACA_0$ arithmetic comprehension) corresponds to the statement that every infinite, finitely branching tree has an infinite path. Thus the representation theorems from Part B may be used to show that certain standard infinite combinatorial theorems are logically equivalent, over the base theory $RCA_0$, to either $WKL_0$ or to $ACA_0$.

For example, consider the completeness theorem for (countable) propositional logic. Given a consistent theory $\Gamma$, the set of complete consistent extensions of $\Gamma$ can be viewed as the infinite paths through a certain infinite binary tree and thus Weak Konig's Lemma can be used to prove that a complete consistent extension exists. On the other hand, given an arbitrary infinite tree $T \subset \{0,1\}^*$, we showed that there exists a consistent theory $\Gamma$ such that $T$ represents the class of complete consistent extensions of $\Gamma$. The completeness theorem tells us that a complete consistent extension exists and therefore $T$ possesses an infinite path. This gives an (informal) proof of Weak Konig's Lemma from the completeness theorem and demonstrates that the two are logically equivalent.

We also present in this chapter the reverse mathematics of propositional logic. Later, in Part B, we will we will consider the proof-theoretic strenth of theorems from various areas of algebra, analysis and combinatorics. This will includee the Cantor-Schroder-Bernstein Theorem and related theorems about symmetric marriages in a highly computable society which are equivalent, variously, to Weak Konig's Lemma or to Arithmetic Comprehension. We also examine several results on infinite partially ordered sets, including Dilworth's theorem that any poset of width $n$ can be covered by $n$ chains.

## 6.1 Subsystems of Second Order Arithmetic

In this section, we discuss the language of second order arithmetic, models of second order arithmetic and the basic axiom system for second order arithmetic as well as certain subsystems closely related to $\Pi^0_1$ classes. These are $RCA_0$, $WKL_0$ and $ACA_0$. For details, see Simpson's [176].

A second order structure includes both objects and sets of objects. Thus a model of second order arithmetic includes a model of first order arithmetic, with a set of objects intended as natural numbers together with the usual operations of addition and subtraction, as well as a collection of *sets* of numbers and the membership relation $(n \in X)$ between the objects and the sets.

The language $\mathcal{L}_2$ of second order arithmetic thus includes the usual language of first order arithmetic, with constant symbols 0 and 1, intended to denote the corresponding natural numbers, with binary function symbols $+$ and $\cdot$, intended to denote the addition and multiplication functions on the natural, and with a relation symbol $<$ intended to denote the ordering of the natural numbers, as well as the usual equality symbol $=$ from predicate logic.

There is also a relation symbol $\in$ which denotes the membership relation. There are two sorts of variables intended to range over numbers and over sets. Number variables $i, j, k, m, n, \dots$ are intended to range over the set $N = \{0, 1, \dots\}$ of natural numbers and set variables $X, Y, Z, \dots$ are intended to range over subsets of $\mathbb{N}$.

Terms are defined as in first order arithmetic to compose the smallest set of strings containing the two constant symbols and all number variables and closed under $t = t_1 + t_2$ and $t = t_1 \cdot t_2$. Atomic formulas are $t_1 = t_2$, $t_1 < t_2$ and $t_1 \in X$, where $t_1$ and $t_2$ are terms and $X$ is a set variable. Formulas compose the smallest set containing all atomic formulas and closed under the propositional connectives, number quantifiers $(\forall n)$ and $\exists n)$ and also under *set quantifiers* $(\forall X)$ and $(\exists X)$.

A model for the language $\mathcal{L}_2$ has the form

$$\mathcal{M} = \langle M, S_M, +_M, \cdot_M, 0_M, 1_M, <_M \rangle,$$

where $M$ is the universe of $\mathcal{M}$, $S_M$ is a set of subsets of $M$, $+_M$ and $\cdot_M$ are binary operations on $M$, $0_M$ and $1_M$ are distinguished elements of $M$ and $<_M$ is a binary relation on $M$.

The intended model for $\mathcal{L}_2$ is $\langle \mathbb{N}, \mathcal{P}(\mathbb{N}), +, \cdot, 0, 1, < \rangle$.

An $\omega$-model $\mathcal{M}$ is a model of $\mathcal{L}_2$ with universe $\mathbb{N}$ and with the standard operations $+$ and $\cdot$, constants 0 and 1, and binary relation $<$, but with $S_M$ merely a *subset* of $\mathcal{P}(\mathbb{N})$. In this case, we simply identify $\mathcal{M}$ with the family $S = S_M$. In addition to the intended model, we will be interested in the following.

**(1)** $REC$ is the $\omega$-model with $S$ the set of *recursive* sets of natural numbers.

**(2)** $ARITH$ is the $\omega$-model with $S$ the set of *arithmetical* sets of natural numbers.

These definitions can be relativized to $REC^B$ and $ARITH^B$ for any fixed $B \subset \mathbb{N}$.

The axioms of second order arithmetic include the eight axioms of Robinson Arithmetic (see Section IV.3.4).

For the induction axiom, we can now discuss *sets* rather than *formulas*, so we have

**IS** $(\forall X)[(0 \in X \ \wedge \ (\forall n)(n \in X \to n+1 \in X)) \to (\forall n) n \in X]$

To ensure that some sets exist, we have a Comprehension Axiom for each formula $\phi(n)$ of $\mathcal{L}_2$:

**C** $(\exists X)(\forall n)[n \in X \iff \phi(n)]$.

Here we allow number and set parameters in the formula $\phi$. These last two axioms imply the induction scheme $IP$ of Peano Arithmetic and in fact a stronger, full second order induction scheme where the formula $\phi$ in $IP$ may be any formula of $\mathcal{L}_2$. Note also that any $\omega$-model also satisfies full second order induction.

These axioms compose the formal system $Z_2$ of second order arithmetic. By a subsystem of second order arithmetic, we mean a theory included in $Z_2$, generally obtained by weakening the axioms of induction and comprehension.

### 6.1.1 Recursive Comprehension

The fundamental system $RCA_0$ consists of the basic axioms of Robinson Arithmetic together with $\Sigma^0_1$ induction and $\Delta^0_1$ comprehension. Some definitions are required to define the hierarchy of formulas.

If $t$ is a numerical term not containing $n$ and $\phi$ is any formula of $\mathcal{L}_2$, then the *bounded quantifiers* $\forall n < t)$ and $(\exists n < t)$ are defined by

$(\forall n < t)\phi \equiv (\forall n)(n < t \to \phi)$ and

$(\exists n < t)\phi \equiv (\exists n)(n < t \ \& \ \phi)$

A formula $\phi$ of $\mathcal{L}_2$ is said to be a *bounded quantifier* (or $\Sigma^0_0$) formula if all of its quantifiers are bounded. $\phi$ is said to be $\Sigma^0_1$ if it is of the form $(\exists m)\psi$ where $\psi$ is a bounded quantifier formula and $\phi$ is said to be $\Pi^0_1$ if it is of the form $(\forall m)\theta$ where $\theta$ is a bounded quantifier formula. More generally, $\phi$ is $\Sigma^0_k$ if it is of the form $(\exists n_1)(\forall n_2)\dots n_k\theta$ where $\theta$ is a bounded quantifier formula, and simlarly for the $\Pi^0_k$ formulas.

**Definition 6.1.1.** *A $\Sigma^0_k$ (respectively $\Pi^0_k$) induction scheme has the form*

$$[\phi(0) \ \& \ (\forall n)(\phi(n) \to \phi(n+1))] \implies (\forall n)\phi(n),$$

*where $\phi$ is any $\Sigma^0_k$ (resp. $\Pi^0_k$) formula of $\mathcal{L}_2$. Here $\phi$ may have other number and set variables.*

**Definition 6.1.2.** 1. *A $\Sigma^0_k$ (respectively $\Pi^0_k$) comprehension scheme has the form*

$$(\exists X)(\forall n)(n \in X \iff \phi(n)),$$

*where $\phi$ is any $\Sigma^0_k$ (resp. $\Pi^0_k$) formula of $\mathcal{L}_2$.*

2. *A* bounded *$\Sigma^0_k$ comprehension scheme has the form $(\forall n)(\exists X)(\forall i)(i \in X \iff (i < n \;\&\; \phi(n))$.*

3. *A $\Delta^0_k$ comprehension scheme has the form*

$$(\forall n)(\phi(n) \iff \psi(n)) \implies (\exists X)(\forall n)(n \in X \iff \phi(n)),$$

*where $\phi$ is a $\Sigma^0_k$ formula and $\psi$ is a $\Pi^0_k$ formula.*
*As above, $\phi$ and $\psi$ may have other number and set variables.*

The $\omega$-models of $RCA_0$ may be characterized as follows.
$S$ is an $\omega$-model of $RCA_0$ if and only if

$S \neq \emptyset$;

$A \in S$ and $B \in S$ imply $A \oplus B \in S$;

$A \in S$ and $B \leq_T A$ imply $B \in S$.

It follows that $RCA_0$ has a minimum $\omega$-model,

$$REC = \{A \in \mathcal{P}(\mathbb{N}) : A \text{ is recursive}\}$$

Simpson [176] outlines the development of ordinary mathematics within $RCA_0$. In particular the coding function $\langle n_1, \dots n_k\rangle$ are definable in $RCA_0$ and Gödel numbering of propositional and also first-order logic may be done there. Functions may be defined by primitive recursion and also by minimization.

Here are some other important results from [176].

**Lemma 6.1.3.** *The following are provable in $RCA_0$.*

1. *For any infinite set $X \subseteq \mathbb{N}$, there exists a strictly increasing function $\pi : \mathbb{N} \to \mathbb{N}$ such that $X$ is the range of $\pi$.*

2. *Let $\phi(n)$ be a $\Sigma^0_1$ formula in which $X$ and $f$ do not occur freely. Then either there exists a finite set $X$ such that $(\forall n)(n \in X \iff \phi(n))$, or there exists a one-to-one function $f : \mathbb{N} \to \mathbb{N}$ such that $(\forall n)(\phi(n) \iff (\exists m)(f(m) = n))$.* □

*Proof.* (1) Let $\pi(0)$ be the least $n \in X$ and for each $k$, let $\pi(k+1)$ be the least $k > \pi(n)$ such that $k \in X$.

(2) Suppose no finite set $X$ exists as stated. Let $\theta$ be $\Sigma^0_0$ such that $\phi(n) \iff (\exists j)\theta(j,n)$ and let $Y = \{\langle j, n\rangle : \theta(j,n) \;\&\; (\forall i < j)\neg\theta(i,n)\}$. Then $Y$ is infinite, so by part (1), there is a function $\pi$ which enumerates $Y$ in increasing order. Let $f(m) = \pi_2(\pi(m))$, where $\pi_2$ is the projection of $\langle x, y\rangle$ onto $y$. □

**Theorem 6.1.4.** *$RCA_0$ proves bounded $\Sigma^0_1$ comprehension.* □

**Corollary 6.1.5.** *$RCA_0$ proves $\Pi^0_1$ induction.*

Here is the basic results for trees.

**Theorem 6.1.6.** *The following is provable in $RCA_0$. If $T \subseteq \{0,1\}^*$ is a tree with no dead ends, then $T$ has an infinite path.*

*Proof.* The leftmost path through $T$ may be defined by primitive recursion. □

On the other hand, König's Lemma is not provable in $RCA_0$, since $REC$ will contain a computable tree with no computable infinite path and therefore no path in $REC$.

### 6.1.2 Weak König's Lemma

In this section we consider the stronger system $WKL_0$ and its relation to $\Pi^0_1$ classes.

**Definition 6.1.7.** 1. *Weak König's Lemma is the statement that every infinite subtree of $\{0,1\}^*$ has an infinite path.*

2. *$WKL_0$ is the subsystem of $Z_2$ consisting of $RCA_0$ plus Weak König's Lemma.*

It is clear that $REC$ is not a model of $WKL_0$ so that $WKL_0$ is a proper extension of $RCA_0$. The formal system $WKL_0$ was first introduced by Friedman [65]. $\omega$-models of $WKL_0$ are sometimes known as *Scott systems* in the literature, referring to [171]. The development of ordinary mathematics in $WKL_0$ is carried out in great detail by Simpson in [176].

The following equivalent forms of Weak König's Lemma are frequently used in the applications.

**Theorem 6.1.8.** *The following are equivalent are equivalent over $RCA_0$*

1. *$WKL_0$, i.e. every infinite tree $T \subset \{0,1\}^{<\mathbb{N}}$ has an infinite path.*

2. *($\Sigma^0_1$ separation) Let $\phi_i(n)$, $i = 0,1$ be $\Sigma^0_1$ formulas in which $X$ is does not occur freely. If $\neg\exists n(\phi_0(n) \wedge \phi_1(n))$, then*

$$\exists X \forall n((\phi_0(n) \rightarrow n \in X) \wedge (\phi_1(n) \rightarrow n \notin X)).$$

3. *If $f, g : \mathbb{N} \rightarrow \mathbb{N}$ are one-to-one with $(\forall m,n) f(m) \neq g(n)$, then there exists a set $X$ such that, for all $m$, $f(m) \in X \ \wedge \ g(m) \notin X$.*

*Proof.* It is clear that (2) implies (1).

(1) $\implies$ (2). Assume (1) and let $T \subset \mathbb{N}^{\mathbb{N}}$ and $g$ be given as stated and define $T^* \subset \{0,1\}^*$ as follows. For any $\tau \in T$ with $|\tau| = n$, let $\tau^* = 0^{\tau(0)}10^{\tau(1)} \ldots 0^{\tau(n-1}$. Then define $T$ by $\Delta^0_1$ Comprehension so that $\sigma \in T^*$ if and only if $\sigma \preceq \tau^*$ for some $\tau \in T$ with $|\tau| \leq g(0) + g(1) + \ldots g(|\sigma|)$.

Then $T^*$ is an infinite subtree of $\{0,1\}^*$ and therefore possesses an infinite path $f^*$. Now define an infinite path $f \in [T]$ by primitive recursion so that $f(0) =$ the least $k < g(0)$ such that $f^*(k) = 0$ and for each $n$, $f(n+1)$ is the least $k < g(n+1)$ such that $f^*(k + g(n)) = 1$. □

**Theorem 6.1.9.** *The following are equivalent are equivalent over $RCA_0$*

1. *$WKL_0$, i.e. every infinite tree $T \subset \{0,1\}^{<\mathbb{N}}$ has an infinite path.*
2. *(Bounded Konig's Lemma) If $T \subset \mathbb{N}^{\mathbb{N}}$ is an infinite tree and there is a function $g$ such that for all $\tau \in T$ and all $m < |\tau|$, $\tau(m) < g(m)$, then $T$ has an infinite path.*

*Proof.* (1) $\implies$ (2). Assume (1) and let $\phi_0, \phi_1$ be given as stated and let $\theta_0, \theta_1$ be bounded quantifier formulas so that $\phi_i(n) \iff (\exists m)\theta_i(m,n)$. Now define $T \subseteq \{0,1\}^*$ by $\Delta^0_1$ Comprehension so that

$$\sigma \in T \iff (\forall i < 2)(\forall m, n < |\sigma|)[\theta_i(m,n) \implies \sigma(n) \neq i].$$

$T$ is an infinite tree and therefore has an infinite path $X$ by Weak König's Lemma, which will satisfy the conclusion of (3).

(2) $\implies$ (1). Let $T \subseteq \{0,1\}^*$ be an infinite tree. Define the $\Sigma^0_1$ formulas $\phi_i$ so that

$$\phi_i(\sigma) \iff (\exists n)(\exists \tau \in \{0,1\}^n)[\sigma^\frown(i)^\frown\tau \in T \,\&\, (\forall \sigma \in \{0,1\}^n)\neg(\sigma^\frown(1-i)^\frown\sigma \in T)].$$

Then $\phi_0, \phi_i$ satsify the hypothesis of (3), so there exists a set $X$ such that for all $\sigma$, $\phi_0(\sigma) \to \sigma \in X$ and $\phi_1(\sigma) \to \sigma \notin X$. We can now define an infinite path through $T$ as follows. Let $\sigma_0 = \emptyset$ and for each $k$, let $\sigma_{k+1} = \sigma_k{}^\frown 0$ if $\sigma_k \in X$ and otherwise $\sigma_{k+1} = \sigma_k{}^\frown 1$. Then $f = \cup_k \sigma_k$ belongs to $[T]$.

(2) $\implies$ (3). Assume (2) and let $f$ and $g$ be given as stated. Let $\phi_0(n) \iff (\exists m) f(m) = n$ and $\phi_1(n) \iff (\exists m) g(m) = n$. The hypothesis of (2) is satisfied by assumption and therefore there exists $X$ as in the conclusion of (3), which will also satisfy the conclusion of (3).

(3) $\implies$ (2). Assume (3) and let $\phi_i$ be given as stated. Apply Lemma 6.1.3 to obtain two cases. First there may exist finite sets $X_i = \{n : \phi_i(n)\}$. If this holds for $i = 0$, let $X = X_0$ and if this holds for $i = 1$, let $X = \mathbb{N} - X_1$. If neither set exists, then there are one-to-one functions $f_i$ such that $\phi_i(n) \iff (\exists m)(f_i(m) = n)$. It follows from the hypothesis of (3) that $(\forall m, n) f(m) \neq g(n)$. Hence by the conclusion of (2), we obtain a set $X$ such that, for all $m$, $f(m) \in X$ and $g(m) \notin X$. This set $X$ then satisfies the conclusion of (3). □

Scott [171] characterized the countable $\omega$-models of $WKL_0$ as those $M \subseteq \mathcal{P}(\mathbb{N})$ such that there exists a complete extension $\Gamma$ of Peano Arithmetic such that $M$ is the family of subsets of $\mathbb{N}$ which are representable in $\Gamma$.

### 6.1.3 Arithmetic Comprehension

In this section we consider the system $ACA_0$ and its relation to $\Pi^0_1$ classes. A formula is said to be *arithmetical* if it is $\Sigma^0_k$ for some $k$.

**Definition 6.1.10.** *1.* **Arithmetical Comprehension** *The arithmetical comprehension scheme is $(\exists X)[n \in X \iff \phi(n)]$ where $\phi$ is an arithmetical formula of $\mathcal{L}_2$ in which $X$ does not occur freely.*

*2. $ACA_0$ is the subsystem of $Z_2$ whose axioms are arithmetical comprehension, full induction and the basic axioms of Robinson arithemtic.*

The $\omega$-models of $ACA_0$ may be characterized as follows.
$S$ is an $\omega$-model of $ACA_0$ if and only if

$S \neq \emptyset$;

$A \in S$ and $B \in S$ imply $A \oplus B \in S$;

$A \in S$ and $B \leq_T A$ imply $B \in S$.

$A \in S$ implies $A' \in S$.

It follows that $ARITH$ is the minimum $\omega$-model for $ACA_0$.

**Theorem 6.1.11.** *The following are equivalent over $RCA_0$.*

1. *$ACA_0$.*
2. *$\Sigma^0_1$ comprehension.*
3. *If $f : \mathbb{N} \to \mathbb{N}$ is an injection, then the range of $f$ is a set.*

*Proof.* The implications (1) $Implies$ (2) and (2) $\implies$ (3) are trivial. The implication (2) $\implies$ (3) follows easily from Lemma 6.1.3. For the implication (1) $\implies$ (2) we prove by induction that $\Sigma^0_k$ comprehension implies $\Sigma^0_{k+1}$ comprehension. Let $\phi(n) \iff (\exists j)\psi(n,j)$ where $\psi$ is $\Pi^0_k$. By $\Sigma^0_k$ comprehension, let $Y = \{(n,j) : \neg\psi(n,j)\}$. Then by $\Sigma^0_1$ comprehension let $X = \{n : (\exists j)(n,j) \notin Y\}$. $\square$

**Theorem 6.1.12.** *The following are equivalent over $RCA_0$.*

1. *$ACA_0$.*
2. *(König's Lemma) If $T$ is an infinite, finitely branching tree, then there is an infinite path through $T$.*
3. *König's Lemma restricted to trees $T$ such that each $\sigma \in T$ has at most two immediate successors in $T$.*

*Proof.* (1) $\implies$ (2). Let $T$ be an infinite, finite-branching tree. By arithmetic comprehension, there is a subtree $T^*$ of $T$ consisting of all $\sigma \in T$ such that $\sigma$ has infinitely many extensions in $T$. Since $T$ is finite branching, every $\sigma \in T^*$ has at least one immediate successor in $T^*$. Clearly $\emptyset \in T^*$ and for each $n$, we may define $g(k)$ to be the least $n$ such that $(g(0), \dots, g(k-1), n) \in T^*$. Then $g \in [T]$ as desired.

(2) $\implies$ (3) is immediate, so it remains to prove (3) $\implies$ (1). Let $f : \mathbb{N} \to \mathbb{N}$ be one-to-one. Use $\Sigma^0_0$ comprehension to define the tree $T$ by $\tau \in T$ if and only if

$$(\forall m, n < |\tau|)[f(m) = n \iff \tau(n) = m + 1]$$

and

$$(\forall n < |t|)[\tau(n) > 0 \implies f(\tau(n) - 1) = n].$$

Each $\sigma \in T$ has at most two possible immediate successors, $\sigma^\frown 0$ and $\sigma^\frown(m+1)$ where $f(m) = |\sigma|$. $T$ is infinite by the following argument. Fix $k$ and define $Y$ by bounded $\Sigma^0_1$ comprehension to be $\{n < k : (\exists m) f(m) = n\}$. Now let $\sigma(n) = 0$ if $n \notin Y$ and $\sigma(n) = m + 1$ if $n \in Y \;\wedge\; f(m) = n$ for $n < k$. Then $|\sigma| = k$ and $\sigma \in T$. Hence by (3), there exists $g \in [T]$. By $\Delta^0_1$ comprehension let $X = \{n : g(n) > 0\}$. Then $X$ is the range of $f$ as desired. $\square$

## 6.2 Mathematical Logic

In this section, we consider the connection between logical theories, infinite trees and subsystems of second order arithmetic.

A weak form of the completeness theorem can be proved even for first-order logic. Here is the propositional version.

**Theorem 6.2.1.** *[[176]] The following is provable in $RCA_0$. If Gamma is a consistent propositional theory, then there exists a countable model $M$ for $X$ such that $M \vdash \phi$ for all $\phi \in \Gamma$.*

*Proof.* Recall from Theorem III.2.9.3 that, for each finite sequence $\sigma = (\sigma(0), \dots, \sigma(n-1))$, we defined $P_\sigma = C_0 \wedge C_1 \wedge \cdots \wedge C_{n-1}$, where $C_i = A_i$ if $\sigma(i) = 1$ and $C_i = \neg A_i$ if $\sigma(i) = 0$. Given the theory $\Gamma$, define the tree $T$ without dead ends by

$$\sigma \in T \iff \neg P_\sigma \notin \Gamma\}.$$

Since $\Gamma$ is a theory, $\sigma \in T$ if and only if $P_\sigma$ is consistent with $\Gamma$. $T$ has no dead ends since $\Gamma$ is consistent. That is, if $|\sigma| = n$ and $P_\sigma$ is consistent with $\Gamma$, then either $P_\sigma \wedge A_i$ is consistent with $\Gamma$ or $P_\sigma \wedge \neg A_i$ is consistent with $\Gamma$. It follows from Theorem 6.1.6 that $T$ has an infinite path $X$ and the model $M$ is defined by letting $M \vdash A_i$ if $X(i) = 1$ and $M \vdash \neg A_i$ if $X(i) = 0$. $\square$

Using Weak König's Lemma, we can prove the completeness theorem and the reverse is also true. (See [176].) We note first that the representation theorem III.2.9.1 for propositional logic can be proved in $RCA_0$, that is, for any countable set $\Gamma$ of sentences, there exists a tree $T \subseteq \{0,1\}^*$ such that $[T]$ represents the set of complete consistent extensions of $\Gamma$. Likewise the reverse representation theorem III.2.9.3 can be proved in $RCA_0$. That is, given the tree $T$, define the set $\Gamma(T)$ to consist of all $P_\sigma \to A_n$ such that $\sigma \in T$ and $\sigma^\frown 0 \notin T$ and all $P_\sigma \to \neg A_n$ such that $\sigma \in T$ and $\sigma^\frown 1 \notin T$, where $|\sigma| = n$. Then there is a one-to-one correspondce between the complete consistent extensions of $\Gamma(T)$ and the infinite paths through $T$.

**Theorem 6.2.2.** *The following are equivalent are equivalent over $RCA_0$*

1. *$WKL_0$, i.e. every infinite tree $T \subset \{0,1\}^{<\mathbb{N}}$ has an infinite path.*
2. *Lindenbaum's Lemma: every countable consistent set of sentences has a complete consistent extension.*
3. *The completeness theorem for propositional logic with countably many variables.*
4. *The compactness theorem for propositional logic with countable many variables.*

*Proof.* (1) $\implies$ (2) follows from the representation theorem as discussed above. That is, given a consistent set $\Gamma$, we can build in $RCA_0$ an infinite tree representing the set of complete consistent extensions of $\Gamma$ and then use Weak König's Lemma to find an infinite path $X$ through $T$ and hence a complete consistent extension of $\Gamma$.

(2) $\implies$ (3). For propositional logic, this is immediate. Just let $\Delta$ be a complete consistent extension of $\Gamma$ and let $M(A_i) = 1$ if and only if $A_i \in \Delta$.

(3) $\implies$ (4). Suppose that every finite subset of $\Gamma$ is satisfiable. Then $\Gamma$ is consistent and hence has a model $M$ by (3) and is therefore satisfiable.

(4) $\implies$ (1). Assume (4) and let $T \subseteq \{0,1\}^*$ be an infinite tree. Let $\Gamma(T)$ be constructed as above. Then $\Gamma(T)$ is finitely satisfiable and hence has a model (and therefore a complete consistent extension) by (4). But this implies that $T$ has an infinite path. □

**Exercises**

6.2.1. Show that a set of natural numbers is c. e. if and only if it is definable by a $\Sigma^0_1$ formula over the standard model of arithmetic and therefore is computable if and only if it is $\Delta^0_1$ definable.

6.2.2. Show that $S \subseteq \mathcal{P}(\mathbb{N})$ is a model of $RCA_0$ if and only if

**(i)** $S \neq \emptyset$;

$A \in S$ and $B \in S$ imply $A \oplus B \in S$;

$A \in S$ and $B \leq_T A$ imply $B \in S$.

# Chapter 7

# Complexity Theory

In this chapter, we examine the notions of computable trees and effectively closed sets in a *resource-bounded* setting. We consider the complexity of the members of effectively closed sets as in Chapter 3 from this point of view. We show that for any $\Pi^0_1$ class $P \subset \mathbb{N}^{\mathbb{N}}$, there is a polynomial time tree $T$ such that $P = [T]$. Resource-bounded variations on the notions of boundedness for trees and classes are defined, such as *locally p-time*, highly p-time, and p-time bounded are defined and basis and antibasis results given. For example, every locally p-time tree possesses an infinite path which is computable in double exponential time and also, there is a p-time bounded $\Pi^0_1$ class with a unique element which is not p-time computable.

We also look at the representation problem and show that for essentially all of the problems from Part 2, polynomial time presented problems suffice to represent all $\Pi^0_1$ classes. This is based on the result that any computable relational structure is computably isomorphic to a polynomial time structure.

Let $\Sigma$ be a (usually finite) alphabet. Then $\Sigma^*$ denotes the set of finite strings of letters from $\Sigma$ and $\Sigma^\omega$ denotes the set of infinite sequences. In particular, each natural number $n$ may be represented in unary form by the string $tal(n) = 1^n$ if $n > 0$ and $tal(n) = 0$ if $n = 0$ and in (reverse) binary form by the string $bin(n) = i_0 \cdots i_k$, where $n = i_0 + i_1 \cdot 2 + \cdots + i_k \cdot 2^k$.

We let $Tal(\omega) = \{tal(n) : n \in \omega\}$ and $Bin(\omega) = \{bin(n) : n \in \omega\}$. Both sets are included in $\{0,1\}^*$. The tally and binary representation of the natural numbers will be essential for our study of feasible structures, problems and solutions. The main reason is due to the fact the feasibility of an algorithm is usually measured in terms of the computation time as a function of the length of the input to the algorithm. Note that since the tally representation of a number is of exponential length in comparison to the binary representation, it follows that a function which is polynomial time computable in the tally representation of the natural numbers is not necessarily polynomial time computable in the binary representation of the natural numbers. Indeed, we can only conclude that such a function is exponential time computable in the binary representation.

Thus it is essential that a definite representation be given for a feasible structure. A related reason is that two feasible sets need not be feasibly isomorphic. In particular, $Tal(\omega)$ and $Bin(\omega)$ are not p-time isomorphic. Thus we may have a p-time structure, say a graph, with universe $Tal(\omega)$, which is not isomorphic to a p-time structure with universe $Bin(\omega)$.

Our basic computation model is the standard multitape Turing machine of Hopcroft and Ullman [82]; see also Papadimitriou [154]. Note that there are different heads on each tape and that the heads are allowed to move independently. This implies that a string $\sigma$ can be copied in linear time. An oracle machine is a multitape Turing machine $M$ with a distinguished work tape, a query tape, and three distinguished states QUERY, YES, and NO. At some step of a computation on an input string $\sigma$, $M$ may transfer into the state QUERY. In state QUERY, $M$ transfers into the state YES if the string currently appearing on the query tape is in an oracle set $A$. Otherwise, $M$ transfers into the state NO. In either case, the query tape is instantly erased. The set of strings accepted by $M$ relative to the oracle set $A$ is $L(M,A) = \{\sigma|$ there is an accepting computation of $M$ on input $\sigma$ when the oracle set is $A\}$. If $A = \emptyset$, we write $L(M)$ instead of $L(M,\emptyset)$.

Let $t(n)$ be a function on natural numbers. A Turing machine $M$ is said to be $t(n)$ *time bounded* if each computation of $M$ on inputs of length $n$ where $n \geq 2$ requires at most $t(n)$ steps. A function $f(x)$ on strings is said to be in $DTIME(t)$ if there is a $t(n)$-time bounded deterministic Turing machine $M$ which computes $f(x)$. For a function $f$ of several variables, we let the length of $(x_1, \dots, x_n)$ be $|x_1| + \cdots + |x_n|$. A set of strings or a relation on strings is in $DTIME(t)$ if its characteristic function is in $DTIME(t)$. A Turing machine $M$ is said to be $t(n)$ *space bounded* if each computation of $M$ on inputs of length $n$ where $n \geq 2$ the work space required to carry out the computation is bounded by $t(n)$ . A function $f(x)$ on strings is said to be in $DSPACE(t)$ if there is a $t(n)$-space bounded deterministic Turing machine $M$ which computes $f(x)$. For a function $f$ of several variables, we let the length of $(x_1, \dots, x_n)$ be $|x_1| + \cdots + |x_n|$. A set of strings or a relation on strings is in $DSPACE(t)$ if its characteristic function is in $DSPACE(t)$.

We let

$LOGTIME = \bigcup_{c \geq 1} DTIME(c \cdot log_2(n))$,

$LOG = \bigcup_{c \geq 1} DSPACE(c \cdot log_2(n))$,

$LIN = \bigcup_{c > 0} DTIME(cn)$,

$P = \bigcup_{i \in \omega} DTIME(n^i)$,

$PSPACE = \bigcup_{i \in \omega} DSPACE(n^i)$,

$DEXT = \bigcup_{c \geq 0} DTIME(2^{c \cdot n})$,

$EXPSPACE = \bigcup_{c\geq 0} DSPACE(2^{c\cdot n})$,

$DOUBEXT = \bigcup_{c\geq 0} DTIME(2^{2^{c\cdot n}})$,

$DOUBEXPSPACE = \bigcup_{c\geq 0} DSPACE(2^{2^{c\cdot n}})$,

$EXPTIME = \bigcup_{c\geq 0} DTIME(2^{n^c})$,

$DOUBEXPTIME = \bigcup_{c\geq 0} DTIME(2^{2^{n^c}})$, and in general,

$DEX(S) = \bigcup_{t(n)\in S} DTIME(2^{t(n)})\}$.

We say that a function $f(x)$ is *polynomial time* if $f(x) \in P$, is *exponential time* if $f(x) \in DEXT$, and is *double exponential time* if $f(x) \in DDOUBEXT$.

A function $f(x)$ on strings is said to be in $NTIME(t)$ if there is a $t(n)$-time bounded nondeterministic Turing machine $M$ which computes $f(x)$. A set of strings or a relation on strings is in $NTIME(t)$ if its characteristic function is in $NTIME(t)$. We let

$NLOG = \bigcup_{c\geq 1} NSPACE(C \cdot log_2(n))$,

$NP = \bigcup_{i\in\omega} NTIME(n^i)$,

$NEXT = \bigcup_{c\geq 0}\{NTIME(2^{c\cdot n})\}$,

$NEXPTIME = \bigcup_{c\geq 0} NTIME(2^{n^c})$,

A function $f$ is said to be *non-deterministic polynomial time (NP)* if there is a finite alphabet $\Sigma$, a polynomial $p$, a p-time relation $R$ and a p-time function $g$ such that, for any $\sigma$ and $\tau$,

$$f(\sigma) = \tau \iff (\exists \rho \in \Sigma^{p(\sigma)})[R(\rho,\sigma) \ \&\ g(\rho,\sigma) = \tau].$$

Similar definitions apply to other complexity classes.

We fix enumerations $\{P_i\}_{i\in N}$ and $\{N_i\}_{i\in N}$ of the polynomial time bounded deterministic oracle Turing machines and the polynomial time bounded non-deterministic oracle Turing machines respectively. We may assume that $p_i(n) = max(2,n)^i$ is a strict upper bound on the length of any computation by $P_i$ or $N_i$ with any oracle $X$ on inputs of length $n$. $P_i^X$ and $N_i^X$ denote the oracle Turing machine using oracle $X$.

For $A,\ B \subset \Sigma^*$, we shall write $A \leq_m^P B$ if there is a polynomial-time function $f$ such that for all $x \in \Sigma^*$, $x \in A$ iff $f(x) \in B$. We shall write $A \leq_T^P B$ if $A$ is polynomial time Turing reducible to $B$. For $r$ equal to $m$ or $T$, we write $A \equiv_r^P B$ if $A \leq_r^P B$ and $B \leq_r^P A$ and we write $A \mid_r^P B$ if not $A \leq_r^P B$ and not $B \leq_r^P A$.

## 7.1 Complexity of Trees

We think of a computable tree $T$ as a set of finite sequences $(n_0, \dots, n_{k-1})$ of natural numbers and of an infinite path $(n_0, n_1, \dots)$ through $T$ as a function from the natural numbers into the natural numbers which maps $i$ to $n_i$. We shall define two natural representations of $T$ which will be useful for the study of the complexity of trees and paths through trees. First we define the *binary representation* of $T$, $bin(T)$, as the set of finite strings $\{(bin(n_0), \dots, bin(n_{k-1})) : (n_0, \dots, n_{k-1}) \in T\}$. We also define the *tally representation* of $T$, $tal(T)$, to be the set of strings $\{(tal(n_0), \dots, tal(n_{k-1}) : (n_0, \dots, n_{k-1}) \in T\}$. The strings in $bin(T)$ and $tal(T)$ are over the finite alphabet $\{0, 1, ',', '(', ')'\}$ which has symbols for the comma and the left and right parentheses. We say that $T$ is *p-time in binary* if $bin(T)$ is a polynomial time subset of $\Sigma^*$. Similarly we say $T$ is *p-time in tally* if $tal(T)$ is p-time subset of $\Sigma^*$. Since $bin(n)$ can be computed in polynomial time from $tal(n)$, it follows that if $bin(T)$ is p-time, then $tal(T)$ is also p-time. Given an infinite path $x = (n_0, n_1, \dots)$ through $T$, the binary representation of $x$ is the function $bin(x)$ from $Tal(\omega)$ to $Bin(\omega)$ defined by $bin(x)(tal(i)) = bin(n_i)$. The tally representation of $x$, $tal(x)$, is similarly defined by $tal(x)(tal(i)) = tal(n_i)$. Then we say that $x$ is a *polynomial time path in binary* if the function $bin(x)$ is the restriction of p-time function from $\{0,1\}^*$ to $\{0,1\}^*$; we say that $x$ is *p-time in tally* if $tal(x)$ is the restriction of a p-time function from $\{0,1\}^*$ to $\{0,1\}^*$. It is clear that if $x$ is p-time in tally, then $x$ is also p-time in binary, since $bin(x)(tal(i))$ can be computed from $tal(x)(tal(i))$ for each $i$. The reason for using $Tal(\omega)$ for the domain of $bin(x)$ is the following. For any path $x$, $x$ is computable if and only if the initial segment function $\overline{x}$ is computable, where $\overline{x}(i) = (n_0, \dots, n_{i-1})$. We want to have a similar result for p-time paths, and this would be impossible if $\overline{x}$ had to map $bin(i)$, which has length roughly $log_2(i)$, to a string which must have length at least $i$. Similar definitions can be given for other notions of complexity, such as exponential time, non-deterministic polynomial time (NP), etc.

Recall that a tree $T \subset \omega^{<\omega}$ is highly computable if there is a recursive function $f$ such that, for any node $\sigma \in T$, $f(\sigma)$ is the number of immediate successors $\sigma^\frown i$ of $\sigma$ in $T$. Now given the number of successors of a node, we can search through all the possible immediate successors and find the largest one. Thus we can find a computable function $g$ such that $g(\sigma)$ is the largest $i$ such that if $\sigma = (\sigma_0, \dots, \sigma_n) \in T$, then $(\sigma_0, \dots, \sigma_n, i)$ is in $T$. Finally, we can also compute recursively the sequence $h(\sigma) = (i_1, \dots, i_d)$ which lists all $i$ such that $(\sigma_0, \dots, \sigma_n, i)$ is in $T$ in increasing order. It is clear that $f$ is computable if and only if $g$ is computable and if and only if $h$ is computable. The situation is different for polynomial time complexity. Consider first the binary representation of $T$ so that we identify a node $\sigma \in T$ with a sequence of numbers in $Bin(\omega)$. It is not hard to see that if $h$ is p-time, then both $f$ and $g$ are p-time. However, these are the only relations which are guaranteed to hold between the three functions. To see this, consider the following three examples.

**Example 7.1.1.** *Define the sequence $x_0, x_1, \dots$ of natural numbers by letting*

*$x_0 = 1$ and, for each $n$, $x_{n+1} = 2^{x_n}$ and let $T = \{(x_0, \ldots, x_{i-1}) : i \in \omega\}$. Then the tree $T$ is p-time and $f$ is p-time, since $f(\sigma) = 1$ for all $\sigma \in T$. However, the function $g$ cannot be p-time since, for if $\sigma = (x_0, \ldots, x_n)$, then in the binary representation $|\sigma| \leq 3|x_n|$ whereas $|x_{n+1}| = 2^{|x_n|}$.*

**Example 7.1.2.** *Define the tree $T_1$ computably by putting $\emptyset \in T_1$ and, for any $\sigma = (x_0, \ldots, x_{n-1}) \in T_1$, putting $\sigma^\frown i \in T_1$ if and only if $i \leq 1 + x_0 + \cdots + x_{n-1}$. $T_1$ is clearly p- time and the function $g$ is also p-time since $g((x_0, \ldots, x_{n-1})) = 1 + x_0 + \cdots + x_{n-1}$. However, the function $h$ which lists the immediate successors of any node is not p-time because, for any $n$, if $\sigma = (1, 2, 4, \ldots, 2^n)$, then $h(\sigma) = (0, 1, \ldots, 2^{n+1})$, so that in the binary representation $|h(\sigma)| > 2^{n+1}$, whereas $|\sigma| = (n+2)(n+3)/2$.*

**Example 7.1.3.** *For this example, we will appeal to the intractability of the well-known $P = NP$ conjecture. That is, we will define a p-time tree $T_2$ for which the function $g$ is p-time and such that if the associated function $f$ were p-time, then the $P = NP$ conjecture would be true. The tree $T_2$ will be defined so that $\sigma = (n_0, n_1, \ldots, n_{2k+1}) \in T_2$ if and only if, for each $i \leq k$, $bin(n_{2i})$ codes a graph on $i$ vertices and $bin(n_{2i+1})$ either codes a Hamiltonian path on the graph coded by $bin(n_{2i})$ or is a string of 1's of the appropriate length. Now a graph $G_i$ on $i$ vertices $v_1, \ldots, v_i$ is determined by a set of unordered pairs $(v_r, v_s)$ of vertices (the edges of the graph). There are $\binom{i}{2} = i(i-1)/2$ possible edges in $G_i$ and these may be lexicographically ordered so that a sequence $e_1, e_2, \ldots, e_{\binom{i}{2}}$ codes the graph $G_i$ where, for all $t \leq \binom{i}{2}$, $e_t = 1$ if $G_i$ has the $t$'th edge and $e_t = 0$ otherwise. Of course the (reverse) binary representation $bin(n_i)$ must end with a 1, so the graph $G_i$ will actually be coded by the string $(e_0, \ldots, e_{\binom{i}{2}}, 1)$. Observe that this code for $G_i$ will always be a string of length $1 + \binom{i}{2}$ and that any binary number $bin(n)$ of length $1 + \binom{i}{2}$ will code a graph on $i$ vertices. Now a Hamiltonian path on $G_i$ is a permutation $(v_{r_0}, v_{r_1}, \ldots, v_{r_i})$ of the vertices such that there is an edge joining $v_{r_t}$ with $v_{r_{t+1}}$ for all $t < i$. Such a path will be coded by the binary sequence $0^{r_0}10^{r_1}1 \ldots 0^{r_i}1$, which will always be a binary number of length $\binom{i+1}{2}$ and binary number $bin(n)$ of length $\binom{i+1}{2}$ will code a possible Hamiltonian path on a graph of $i$ vertices if and only if $bin(n)$ has exactly $i$ 1's. It is easy to see that there is a p-time algorithm which will decide, given two binary numbers $bin(n)$ and $bin(m)$, whether $bin(n)$ has length $\binom{i}{2} + 1$ for some $i < |bin(n)|$ and therefore codes a graph $G$ on $i$ vertices and whether $bin(m)$ codes a Hamiltonian path on that graph. The tree $T_2$ can now be defined by putting $\sigma = (bin(n_0), \ldots, bin(n_{2k+1}) \in T_2$ if and only if, for each $i < k$, $bin(n_{2i})$ codes a graph $G_i$ on $i$ vertices and $bin(n_{2i+1})$ either codes a Hamiltonian path on $G_i$ or equals $tal(\binom{i+1}{2})$. It follows from the discussion above that $T_2$ is a p-time tree. Now the function $g$ for this tree is p-time since, for any $\sigma = (\sigma_0, \ldots, \sigma_t) \in T_2$, we have $g(\sigma) = 2^{\binom{i+1}{2}} - 1$ if $t = 2i$ and $g(\sigma) = 2^{1+\binom{i}{2}} - 1$ if $t = 2i - 1$. (In each case, $g(\sigma)$ is just a string of 1s of the right length.) On the other hand, the function $f$ associated with the tree $T_2$ has the property that for any $\sigma = (\sigma_0, \ldots, \sigma_{2i}) \in T_2$, $f(\sigma) = 1$ if and only if the graph $G_i$ coded by $\sigma_{2i}$ has no Hamiltonian path. Now suppose that $f$ were p-time and let $bin(n)$ be a code* *for a finite graph on $k$ vertices. For all $i < k$, let $n_{2i} = 2^{\binom{i+1}{2}} - 1$ and let $n_{2i+1} = 2^{1+\binom{i+1}{2}} - 1$. Finally, let $\sigma = (bin(n_0), \ldots, bin(n_{2k-1}), bin(n))$. Then the sequence $\sigma$ can be computed from $bin(n_0)$ in polynomial time and $G$ has a Hamiltonian path if and only if $f(\sigma) > 1$. It follows that if $f$ were p-time, then the Hamiltonian path problem would be p-time. But it is well-known that the Hamiltonian path problem is NP-complete. (See Garey and Johnson [6] for an explanation of NP-completeness and the $P = NP$ problem.) Thus we have demonstrated that if the function $f$ associated with the tree $T_2$ were p-time, then $P = NP$ would true.*

Now the situation is slightly different for tally representation of $T$ where we identify a $\sigma \in T$ with a sequence of numbers in $Tal(\omega)$. Once again it is easy to see that if $h$ is p-time, then $f$ and $g$ are p-time. Moreover, example (1) above will still show that $f$ may be p-time without $g$ being p-time. However in this case, if $T$ is p-time in tally and $g$ is p-time, then $h$ is also p-time. To see this, suppose $\sigma = (\sigma_0, \ldots, \sigma_n)$. Note that to find $h(\sigma)$, we need only check whether $(\sigma_0, \ldots, \sigma_n, i) \in T$ for $i \leq g(\sigma)$. Now in the tally representation, $|(\sigma_0, \ldots, \sigma_n, 0)| < |(\sigma_0, \ldots, \sigma_n, 1)| < \ldots < |(\sigma_0, \ldots, \sigma_n, g(\sigma))|$. Then if it takes $q(|\sigma|)$ steps to check whether $\sigma \in tal(T)$ for each $\sigma = (tal(\sigma_0), \ldots, tal(\sigma_n))$, than it will take approximately

$$\sum_{i=0}^{|(\sigma_0,\ldots,\sigma_n,g(\sigma))|} q(i) \leq q(|(\sigma_0, \ldots, \sigma_n, g(\sigma))|)^2$$

steps to check whether $(\sigma_0, \ldots, \sigma_n, i) \in T$ for $i \leq g(\sigma)$. Thus it is easy to see that we can find $h(\sigma)$ in polynomial time in the tally representation of $\sigma$.

We say that a tree $T$ is *locally p-time* in binary (respectively in tally) if all three of the functions defined above are p-time in binary (resp. tally). In the case that $T$ is not itself p-time, then we will say that $T$ is locally p-time if each of the functions is the restriction to $T$ of a function which is p-time (in binary or tally).

Next we will show that if $T$ is locally p-time, then $T$ is also p-time. The same argument works for both binary and tally. Let $Q$ be either $bin(T)$ or $tal(T)$ and suppose that the function $h$ associated with $Q$ is p-time. Given a sequence $\sigma = (\sigma_0, \ldots, \sigma_k)$, here is the procedure for testing whether $\sigma \in tal(T)$. Begin by computing $h(\emptyset) = (tau_1, \ldots, tau_d)$ and checking whether $\sigma_0 = \tau_i$ for some $i \leq d$. Then, for $j < k$ in turn, compute $h(\sigma_0, \ldots, \sigma_j)$ and check to see that $\sigma_{j+1}$ is in this list. Suppose that $h(\tau)$ may be calculated in time $p(|\tau|)$, where $p$ is some polynomial, then since each $(\sigma_0, \ldots, \sigma_j)$ is a substring of $\sigma$, we see that we can do each of the $h$ computations in time no greater than $p(|\sigma|)$. To read the resulting list of possible successors of $(\sigma_0, \ldots, \sigma_j)$ and compare each one with $\sigma_{j+1}$ can then be done in time at most $(c-1)p(|\sigma|)$ for some fixed constant $c$. Thus each step of the procedure takes time at most $cp(|\sigma|)$. Now there are $k$ such steps and $k \leq |\sigma|$, so that the entire procedure takes time at most $c|\sigma|p(|\sigma|)$, which is again a polynomial function of $|\sigma|$.

The functions $f$, $g$ and $h$ describe the behavior of the tree at a particular node. Now sometimes we need to have a global bound as well. Note that for a computably bounded tree, there is a computable function $p$ such that, for all natural numbers k and all $\sigma = (n_0, \dots, n_k) \in T$, $n_k \leq p(k)$. We will say that a tree $T$ is *p-time bounded in binary* if there is a p-time function $p$ such that, for all natural numbers $k$ and all $\sigma = (n_0, \dots, n_k) \in T$, $|bin(n_k)| \leq p(1^k)$. A tree $T$ is *p-time bounded in tally* if there is a p-time function $p$ such that for any $\sigma = (n_0, \dots, n_k) \in T$, we always have $n_k = |tal(n_k)| \leq p(1^k)$. Since we can compute $bin(n)$ from $tal(n)$ in polynomial time, it follows that any tree which is p-time bounded in tally is also p-time bounded in binary. Note that any tree $T \subset \{0,1\}^{<\omega}$ is p-time bounded, so that a tree may be p-time bounded without being p-time. One additional observation is worth making at this point. If $T$ is p-time bounded in tally, then there will also be a p-time function $q$ such that, for any $\tau = (n_0, \dots, n_k) \in T$, $|tal(\tau)| \leq q(1^k)$. To see this, note that $\tau$ consists of the strings $tal(n_i)$ for $i \leq k$, separated by commas and with parentheses at the beginning and end. Thus

$$|\tau| = 2 + k + |n_0| + \cdots + |n_k| \leq 2 + k + p(1^0) + \cdots + p(1^k).$$

Thus we can define a p-time bound $q(1^k) = 2 + k + p(1^0) + \cdots + p(1^k)$, which is clearly p-time computable. The same observation holds for p-time bounded in binary.

Now suppose that $T$ is p-time bounded in tally and that $Tal(T)$ is p-time. This implies that there are at most $p(1^k)$ possible choices for $tal(n_k)$, that is, the strings $1^e$ for $e < p(1^k)$. To compute $h(\sigma)$, where $\sigma = (tal(n_0), \dots, tal(n_{k-1}))$, we simply use the p-time algorithm for membership in $tal(T)$ to test whether $\sigma * tal(i) \in tal(T)$ for all $i \leq p(1^k)$ and compile the list $(tal(i_1), \dots, tal(i_d)) = h(\sigma)$ of all $tal(i)$ such that $\sigma * tal(i) \in T$. This shows that the function $h$ is p-time in tally. It then follows by the discussion above that $g$ and $f$ are also p-time in tally. Hence in the tally representation, any p-time bounded, p-time tree is also locally p-time.

Let us say that a tree $T$ is *highly p-time in binary* if $T$ is p-time, locally p-time and also p-time bounded in binary. Similarly, $T$ is highly p-time in tally if $T$ is p-time, locally p-time and also p-time bounded in tally. Then we have shown that in tally, p-time plus p-time bounded implies highly p-time. On the other hand, we have also seen that these notions are distinct for the binary representation.

Similar definitions can be given for other notions of complexity. Our next theorem shows that any $\Pi^0_1$-class can be realized as the set of infinite paths through a p-time tree.

**Theorem 7.1.4.** *Let $T$ be a computable tree. Then there is a polynomial time tree $P$ such that $[T] = [P]$. Furthermore, if $T$ is computably bounded, then $P$ is also computably bounded and if $T$ is p-time bounded, then $P$ is also p-time bounded.*

*Proof.* The same argument works for the binary and for the tally representation. We will give the binary argument for the first part and the tally argument for the second part, since these are the stronger results. Let $\phi$ be a computable function from $\omega^{<\omega}$ into $\{0,1\}$ such that $\sigma \in bin(T) \iff \phi(\sigma) = 1$. Let $\phi^s$ denote the partial computable function which results by computing $\phi$ for exactly $s$ steps on any input and let $T^s$ be the s'th approximation to $T$, given by

$$\sigma \in T^s \iff \phi^s(bin(\sigma)) = 1 \text{ or is undefined .}$$

Thus $T^0 \supset T^1 \supset \cdots$ and, for any $\sigma$, $\sigma \in T \iff (\forall s)(\sigma \in T^s)$.

Now define the p-time tree P by letting

$$\sigma \in P \iff (\forall \tau \prec \sigma)\tau \in T^{|bin(\sigma)|}.$$

Note that $P$ is a p-time tree in binary since to compute whether $\tau \in T^{|bin(\sigma)|}$ requires $|bin(\sigma)|$ steps for all $\tau$ so that to compute whether $\sigma \in P$ requires roughly
$|bin(\sigma)|(|bin(\sigma)| + 1)$ steps.

It follows from the definition of $P$ that $T \subset P$, so that $[T] \subset [P]$. Now suppose that $x \notin [T]$. Then there is some initial segment $\tau = x \restriction n$ which is not in T. This means that, for some s, $\tau \notin T^s$. Since the sequence $T^s$ is decreasing, we may assume that $s > n$. Now let $\sigma = x \restriction s$, so that $|bin(\sigma)| \geq s$. It follows from the definition of P that $\sigma \notin P$. This implies that $x \notin [P]$. Thus $[T] = [P]$.

Now suppose that $T$ is computably bounded in tally and let $p$ be the computable function which computes, for each $k$, an upper bound $p(1^k)$ (in tally) for the possible value of $n_k$ for any node $\sigma = (n_0, \ldots, n_k) \in T$.

Suppose first that $p$ is actually p-time. Then we can recursively define a tree $Q$ such that $T \subset Q \subset P$ by putting $\sigma = (n_0, \ldots, n_k) \in Q$ if and only if $\sigma \in P$ and, for all $i \leq k$, $n_i \leq p(1^i)$. It is clear that $[Q] = [T]$ and that $Q$ is p-time since $P$ and $p$ are p-time.

Finally, suppose only that $p$ is recursive and let $p^s$ be the usual result of computing $p$ for $s$ steps. Once again we can define a highly recursive tree $Q$ such that $T \subset Q \subset P$ by putting $\sigma = (n_0, \ldots, n_k) \in Q$ if and only if $\sigma \in P$ and, for all $i \leq k$, either $p^k(1^i)$ is undefined or $n_i \leq p^k(1^i)$. Then again it is easy to check that $Q$ is p-time in binary and that $[Q] = [T]$. □

Next we would like to consider conditions which might force the tree $T$ to have a p-time (exponential time, etc.) path. Recall that a $\Pi^0_1$ class $P$ is decidable if $P = [T]$ for a computable tree with no dead ends (or with $Ext(T)$ computable) and that a decidable $\Pi^0_1$ class always has a computable member. Recall also that any $\Pi^0_1$ singleton is necessarily computable. Next we show that the obvious p-time analogues of these results fail for p-time decidable trees.

**Theorem 7.1.5.** *For any computable $x \in \{0,1\}^\omega$, there is a tree $T$ which is polynomial time in binary and in tally and such that $[T] = \{x\}$.*

*Proof.* This follows from Theorem 7.1.4, since for $x \in \{0,1\}^\omega$, the tree $T = \{(x(0), \ldots, x(n-1) : n < \omega\}$ is computable □

Theorem 7.1.5 shows that even if a polynomial time bounded p-time tree has a unique infinite path $\Pi$, $\Pi$ may not be polynomial time. However there are some natural conditions which we can put on $T$ which will ensure that in such situations we can at least get double exponential time paths or winning strategies and in some cases actually guarantee the existence of polynomial time paths or winning strategies.

**Theorem 7.1.6.** *(a) Let $Ext(T)$ be a locally p-time tree in tally (respectively binary) and let $[T]$ be nonempty. Then $[T]$ contains an infinite path which is double exponential time computable in tally (resp. binary). Furthermore, if $Ext(T)$ is locally p-time in tally (resp. binary) and $[T]$ is finite, then every element of $[T]$ is computable in double exponential time in tally (resp. binary).*

*(b) Let $Ext(T)$ be a locally p-time tree in tally (respectively binary) and let $[T]$ be nonempty. Moreover, assume that there is a linear time function $h$ such that for all $\sigma = (n_0, \ldots, n_k) \in T$, $h(b(\sigma))$ lists all $b(n)$ such that $(n_0, \ldots, n_k, n) \in T$ where $b() = tal()$ if $T$ is p-time in tally and $b() = bin()$ if $T$ is p-time in binary. Then $[T]$ contains an infinite path which is exponential time computable in tally (resp. binary). Furthermore, if $[T]$ is finite, then every element of $[T]$ is computable in exponential time in tally (resp. binary)*

*(c) If $Ext(T)$ is a highly p-time tree in tally (resp. binary) and $[T]$ is nonempty, then $[T]$ contains an infinite path which is p-time time in tally (resp. binary). Furthermore, if $[T]$ is finite, then every element of $[T]$ is p-time in tally (resp. binary).*

*(d) If $Ext(T)$ is a p-time bounded, p-time tree in binary and $[T]$ is nonempty, then $[T]$ contains an infinite path which is $EXPTIME$ in binary. Furthermore, if $[T]$ is finite, then every element of $[T]$ is NP in binary.*

*Proof.* To simplify the discussion, we will assume in all cases that $T = Ext(T)$, that is, that $T$ has no dead ends. Thus the conditions set out for $Ext(T)$ will become the conditions for $T$.

(a) We give the proof for the binary representation of $T$. The proof for the tally representation is exactly the same except for replacing $bin(...)$ with $tal(...)$ at appropriate locations throughout. Let $h$ be the p-time function such that for all $\sigma = (n_0, \ldots, n_k) \in T$, $h(bin(\sigma))$ lists all $bin(n)$ such that $(n_0, \ldots, n_k, n) \in T$. Then we can recursively define the p-time path $x$ through $T$ by letting $x(k)$ be the number $n$ such that $bin(n)$ is the first entry of $h(bin(x \upharpoonright k))$. It remains to be checked that the computation of $bin(x(n))$ from $1^n$ can be done in double exponential time. Let $c$ be a number such that $h(\tau)$ can be computed from $\tau$ in time bounded by $|\tau|^{c-1}$ for all $\tau \in Bin(Ext(T))$ with $|\tau| \geq 2$. For each $k$, let $\tau_k = (bin(x(0), \ldots, bin(x(k-1))$. Then $\tau_0 = \emptyset$ and, for each $k > 0$, $\tau_{k+1}$ can be computed from $\tau_k$ in time bounded by $|\tau_k|^c$. (Just start the computation of $h(\tau_k)$, stop it as soon as you have the first element $\rho = bin(x(k))$ in the list and then append $\rho$ to the end of $\tau_k$). Thus in particular $|\tau_{k+1}| \leq |\tau_k|^c$ for all

$k > 0$. Now choose $c$ large enough so that $|\tau_1| \leq 2^c$. It then follows by induction that, for all $k$, $|\tau_k| \leq 2^{c^k}$. It follows that the computation of $\tau_{k+1}$ from $\tau_k$ can be done in time bounded by $(2^{c^k})^{c-1} \leq 2^{c^{k+1}}$. Thus the entire computation of $\tau_n = (bin(x(0)), \ldots, bin(x(n)))$ from $1^n$ takes time bounded by

$$\Sigma_{k<n} 2^{c^{k+1}} < 2^{c^n+1} < 2^{(c+1)^n},$$

which shows that $x$ is computable in double exponential time in binary.

(b) Now suppose that $[T]$ is finite and let $x \in [T]$. By resticting $T$ to the extensions of $x \upharpoonright n$ for sufficiently large $n$, we may assume that $x$ is the unique infinite path through $T$. The result now follows immediately from the first part above.

(c) The proof is essentially the same as the proof of (a). Again we shall only give the proof in the case that $T$ is p-time in binary. The point is that if $h$ is linear time then it follows that for all $k \geq 1$, $\tau_{k+1}$ can be computed from $\tau_k$ in time $c \cdot |\tau_k|$ for some fixed constant $c$. If we pick $c$ so that $c \geq |\tau_0|$, then it is easy to prove by induction that $|\tau_k| \leq c^{k+1}$ for all $k \geq 0$. Thus the entire computation of $\tau_n = (bin(x(0)), \ldots, bin(x(n)))$ from $1^n$ takes time bounded by

$$\Sigma_{k<n} c^{k+2} < (n+1)c^{n+1} < c^{2n+2},$$

which shows that $x$ is computable in exponential time in binary.

Now if $[T]$ is finite and $x \in [T]$, then again by resticting $T$ to the extensions of $x \upharpoonright n$ for sufficiently large $n$, we may assume that $x$ is the unique infinite path through $T$. Then by the above argument if follows that $x \in DEXT$.

(d) The proof will be a minor modification of the proof of (a) above. Again the proofs are the same for tally and for binary so we will just give a binary version. Let $q$ be a p-time function such that for any $\tau = (bin(n_0), \ldots, bin(n_k) \in bin(T)$, $|\tau| \leq q(1^k)$. Since $|1^k| = k$, it follows that for some constant $b$ and all $k > 1$, we have $|\tau_k| \leq k^b$. Let $c$ be a number such that $h(\tau)$ can be computed from $\tau$ in time bounded by $|\tau|^{c-1}$ for all $\tau \in Bin(Ext(T))$ with $|\tau| > 1$. Then the computation of $\tau_{k+1}$ from $\tau_k$ can be done in time bounded by $|\tau_k|^c \leq k^{bc}$ for all $k > 1$. Now let $a$ be large enough so that $a > bc$ and also large enough so that $\tau_0$ and $\tau_1$ can both be computed in time bounded by $a$. Then the entire computation of $\tau_n = (bin(x(0)), \ldots, bin(x(n)))$ from $1^n$ takes time bounded by

$$a + 2^a + 3^a + \cdots + (n-1)^a \leq \Sigma_{k<n^a} k \leq n^{2a},$$

which shows that $x$ is computable in polynomial time in binary.

If we assume further that $[T]$ is finite, then the same argument as given in (a) and (b) above shows that every element of $[T]$ is polynomial time computable.

(d) As in (c), we may assume that if $\tau = (bin(n_0), \ldots, n_k) \in T$ and $k > 1$, then $|\tau| \leq k^b$ so that in particular $n_k \leq k^b$. In this case, we are not assuming that $T$ is locally p-time, so that we need a different algorithm for producing an infinite path $x$ in $[T]$. We will define $x(k)$ recursively by making $x(k)$ be the least number $n$ such that $(x(0), \ldots, x(k-1), n) \in T$. This means that we may have to check whether $(x(0), \ldots, x(k-1), x) \in T$ for all $x$ with $|bin(x)| \leq k^b$. This is where the binary representation differs from the p-time representation, because there will now be $2^{k^b}$ different strings to check. Each check will require time at most $(k^b)^c$, so that the computation of $bin(x(k))$ from $(bin(x(0)), \dots, bin(x(k-1))$ will require time less than $2^{k^{bc+b}}$ for $k > 1$. Now let $a$ be large enough so that $a \geq bc+b$ and also large enough so that $bin(x(0))$ and $bin(x(1))$ can be computed in time $\leq a$. Then the entire computation of $\tau_n = (bin(x(0)), \dots, bin(x(n)))$ from $1^n$ takes time bounded by

$$a + 2^{2^a} + 2^{3^a} + \cdots + 2^{n-1)^a} \leq \Sigma_{k<n^a} 2^k \leq 2^{n^{2a}},$$

which shows that $x \in EXPTIME$.

If we assume further that $[T]$ is finite, then the same argument as given in (a) above shows that every element of $[T]$ is $EXPTIME$. However, it is easy to show that the infinite paths through $T$ are actually $NP$ computable.

As above, we may assume that $T$ has no dead ends and has a unique infinite path $x$. Thus for any $k$, $x(0), x(1), \dots, x(k))$ is the unique finite path in $T$ with $k+1$ entries. Furthermore, since $T$ is p-time bounded, we know as above that $|(bin(x(0)), \dots, bin(x(k)))| \leq k^b$ for some fixed $b$. Thus to compute $x(k)$ nondeterministically, we simply guess a string $\sigma = (bin(n_0), \dots, bin(n_k))$ of length $\leq k^b$ and then use the p-time algorithm for $T$ to test whether $\sigma \in T$. When the answer is yes, we read the value of $x(k)$ from the end of $\sigma$. Since there is only one possible correct guess for $\sigma$, this procedure will compute $x(k)$. □

Next we shall give two examples to show that the bounds given in parts (a) and (b) of Theorem 7.1.6 can not be improved. Consider the following.

**Example 7.1.7.** *A locally p-time tree $T$ with a unique infinite path $x$ such that $T = EXT(T)$ and $x$ is double exponential time.*

*Let $x(n) = 2^{2^n}$ for all $n$ and let the tree $T$ consist of all initial segments of $x$. Then $(n_0, \dots, n_k) \in T$ if and only if $n_0 = 1$ and, for all $i < k$, $n_{i+1} = n_i^2$. It is clear that both $tal(T)$ and $bin(T)$ are p-time. Furthermore, for any $\sigma = (n_0, \dots, n_k) \in T$, we have $h(\sigma) = n_k^2$, so that $T$ is locally p-time in both binary and tally.*

**Example 7.1.8.** *A locally p-time tree $T$ with a unique infinite path $x$ such that $T = EXT(T)$, there is a linear time function $h$ such that for all $\sigma = (n_0, \dots, n_k) \in T$, $h(b(\sigma))$ lists all $b(n)$ such that $(n_0, \dots, n_k, n) \in T$ where $b() = tal()$ if $T$ is p-time in tally and $b() = bin()$ if $T$ is p-time in binary, and $x$ is exponential time.*

*Let $x(n) = 2^n$ for all $n$ and let the tree $T$ consist of all initial segments of $x$. Then $(n_0, \dots, n_k) \in T$ if and only if $n_0 = 1$ and, for all $i < k$, $n_{i+1} = 2n_i$. It is clear that both $tal(T)$ and $bin(T)$ are p-time and that the function $h$ is linear time in both cases, so that $T$ is locally p-time in both binary and tally.*

For $\Pi^0_1$ classes in $\{0,1\}^{\mathbb{N}}$, boundedness conditions are not needed and tally and binary representations are identical. Here are the basis results for classes of various complexity. These will be applied later to logical theories and other mathematical examples.

**Theorem 7.1.9.** *Let $P = [T]$ be a $\Pi^0_1$ class in $\{0,1\}^{\mathbb{N}}$, where $T$ has no dead ends.*

*(a) If $T$ is computable in time $nlog(n)^{O(1)}$, then $P$ has a member which is computable in time $nlog(n)^{O(1)}$.*

*(b) If $T$ is $LIN$, then $P$ has a member which is computable in time $O(n^2)$.*

*(c) If $T$ is $PTIME$, then $P$ has a $PTIME$ member.*

*(d) If $T$ is $DEXT$, then $P$ has a $DEXT$ member.*

*(e) If $T$ is $EXPTIME$, then $P$ has an $EXPTIME$ member.*

*Proof.* Part (c) follows from Theorem 7.1.6. We give the proof of (d) and leave the others as an exercise 2.

As in the proof of Theorem 7.1.6, we compute the desired path recursively so that $x(k) = 1$ if $(x(0), \ldots, x(k-1), 0) \in T$ and $x(k) = 1$ otherwise. By assumption, there exists $c$ so that this last step in the computation of $x(k)$ takes time $\leq 2^{c(k+1)}$ and thus the total computation requires time

$$\leq 2^c + 2^{2c} + \cdots + 2^{c(k+1)} \leq 2^{c(k+2)} = 2^{2c} \cdot 2^{ck},$$

plus a little time for bookkeeping. □

**Theorem 7.1.10.** *Let $P = [T]$ be a $\Pi^0_1$ class in $\{0,1\}^{\mathbb{N}}$, where $T$ has no dead ends.*

*(a) If $T$ is $LINSPACE$, then $P$ has a member which is computable in $LINSPACE$.*

*(b) If $T$ is $PSPACE$, then $P$ has a $PSPACE$ member.*

*(c) If $T$ is $EXPSPACE$, then $P$ has an $EXPSPACE$ member.*

*Proof.* We just sketch the proof of (a) and leave the others to the reader. To compute $x(k)$, we need to have stored $x(0), \ldots, x(k-1)$, whoch requires space $k$ and then test whether $x(0), \ldots, x(k-1), 0) \in T$, which requires additional space $ck$, where $c$ is fixed. This additional space may be reused, so that we can stay in $LINSPACE$. □

xxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxx

**Exercises**

7.1.1. Show that there is a highly computable tree $T$ with no dead ends such that there is not highly p-time tree wihout dead ends such that $[T] = [S]$.

7.1.2. Show that if $T$ is decidable in time $nlog(n)^{O(1)}$, then $T$ has an infinite path which is computable in time $nlog(n)^{O(1)}$, if $T$ is decidable in linear space, then $T$ has a infinite path which is computable in time $O(n^2)$ and if $T$ is $EXPTIME$, then $[T]$ has an $EXPTIME$ member.

## 7.2 Complexity of Structures

Complexity theoretic or feasible model theory is the study of resource-bounded structures and isomorphisms and their relation to computable structures and computable isomorphisms. This subject has been developed during the 1990's by Cenzer, Nerode, Remmel and others. See the survey article [34] for an introduction. Complexity theoretic model theory is concerned with infinite models whose universe, functions, and relations are in some well known complexity class such as polynomial time, exponential time, polynomial space, etc. By far, the complexity class that has received the most attention is polynomial time. One immediate difference between computable model theory and complexity theoretic model theory is that it is not the case that all polynomial time structures are polynomial time equivalent. For example, there is no polynomial isomorphism $f$ with a polynomial time inverse $f^{-1}$ which maps the binary representation of the natural numbers $Bin(\mathbb{N}) = \{0\} \cup \{1\}\{0,1\}^*$ onto the tally representation of the natural numbers $Tal(\mathbb{N}) = \{1\}^*$. This is in contrast with computable model theory where all infinite computable sets are computably isomorphic so that one usually only considers computable structures whose universe is the set of natural numbers $\mathbb{N}$.

There are two basic types of questions which have been studied in polynomial time model theory. First, there is the basic existence problem, i.e. whether a given infinite computable structure $\mathcal{A}$ is isomorphic or computably isomorphic to a polynomial time model. That is, when we are given a class of structures $\mathcal{C}$ such as a linear orderings, Abelian groups, etc., the following natural questions arise.

**(1)** Is every computable structure in $\mathcal{C}$ isomorphic to a polynomial time structure?

**(2)** Is every computable structure in $\mathcal{C}$ computably isomorphic to a polynomial time structure?

For example, the authors showed in [30] that every computable relational structure is computably isomorphic to a polynomial time model and that the standard model of arithmetic $(\omega, +, -, \cdot, <, 2^x)$ with addition, subtraction, multiplication, order and the 1-place exponential function is isomorphic to a polynomial time model. The fundamental effective completeness theorem says that any decidable theory has a decidable model. It follows that any decidable relational theory has a polynomial time model. These results are examples of answers to questions (1) and (2) above. However, one can consider more refined existence questions. For example, we can ask whether a given computable structure $\mathcal{A}$ is isomorphic or computably isomorphic to a polynomial time model with a standard universe such as the binary representation of the natural numbers, $Bin(\mathbb{N})$, or the tally representation of the natural numbers, $Tal(\mathbb{N})$. That is, when we are given a class of structures $\mathcal{C}$, we can ask the following questions.

**(3)** Is every computable structure in $\mathcal{C}$ isomorphic to a polynomial time structure with universe $Bin(\mathbb{N})$ or $Tal(\mathbb{N})$?

**(4)** Is every computable structure in $\mathcal{C}$ computably isomorphic to a polynomial time structure with universe $Bin(\mathbb{N})$ or $Tal(\mathbb{N})$?

It is often the case that when one attempts to answer questions of type (3) and (4) that the contrasts between computable model theory and complexity theoretic model theory become more apparent. For example, Grigorieff [75] proved that every computable linear order is isomorphic to a Ptime linear order which has universe $Bin(\mathbb{N})$. However Grigorieff's result can not be improved to the result that every computable linear order is computably isomorphic to a Ptime linear order over $Bin(\mathbb{N})$. For example, Cenzer and Remmel [30] proved that for any infinite polynomial time set $A \subseteq \{0,1\}^*$, there exists a computable copy of the linear order $\omega + \omega^*$ which is not computably isomorphic to any polynomial time linear order which has universe $A$. Here $\omega + \omega^*$ is the order obtained by taking a copy of $\omega = \{0, 1, 2, \ldots\}$ under the usual ordering followed by a copy of the negative integers under the usual ordering.

The general problem of determining which computable models are isomorphic or computably isomorphic to feasible models has been studied by the authors in [30], [31], and [33]. For example, it was shown in [31] that any computable torsion Abelian group $G$ is isomorphic to a polynomial time group $A$ and that if the orders of the elements of $G$ are bounded, then $A$ may be taken to have a standard universe, i.e. either $Bin(\mathbb{N})$ or $Tal(\mathbb{N})$. It was also shown in [31] that there exists a computable torsion Abelian group which is not isomorphic, much less computably isomorphic, to any polynomial time (or even any primitive recursive) group with a standard universe. Feasible linear orderings were studied by Grigorieff [75], by Cenzer and Remmel [30], and by Remmel [162, 163]. Feasible vector spaces were studied by Nerode and Remmel in [146] and [147]. Feasible Boolean algebras were studied by Cenzer and Remmel in [30] and by Nerode and Remmel in [145]. Feasible permutation structures and feasible Abelian groups were studied by Cenzer and Remmel in [31] and [33]. By a *permutation structure* $\mathcal{A} = (A, f)$, we mean a set $A$ together with a unary function $f$ which maps $A$ one-to-one and onto $A$.

General conditions were given in [34] which allow the construction of models with a standard universe such as $Tal(\mathbb{N})$ or $Bin(\mathbb{N})$ and these conditions were applied to graphs and to equivalence structures. For example, it was shown that any computable graph with all but finitely many vertices of finite degree is computably isomorphic to a polynomial time graph with standard universe. On the other hand, a computable graph was constructed with every vertex having either finite degree or finite co-degree (i.e. joined to all but finitely many vertices) which is not computably isomorphic to any polynomial time graph with a standard universe. An *equivalence structure* $\mathcal{A} = (A, R^{\mathcal{A}})$ consists of a set $A$ together with an equivalence relation. It was also shown that any computable equivalence structure is computably isomorphic to a polynomial time structure with a standard universe.

In this section, we want to consider the connection between computable structures and resource-bounded structures and the corresponding connection between computable trees and resource-bounded trees as developed in section

7.1.

A relational structure is simply a structure which has no functions. We will present an improved version of the theorem (first due to Grigorieff [75]) from [30] that every computable relational structure is computably isomorphic to a polynomial time structure. This theorem will be our primary tool in the analysis of computable combinatorial structures. It is important to note that the polynomial time structure provided will have for its universe a polynomial-time set possibly different from $\{1\}^*$ or $\{0,1\}^*$. An example is constructed in [30] which shows that the theorem fails if any fixed polynomial time set $A$ is specified in advance as the universe of the structure. The improved version of the theorem presented here applies to structures with two distinct types of objects, the first type being the normal universe of the structure, and with functions which map the first type into the second type. The type of example that we have in mind is a function from the vertices of a graph into the natural numbers which computes the degree of a vertex. The universe of the graph is now expanded by adding a p-time set which represents the natural numbers and the degree function now becomes part of the structure. Naturally, the new objects are not vertices and therefore are not joined to any other objects by edges.

**Theorem 7.2.1.** *Let*

$$\mathcal{C} = (C, A, B, \{R_i^{\mathcal{C}}\}_{i\in S}, \{f_i^{\mathcal{C}}\}_{i\in T},),$$

*be a computable structure such that*

**(i)** *$A$ and $B$ are disjoint subsets of $C$ with $C = A \cup B$ and $B$ is a polynomial time set.*

**(ii)** *there is a computable isomorphism from $Bin(\omega)$ onto a subset of $Bin(\omega)\setminus B$ with a p-time inverse.*

**(iii)** *for each $i \in T$, $f_i$ maps $C$ into $B$.*

**(iv)** *for each $i \in S$, the relation $R_i$ is independent of $B$, that is, for any $(x_1, \ldots, x_n) \in C^n$, where $n = s(i)$, any $j \le n$ such that $x_i \in B$, and any $b \in B$, $R_i^{\mathcal{C}}(x_1, \ldots, x_n)$ if and only if $R_i^{\mathcal{C}}(x_1, \ldots, x_{j-1}, b, x_{j+1}, \ldots, x_n)$.*

**(v)** *for each $i \in T$, the function $f_i$ is independent of $B$, that is, for any $(x_1, \ldots, x_n) \in C^n$, where $n = t(i)$, any $j \le n$ such that $x_i \in B$, and any $b \in B$, $f_i^{\mathcal{C}}(x_1, \ldots, x_n) = f_i^{\mathcal{C}}(x_1, \ldots, x_{j-1}, b, x_{j+1}, \ldots, x_n)$.*

*Then there is a computable isomorphism $\phi$ of $\mathcal{C}$ onto a p-time structure $\mathcal{M}$ such that $\phi(b) = b$ for all $b \in B$.*

*Proof.* The idea of the proof is that we will replace each element $x$ of $A$ by a string $y$ which codes $x$ and is long enough to allow us to compute whether $x \in A$ in time $|y|$ and also to compute the relations and functions on $A$ in time $|y|$ for all inputs which are less than or equal to $x$. These new strings may accidentally be in the set $B$, which must be kept disjoint from $A^{\mathcal{M}}$. This is the reason for the p-time mapping which takes an arbitrary string to one which is not in $B$. Let $\psi$ be a p-time map from $Bin(\omega)$ into $Bin(\omega) \setminus B$ such that $\psi^{-1}$ is also p-time. We can assume that $A$ is an infinite set, since, if $A$ is finite, then $\mathcal{C}$ is p-time itself. Let $\sigma_0, \sigma_1, \ldots$ be an effective enumeration of $A$ in the usual order. Let $b_0$ be the shortest element of $B$. For any $x \in A$, we let $\nu(x)$ denote the number of steps needed to run the following algorithm.

> First start to list $\sigma_0, \sigma_1, \ldots$ until we find an $s$ such that $\sigma_s = x$. Next for each $i \leq s$ such that $i \in S \cup T$, list all sequences $(x_1, \ldots, x_n)$ from $\{b_0, \sigma_0, \ldots, \sigma_s\}^n$ for $n = s(i)$ or $t(i)$ and then, for $i \in S$, compute whether $R_i(x_1, \ldots, x_n)$ holds and, for $i \in T$, compute $f_i^{\mathcal{C}}(x_1, \ldots, x_n)$.

Observe that the algorithm is completely uniform in $x$ because our definition of computable structure ensures that there is a computable relation $R$ such that $R(i, \langle x_1, \ldots, x_{t(i)} \rangle) \iff R_i(x_1, \ldots, x_{t(i)})$ and a computable function $f$ such that $f(i, \langle x_1, \ldots, x_{t(i)} \rangle) = f_i(x_1, \ldots, x_{t(i)})$ Note that in order to obtain the list $\sigma_0, \ldots, \sigma_s$, we have to test whether $a \in A$ for all $a \leq x$. We then define a structure

$$\mathcal{M} = (M, \{R_i^{\mathcal{M}}\}_{i \in S}, \{f_i^{\mathcal{M}}\}_{i \in T})$$

as follows. For each $a \in A$, let $\phi(a) = \psi(a^\frown 0^\frown 1^{\nu(a)})$ and, for each $b \in B$, we let $\phi(b) = b$. It is clear that $\phi$ is a computable isomorphism from $C$ onto a subset $M$ of $Bin(\omega)$, that $\phi(B) = B$ and that $\phi(A)$ is disjoint from $B$. The structure $\mathcal{M}$ is the image of $\mathcal{C}$ under the isomorphism $\phi$. This means that $A^{\mathcal{M}} = \{\phi(a) : a \in A\}$, $B^{\mathcal{M}} = B$, and $M = A^{\mathcal{M}} \cup B^{\mathcal{M}}$. For each $i \in S$ and $(x_1, \ldots, x_n) \in C$, $R_i^{\mathcal{M}}$ is defined by

$$R_i^{\mathcal{M}}(\phi(x_1, \ldots, \phi(x_n)) \iff R_i^A(x_1, \ldots, x_n),$$

where $s(i) = n$. For each $i \in T$, $f_i^{\mathcal{M}}$ is defined by

$$f_i^{\mathcal{M}}(\phi(x_1), \ldots, \phi(x_n)) = \phi(f_i^{\mathcal{A}}(x_1, \ldots, x_n)),$$

where $t(i) = n$.

It is clear that the function $\phi$ is a computable isomorphism from $\mathcal{A}$ onto $\mathcal{M}$. It remains to be seen that $\mathcal{M}$ is a polynomial time structure, that is, that $M$ is a polynomial time set and that each relation $R^M$ and function $f^M$ is p-time.

We show that $M$ is p-time as follows. It clearly suffices to show that $A^{\mathcal{M}}$ is p-time, since $B^{\mathcal{M}} = B$ is p-time. The procedure for testing whether an input $y$ is in $A^{\mathcal{M}}$ is to compute $\psi^{-1}(y)$, check to make sure that it has a 0 in it, and then determine $x$ and $n$ such that $\psi^{-1}(y) = x^\frown 0^\frown 1^n$. Then we simply run the algorithm outlined above to input $x$ for $n$ steps. Then $y \in A^{\mathcal{M}}$ if and only if the algorithm terminates in exactly $n$ steps and gives the answer that $x \in A$.

We show that the function $f_i^{\mathcal{M}}$ is p-time as follows. Fix $i$ and let $f = f_i$, let $n = t(i)$ and let $c$ be the maximum amount of time required to compute $f^C(x_1, \ldots, x_n)$ when $\{x_1, \ldots, x_n\} \subseteq \{b_0, \sigma_0, \sigma_1, \ldots, \sigma_{i-1}\}$. Now given input

$(y_1, \ldots, y_n)$, where each $y_i \in M$, the procedure for computing $f^{\mathcal{M}}(y_1, \ldots, y_n)$ is the following. First replace every $x_i \in B$ with $x_i' = b_0$ and let $x_i' = x_i$ for $x_i \in A$. Then compute $f^C(x_1', \ldots, x_n')$. We claim that this computation takes time at most $c + max\{|y_j| : 1 \leq j \leq n\}$. There are two cases of this claim to consider. First, if $\{x_1', \ldots, x_n'\}$ is a subset of $\{b_0, \sigma_0, \ldots, \sigma_{i-1}\}$, then, by the definition of $c$, the computation takes at most $c$ steps. On the other hand, if at least one of the $x_j' = x_j = \sigma_s$ for some $s \geq i$, then by the definition of $\nu$, the computation takes less than $\nu(x_j)$ steps for some j; but of course $\nu(x_j) < |y_j| \leq max\{|y_j| : 1 \leq j \leq n\}$.

The argument for the relations is similar. This completes the proof of Theorem 7.2.1. □

For an example, let $(\mathbb{N}, R, 0, f)$ be a computable structure where $R$ is a binary relation defining a tree with root 0 on the set of even numbers and $f$ is an injection mapping the even numbers onto the odd numbers so that $f(0) = 1$ and, for each $n$, $R(f^{-1}(2n+1), f^{-1}(2n+3))$. That is, $f$ defines an infinite computable path through $T$. (We assume that $R(m, n)$ implies that both $m$ and $n$ are even.) Then the theorem provides a polynomial time tree with a polynomial time infinite path starting from the root.

## 7.3 Propositional Logic

In this section, we shall consider the complexity of theories in propositional logic and of the corresponding $\Pi_1^0$ class of complete consistent extensions of the theory.

It is first necessary to define the length $|\phi|$ of a formula $\phi$. Suppose that the underlying set of propositional letters in our propositional language is $\{A_0, A_1, \ldots\}$. In the standard or binary representation of a sentence $\phi$, the numeral $i$ in a propositional letter $A_i$ is written in binary representation $bin(i)$ so that the length $|A_i|$ in binary is $1 + |bin(i)|$. That is, $|bin(A_i)| = r + 2$ when $2^r \leq i < 2^{r+1}$. In the tally representation, the numeral $i$ is written as $1^i$ so that $|tal(A_i)| = i + 1$. A complete consistent theory $\Gamma$ is represented by a subset of $\omega$, $S(\Gamma) = \{i : A_i \in \Delta\}$, or, equivalently, by the characteristic function in $\{0,1\}^\omega$ of $S(\Gamma)$. The set of all complete consistent extensions of a consistent set $\Delta$ of sentences is denoted as $CC(\Delta)$. We shall let a finite sequence $\sigma \in \{0,1\}^n$ represent the sentence $B(\sigma) = B_0 \wedge B_1 \wedge \ldots B_n$, where $B_i = A_i$ if $\sigma(i) = 1$ and $B_i = \neg A_i$ if $\sigma(i) = 0$.

We note that there is a lower limit on the complexity of non-trival propositional theories. To be more precise, the set $SAT$ of consistent, or satisfiable, sentences is the classic $NP$ complete set. Now a sentence $\phi$ is valid if and only if $\neg\phi$ is not satisfiable. Thus the smallest theory, the set of valid sentences is $Co$-$NP$ complete. On the other hand, any complete propositional theory is determined by its underlying set of literals. That is, let $V = \{A_0, A_1, \ldots\}$ be a set of propositional variables, $S$ be any subset of $V$ and $\Gamma(S)$ be the consequences of $\{A_i : i \in S\} \cup \{\neg A_i : i \notin S\}$. Then $S$ is computable from $\Gamma(S)$ in constant time. On the other hand, given any sentence $\phi$ containing variables $A_{i_1}, \ldots, A_{i_k}$, we can decide whether $\phi \in \Gamma(S)$ by first making each $A_t$ true if it is in $S$ and false if not, and then evaluating $\phi$. That is, $\phi \in \Gamma(S)$ if and only if the value of $\phi$ is true. Thus $\Gamma(S)$ is computable from $S$ in linear time and linear space. Thus there are complete propositional theories in any of the standard complexity classes such as linear time, linear space, polynomial time, polynomial space, etc.

**Lemma 7.3.1.** *$tal(B(\sigma))$ has length $O(n^2)$ and may be computed in time $O(n^2)$ and $Bin(B(\sigma))$ has length $O(n \cdot log\ n)$ and may be computed in time $O(n\ log\ n)$.*

*Proof.* The sentence $B(\sigma)$ contains the atoms $A_0, A_1, \ldots, A_{n-1}$, $n-1$ conjunction symbols $\wedge$ and between 0 and $n$ negation symbols $\neg$. The total length of the atoms in tally is

$$2 + 2 + 3 + 4 + \cdots + n = \frac{n^2}{2} + \frac{n}{2} + 1,$$

so that

$$\frac{n^2}{2} + \frac{3n}{2} \leq |tal(B(\sigma)| \leq \frac{n^2}{2} + \frac{5n}{2}$$

In binary, suppose first that $n = 2^{k+1} - 1$. Then the total length of the atoms is

$$2 \cdot 2 + \sum_{j=1}^{k} (j+1)2^{j-1}$$

so that the total length of the atoms is strictly between $(k+1)2^{k-1} + 1$ and $(k+1)2^k$ and the length of $bin(B(\sigma))$ is between $(k+5)2^{k-1}$ and $(k+5)2^k$. Now suppose that $k \leq log(n) \leq k+1$, so that $2^k \leq n < 2^{k+1}$. It follows that $(5 + log(n))n/2 \leq (k+5)2^{k-1} \leq |bin(B(\sigma))| \leq (k+5)2^k \leq (5 + log(n)) \cdot 2n$. $\square$

A set $\Delta$ of sentences is said to be *$P$-decidable* in binary (in tally) if there is a polynomial time Turing machine which given as input the binary (tally) representation of a formula $\phi$, computes 1 if $\Delta \vdash \phi$ and computes 0 otherwise. We say that $\Delta$ is *weakly $P$-decidable in binary* (in tally) if there is a polynomial time Turing machine which given as input the binary (tally) representation of a conjunction $\phi$ of literals, computes 1 if $\phi \in SAT(\Delta)$ and computes 0 otherwise. One can define the notion of $\Delta$ being (weakly) $\mathcal{C}$-decidable in binary or tally for any complexity class $\mathcal{C}$ in a similar manner. A theory $\Gamma$ is said to be $P$-axiomatizable if it possesses a polynomial time set $\Delta$ of axioms such that $\Gamma = Cn(\Delta)$. Again similar definitions apply to other notions of complexity.

Recall that the tree $T$ represents $CC(\Delta)$, the set of complete consistent extensions of $\Delta$, if the set $[T]$ of infinite paths through $T$ equals the family of sets $S \subseteq \{A_0, A_1, \ldots\}$ such that $\Gamma(S)$ is a complete consistent extension of $\Delta$. The canonical tree $T$ which represents $CC(\Delta)$ is given by $\sigma \in T \iff B(\sigma) \in SAT(\Delta)$.

**Theorem 7.3.2.** *Let $\Delta$ be a propositional theory.*

*(a) If $\Delta$ is weakly $DTIME(n\ log(n)^{O(1)})$ decidable in binary, then $CC(\Delta)$ may be represented as the set of paths through a tree in $DTIME(n\ log(n)^{O(1)})$.*

*(b) If $\Delta$ is weakly $P$-decidable (respectively $PSPACE$ decidable) in either binary or tally, then $CC(\Delta)$ may be represented as the set of paths through a $P$-tree (resp. $PSPACE$-tree).*

*(c) If $\Delta$ is weakly $DEXT$-decidable (respectively, $EXPSPACE$-decidable) in tally or binary, then $CC(\Delta)$ may be represented as the set of paths through an $EXPTIME$-tree (resp. $\bigcup_{k\in\omega} DSPACE(2^{n^k})$-tree).*

*Proof.* In each case, we shall let $T$ be the canonical tree which represents $CC(\Delta)$. That is, $\sigma \in T \iff B(\sigma) \in SAT(\Delta)$.

(a) Suppose that $\Delta$ is weakly $DTIME(n\ log(n)^{O(1)})$ decidable in binary. By Lemma 7.3.1, we can compute $bin(B(\sigma))$ from $\sigma$ in time $O(n\ log\ n)$, so that $T$ is in $DTIME(n\ log(n)^{O(1)})$.

(b) It easily follows from Lemma 7.3.1 that we can compute $bin(B(\sigma))$ and $tal(B(\sigma))$ in polynomial time and space from $\sigma$. Thus if $\Delta$ is weakly $P$-decidable (weakly $PSPACE$-decidable), then $T$ is a $P$-tree ($PSPACE$-tree).

(c)If $\Delta$ is weakly $DEXT$-decidable in tally ($EXPSACE$-decidable), it will require on the order of $2^{|\sigma|^2}$ time (space) to determine if $B(\sigma) \in SAT(\Delta)$ so that $T$ is an $EXPTIME$-tree ($\bigcup_{k\in\omega} DSPACE(2^{n^k})$-tree). Similarly if $\Delta$ is weakly $DEXT$-decidable in binary ($EXPSACE$-decidable), it will require on the order of $2^{|\sigma| log(|\sigma|)}$ time (space) to determine if $B(\sigma) \in SAT(\Delta)$ so that again $T$ is an $EXPTIME$-tree ($\bigcup_{k\in\omega} DSPACE(2^{n^k})$-tree). □

Thus we have the following corollary.

**Corollary 7.3.3.** *Let $\Delta$ be a propositional theory.*

*(a) If $\Delta$ is weakly $P$-decidable (respectively $PSPACE$ decidable) in tally or binary, then $\Delta$ has a $P$-decidable (resp. $PSPACE$-decidable) complete consistent extension in tally*

*(b) If $\Delta$ is weakly $DEXT$-decidable (respectively, $EXPSPACE$-decidable) in tally or binary, then $\Delta$ has a complete consistent extension which is $EXPTIME$ decidable (resp. $EXPSPACE$ decidable) in tally.*

*Proof.* (a) Let $\Delta$ be weakly $PTIME$ decidable in tally or binary. By Theorem 7.3.2, $CC(\Delta)$ may be represented as the set of paths through a $P$-decidable tree $T$. It follows from Theorem 7.1.9 that $T$ has an infinite $PTIME$ path $x \in \{0,1\}^{\mathbb{N}}$. The complete consistent extension $\Gamma$ corresponding to $x$ has axioms $A_i$ for $x(i) = 1$ and $\neg A_i$ for $x(i) = 0$. Thus given an arbitrary sentence $\phi(A_0, \ldots, A_n)$ of length $geqn$, we can first compute $x(0), \ldots, x(n))$ in polynomial time and then use this to substitute true and false for the occurrences of the propositional variables in $\phi$ to compute the value of $\phi$. This can certainly be done in polynomial time in tally. The proof for $PSPACE$ is similar.

(b) The proof is similar to (a). □

For the binary representation, note that $|A_n|$ is of order $logn$ and hence it may take exponential time to decide $A_n$ from a polynomial time $x \in \{0,1\}^{\mathbb{N}}$. Thus we have

**Corollary 7.3.4.** *Let $\Delta$ be a propositional theory.*

*(a) If $\Delta$ is weakly $P$-decidable (respectively $PSPACE$ decidable) in either binary or tally, then $\Delta$ has a $DEXT$-decidable (resp. $EXPSPACE$-decidable) complete consistent extension in binary.*

*(b) If $\Delta$ is weakly $EXPTIME$-decidable (respectively, $EXPSPACE$-decidable) in tally or binary, then $\Delta$ has a complete consistent extension which is $DOUBEXT$ decidable (resp. $DOUBEXPSPACE$ decidable) in binary.*

Since any $PTIME$ decidable theory is certainly weakly $PTIME$ decidable and likewise for other comlexity classes, these results hold with the "weakly" removed from the hypothesis.

It was shown in [40] that this difference in the complexity of the complete consistent extension between the tally and binary representations is necessary. That is,

**(1)** There is a propositional theory which is $NP$-decidable in binary but has no $P$-decidable complete consistent extension in binary.

**(2)** There is a propositional theory which is $DEXT$-decidable in binary but has no $EXPTIME$-decidable complete consistent extension in binary.

There are no nice basis results for *axiomatiable* theories. The corresponding representation results for axiomatizable theories do not require any restriction on the complexity of the set of axioms. In fact, our next result strengthens Theorem 4.1 of [32] which showed that any $\Pi^0_1$ class may be represented as the set of paths through a polynomial time tree.

A computable function $f$ is said to be *time constructible* if and only if there is a Turing machine which on every input of size $n$ halts in exactly $f(n)$ steps. In particular, the functions $log_2^k(n)$ are time constructible for $k \geq 1$ where we define $log_2^k(n)$ by induction as $log^1(n) = log(n)$ and for $k > 1$, $log_2^k(n) = log_2(log_2^{k-1}(n))$.

It was shown in [40] that for any time constructible function $f$ which is nondecreasing and unbounded and any axiomatizable propositional theory $\Gamma$, $\Gamma$ has a $DTIME(O(f))$ set of axioms and may be represented as the set of paths through a $DTIME(O(f))$-tree. Note that this is not necessarily a decidable tree.

The results for decidable trees are somewhat surprising. Let us first give a few definitions. Recall that $SAT$ is the set of satisfiable, or consistent, propositional sentences and is the standard $NP$-complete set.

**Theorem 7.3.5.** *The following are equivalent:*

**(a)** *$P = NP$;*

**(b)** *Every $P$-decidable tree represents the set of complete consistent extensions of some theory which is $P$-decidable in tally.*

*(b) $\rightarrow$ (a).* Let $T = \{0,1\}^*$ and suppose that $\Delta$ is a theory which is $P$-decidable in tally such that $\{0,1\}^\omega = [T]$ represents the set of complete consistent extensions of $\Delta$. Then it is easy to see that $SAT(\Delta) = SAT$. But this means that

$$\phi \in SAT \iff \neg[\Delta \vdash \neg\phi]. \tag{7.1}$$

Since $\Delta$ is $P$-decidable in tally, (7.1) would imply that $SAT$ is polynomial time and hence $P = NP$.

[(a) $\rightarrow$ (b)] Next suppose that $P = NP$ and let $T$ be a $P$-decidable tree. Let $\phi(A_0, \ldots, A_n)$ be a propositional formula whose propositional letters are a a subset of $\{A_0, \ldots, A_n\}$ and which contains $A_n$. The canonical theory $\Delta$ such that $[T]$ represents the set of complete consistent extensions of $\Delta$ is defined by

$$\Delta \vdash \phi(A_0, \ldots, A_n) \iff (\forall \sigma \in T \cap \{0,1\}^n)(B(\sigma) \vdash \phi(A_0, \ldots, A_n)).$$

We will show that $SAT(\Delta)$ is $NP$ and hence in $P$ by our assumption. In tally, $n \leq |B(\sigma)| \leq 2n^2$, so that for each $\sigma \in \{0,1\}^n$, we can test whether $B(\sigma) \vdash \neg\phi(A_0, \ldots, A_n)$ in polynomial time in the length of $\phi(A_0, \ldots, A_n)$. Thus we can test $\phi(A_0, \ldots, A_n) \in SAT(\Delta)$ in the usual $NP$ fashion, by guessing a string $\sigma$ of length $n$ and checking that $\sigma \in T$ and $B(\sigma)$ implies $\phi(A_0, \ldots, A_n)$. Thus $SAT(\Delta)$ is in $P$. □

The corresponding result does not follow relative to the binary representation of theories. That is, the direction [(ii) $\rightarrow$ (i)] still holds, since the $SAT$ problem is $NP$-complete in either tally or binary. However, the argument given for the reverse direction only shows that $SAT(\Delta)$ is $DTIME(2^{O(1)})$-decidable in binary. This is due to the fact that a short formula $\phi$ with a high numbered variable, such as a propositional variable $A_{2^n-1}$ requires us to check whether $B(\sigma) \vdash \phi(A_0, \ldots, A_n)$ for $|\sigma| = 2^n - 1$ which would require time of order $2^n$ since $|B(\sigma)| \geq 2^n$. Thus since $|A_{2^n-1}| = n + 1$, such a check would require exponential time in $|\phi|$.

# Part B

# Applications of $\Pi_1^0$ Classes

Effectively closed sets arise naturally in the study of computable mathematics. In many problems associated with mathematical structures, such as the problem of finding a 4-coloring of a planar graph, the family of solutions may be viewed as a closed set under some natural topology. Thus for a computable structure, the set of solutions may be viewed as a $\Pi^0_1$ class. For another example, the set of zeroes of a continuous real function on the unit interval is a closed set and the set of zeroes of a computable real function will be a $\Pi^0_1$ class.

We will say that the $\Pi^0_1$ class $P$ *represents* the set of solutions to a given problem if there is a one-to-one degree preserving correspondence between the elements of the class $P$ and the solutions to the problem. It will be important whether the class $P$ is bounded, or computably bounded. For example, the set of 4-colorings of a given computable graph $G$ may be represented as a subclass of $\{0, 1, 2, 3\}^{\mathbb{N}}$ and is therefore computably bounded. Then we may apply the basis results surveyed in Chapter 3, for example, Theorem 2.2.15, and conclude that if $G$ has a 4-coloring, then it has a 4-coloring of c.e. degree.

Now a fundamental observation is that computable problems, such as the graph-coloring problem, often do not have computable solutions. The results from Chapter 3 on special $\Pi^0_1$ classes strengthen in various ways the basic result that a computably bounded $\Pi^0_1$ class may not have any computable members. In order to be able to transfer these results to results about the degrees of solutions to a given computable mathematical problem of a given type, one must establish that *every* computably bounded $\Pi^0_1$ class represents the set of solutions to some computable problem of that type. For example, Remmel [161] showed that, up to a permutation of the colors, every computably bounded $\Pi^0_1$ class represents the set of 3-colorings of some computable graph. It then follows from Theorem 3.2.4 that for any c.e. degree $\mathbf{c}$, there is a computable, 3-colorable graph $G$ such that every 3-coloring of $G$ has degree $\geq \mathbf{c}$.

It is very important to make the representation effective. For each type of mathematical problem, we shall establish a natural enumeration of the computable (and sometimes the c.e.) problems and then use the effective correspondence between solution sets and $\Pi^0_1$ classes to transfer results on index sets from Chapter 5. For example, we will show that the set of indices of computable graphs which have a computable 3-coloring is a $\Sigma^0_3$ complete set.

Many important problems in the history of $\Pi^0_1$ classes come from mathematical logic. The problem associated with a logical theory $\Gamma$ is to find a complete consistent extension of $\Gamma$. For any effectively presented language $\mathcal{L}$, the set of sentences of $\mathcal{L}$ may be effectively listed as $\{\phi_n : n \in \mathbb{N}\}$. Then $\Gamma$ is said to be *decidable* if $\{n : \phi_n \in \Gamma\}$ is a computable set and $\Gamma$ is said to be *axiomatizable* if $\{n : \phi_n \in \Gamma\}$ is a c.e. set.

Shoenfield [174] showed in 1960 that the set of complete consistent extensions of a axiomatizable first theory can always be represented by a $\Pi^0_1$ class. The classical undecidability theorem of Turing and Church may be viewed as showing that the $\Pi^0_1$ class of complete consistent extensions of Peano Arithmetic has no computable element. The complete consistent extensions of a decidable theory $\Gamma$ may be represented by a decidable $\Pi^0_1$ class.

Ehrenfeucht [63] showed in 1961 that, conversely, every computable bounded

$\Pi_1^0$ class $P$ represents the set of complete consistent extensions of some axiomatizable theory $\Gamma$. In particular, $\Gamma$ may be a theory of propositional logic or a theory for the language with a single, binary relation symbol. If $P$ is decidable, then we may take $\Gamma$ to be a decidable theory.

The chapters below include proofs of the theorems cited above, together with applications of results on members of $\Pi_1^0$ classes and on index sets for $\Pi_1^0$ classes.

# Chapter 8

# Algebra

Three types of computable and computably enumerable algebraic structures are considered: Boolean algebras, abelian groups, and commutative rings with unity. The associated problems are to find proper, prime and maximal ideals of rings and Boolean algebras and to find proper and maximal subgroups of abelian groups.

The set of prime ideals of a c. e. Boolean algebra or commutative ring with unity may always be represented by a c. b. $\Pi_1^0$ class, and the set of maximal ideals of a computable Boolean algebra may be represented by a *decidable* c. b. $\Pi_1^0$ class. The set of maximal ideals of a c. e. commutative ring with unity or of a c. e. Boolean algebra may always be represented by a $\Pi_2^0$ class.

For the reverse direction, any c. b. $\Pi_1^0$ class $P$ may be represented by the set of maximal ideals of a c. e. Boolean algebra $\mathcal{B}$ and if $P$ is decidable, then $\mathcal{B}$ may be taken to be computable. Finally, any $\Pi_1^0$ class of separating sets may be represented as the set of prime ideals of some c. e. commutative ring with unity [66].

A recursively presented group, ring, or field consists of a recursive subset $U$ of $\omega$, the universe of the structure, together with appropriate partial recursive functions over $U$ for addition, subtraction, multiplication and/or division functions as required. Unless, explicitly stated otherwise, we will assume that all our structures are countably infinite so that there is no loss in generality in assuming that the underlying universe is $\omega$. A *c. e. ring* is the quotient of a recursive ring modulo an *r.e.* ideal, a *c. e. group* is the quotient of a computable group modulo a c. e. normal subgroup and a *c. e. Boolean Algebra* is the quotient of a recursive Boolean Algebra modulo a c. e. ideal.

We will show that the set of prime ideals of c. e. commutative ring with unity and the set of prime ideals of a c. e. Boolean algebra can always be represented by a c. b. $\Pi_1^0$ class. We will show that the set of maximal ideals of a c. e. commutative ring with unity and the set of maximal subgroups of a c. e. group can always be represented by a $\Pi_2^0$ class. We shall also show that the set of all ideals or the set of all maximal ideals of a recursive Boolean algebra can be represented as the set of paths through a recursive tree with no dead ends.

Reversing such results, we will show that any c. b. bounded $\Pi^0_1$ class can be strongly represented by the set of maximal ideals of an c. e. Boolean algebra. We show that the set of paths through any recursive tree $T$ with no dead ends can be represented as the set of maximal ideals of a recursive Boolean algebra. We shall also show that the set of separating set $S(A, B)$ of a pair of c. e. sets can be represented by the set of prime ideals or the set of maximal ideals of a c. e. commutative ring with identity.

We refer the reader to Downey [56] for a general survey of computable algebra.

Some definitions are needed. Recall that a subset $H$ of an Abelian group $G = (G, +^G, -^G, 0^G)$ is a subgroup if it satisfies the following conditions:

**(i)** $0^G \in H$.

**(ii)** $a \in H$ and $b \in H$ implies $a -^G b \in H$.

$H$ is a *maximal subgroup* if, in addition, there is no subgroup $J$ of $G$ such that $H \subset J \subset G$.

A subset $I$ of a commutative ring with unity $R = (R, +^R, -^R, \cdot^R, 0^R, 1^R)$ is an ideal $I$ is a subgroup of $R = (R, +^R, 0^R)$ and it satisfies the following additional conditions:

**(iii)** $a \in I$ and $r \in R$ implies $a \cdot^R b \in I$.

**(iv)** $1^R \notin I$.

$I$ is a *prime ideal* if, in addition,

**(v)** $a \cdot^B b \in I$ implies $a \in I$ or $b \in I$.

$I$ is a *maximal ideal* if, in addition, there is no ideal $J$ such that $I \subset J$. It is easy to see that any maximal ideal is prime, but the converse is not always true.

The classical results that every proper subgroup of a group has an extension to a maximal (and therefore proper) subgroup and that every ideal in a ring has an extension to a maximal (and therefore prime) ideal follow easily from Zorn's Lemma. In particular, if the commutative ring $R$ with unity is not a field, then $R$ has, for each non-unit $a$ a proper ideal $Ra = \{ra : r \in R\}$ and therefore has a maximal ideal.

Any Boolean algebra $(B, \vee^B, \wedge^B, \neg^B, 0^B, 1^B)$ may be viewed as a commutative ring with unity where $a \cdot^B b = a \wedge b$ and $a + b = (a \wedge^B \neg^B b) \vee^B (\neg^B a \wedge^B b)$. In a Boolean ring any prime ideal is maximal, so it follows from the Boolean algebra results that, for any $\Pi^0_1$ class $P$, there is a c. e. commutative ring with unity such that $Max(R) = Prime(R)$ is represented by $P$. However, there turns out to be a significant difference between Boolean rings and rings in general. The proof that any recursive Boolean ring has a recursive maximal ideal cannot be extended to arbitrary rings and in fact, a recursive ring need not have a recursive maximal ideal. This naturally led to the conjecture that any $\Pi^0_1$ class could be represented as the set of prime ideals of some commutative ring. By considering rings of polynomials, Friedman-Simpson-Smith obtained in [67] the partial result that any $\Pi^0_1$ class of separating sets can be represented as the set of prime ideals of some recursive commutative ring with unity.

## 8.1 Boolean algebras

The Stone Representation Theorem implies that every Boolean algebra is isomorphic to the Boolean algebra of clopen sets of a topological space (indeed of a Boolean space). If the Boolean algebra is countable, the proof shows that it is isomorphic to the Boolean algebra $RC(P)$ of relatively clopen sets of a closed class P contained in $\{0,1\}^{\mathbb{N}}$, and of course $RC(P)$ is countable for every closed class $P$ contained in $\{0,1\}^{\mathbb{N}}$. In this section we point out effectivized versions of this correspondence and use them to transfer some of our results on $\Pi^0_1$ classes to results on computable and c. e. Boolean algebras. In particular, we give an effective version of the Stone Representation Theorem, that every computable (c. e. ) Boolean algebra is isomorphic to the set of its prime ideals. We determine the meaning of thinness and of the Cantor-Bendixson derivative in the setting of Boolean algebras. We also look at the connection between computable Boolean algebras and theories of propositional calculus, in particular with Martin–Pour-El theories. Finally, we interpret the results of the previous sections on $\Pi^0_1$ classes for computable and c. e. Boolean algebras. Here the $\Pi^0_1$ class represents the set of prime ideals of a c. e. Boolean algebra.

Some of the results are known as part of the folklore of the subject. For more on computable Boolean algebras, see Remmel [159].

A computable Boolean algebra $B$ is given by a model $(\mathbb{N}, \preceq, \neg, \vee, \wedge)$ where $\preceq$ is a computable binary relation, $\neg$ is a computable unary operation, and $\vee$ and $\wedge$ are computable binary functions satisfying the usual properties of a Boolean algebra. In particular, there is a $\preceq$-least element 0 and a $\preceq$-greatest element 1, and we assume that $0 \in \mathbb{N}$ names the last and $1 \in \mathbb{N}$ names the greatest. We note that the *complement* $\neg a$ may be computed by searching for the element $b \in B$ such that $a \wedge b = 0$ and $a \vee b = 1$, and thus we do not need to assume that it is computable. The partial ordering $\preceq$ may be defined (and in fact computed) from the two binary operations in that $a \preceq b \iff a \vee b = b \iff a \wedge b = a$. (See the exercises.) We will also use the operation $a + b = (a \wedge \neg b) \vee (b \wedge \neg a)$, which will be computable for any computable Boolean algebra.

An element $a$ of a Boolean algebra $\mathcal{A}$ is an *atom* if there does not exists $b \in \mathcal{A}$ such that $0 < b < a$. $\mathcal{A}$ is said to be *atomless* if it has no atoms. Alternatively, we may say that $\mathcal{A}$ is *dense* if the ordering $\preceq$ is dense, that is, whenever $a < b$ in $\mathcal{A}$, then there exists $c \in \mathcal{A}$ such that $a < c < b$. $\mathcal{A}$ is said to be *atomic* if for every $b \in \mathcal{A}$, there exists an atom $a \in \mathcal{A}$ such that $a \preceq b$.

The fundamental computable atomless Boolean algebra $\mathcal{Q}$ may be thought of as the family of clopen subsets of $\{0,1\}^{\omega}$. Each clopen set has a unique representation as a finite union of disjoint intervals $I(\sigma_1) \cup \cdots \cup I(\sigma_k)$, where each $\sigma_i$ has the same length and $k$ is taken to be as small as possible. Then the join ($\vee$) and meet ($\wedge$) operations are clearly computable, as well as the complement operation and the partial ordering relation on $\mathcal{Q}$.

Alternatively, we may consider the fundamental Boolean algebra $\mathcal{Q}(\omega)$ to be the Lindenbaum algebra of propositional calculus over an infinite set $\{A_0, A_1, \dots\}$ of propositional variables. Here two propositions $p$ and $q$ are equal in $\mathcal{Q}(\omega)$ if they have the same truth table, so that this is a computable equivalence relation.

A c. e. Boolean algebra is given by a model $(\mathbb{N}, \preceq, \vee, \wedge)$ such that $\preceq$ is a c.e. relation which is a pre-ordering, $\vee, \wedge$ are total computable binary functions, and the quotient structure $B = (\mathbb{N}, \preceq, \vee, \wedge)/\equiv$ is a Boolean algebra (where $n \equiv m \iff n \preceq m \ \& \ m \preceq n$). We can suppose that $0 \in \mathbb{N}$ names the least and $1 \in \mathbb{N}$ the greatest element of $\mathcal{B}$. Note here that $\equiv$ is preserved under the operations.

A subset $I$ of a Boolean algebra $\mathcal{B}$ is said to be an *ideal* if for all $a, b \in B$,

**(i)** If $a \in I$ and $b \in I$, then $a \vee b \in I$;

**(ii)** If $b \in I$ and $a \leq^B b$, then $a \in I$.

An ideal $I$ is *proper* if $1 \notin I$ and is *prime* if, for all $a, b$: (iii) $a \vee b \in I \to a \in I$ or $b \in I$.

An ideal $I$ is *principal* if, for some $b \in \mathcal{B}$, $I = I(b) = \{a \in \mathcal{B} : a \leq b\}$.

Finally, an ideal $I$ is *maximal* if $I$ is proper and there is no proper ideal $J$ with $I \subset J$.

For any ideal $I$, the equivalence relation $\equiv^I$ is defined by

$$a \equiv^I b \iff a + b \in I.$$

It is clear that $\equiv^I$ is c. e. if $I$ is c. e. and is computable if $I$ is computable.

Conversely, given an operation-preserving equivalence relation $\equiv$ on $\mathcal{B}$, the corresponding ideal $I$ may be defined as $\{a : a \equiv 0\}$. Then $I$ will be computable (c. e. ) if $\equiv$ is computable (c. e. ).

The dual notion of an ideal is a *filter*. A subset $M$ of a Boolean algebra $\mathcal{B}$ is a filter if it is closed uner $\wedge$ and is closed upwards. It is easy to see that $M$ is a filter if and only if the set $M^d = \{\neg b : b \in M\}$ is an ideal; similarly for any ideal $I$, we may define the dual filter $I^d = \{\neg b : b \in I\}$. Downey [54] develops the theory of c. e. Boolean algebras from the point of view of c. e. filters.

Let us define a *computable quotient Boolean algebra* to be the quotient $\mathcal{B}/\equiv^B$, where $\mathcal{B} = (B, \equiv^B, \neg^B, \wedge^B, \vee^B)$ is a computable structure such that $B \subset \omega$, such that $\equiv^B$ is an equivalence relation on $B$, such that the unary operation $\neg^B$ and the two binary operations $\vee^B$ and $\wedge^B$ preserve the equivalence classes, and hence the set of equivalence classes forms a Boolean algebra.

**Lemma 8.1.1.** *Any computable quotient Boolean algebra $\mathcal{B}$ is isomorphic to a computable Boolean algebra $\mathcal{A}$.*

*Proof.* Define the universe $A$ of $\mathcal{A}$ by
$A = \{b \in B : (\forall a < b)\neg(a \equiv^B b)\}$.

For any $b \in B$, let $\psi(b)$ be the least $a$ such that $a \equiv^B b$. Then define the operations on $A$ by
$\neg^A(a) = \psi(\neg^B(a))$,
$a \vee^A b = \psi(a \vee^B b)$, and
$a \wedge^A b = \psi(a \wedge^B b)$.

It is clear that the set $A$ together with these operations forms a Boolean algebra which is isomorphic to the Boolean algebra on the equivalence classes of $\mathcal{B}$ and that the set $A$ and each of the Boolean operations is computable. $\square$

For any Boolean algebra $\mathcal{B}$ with universe $B = \omega$, let $P(\mathcal{B})$ be the class of maximal ideals $\mathcal{B}$. It is easy to see that $P(\mathcal{B})$ is a closed subclass of $2^\omega$, where an ideal $J$ is represented as by its characteristic function.

**Theorem 8.1.2.** *If $\mathcal{A}$ is a c. e. quotient Boolean algebra, then $P(\mathcal{A})$ is a $\Pi^0_1$ class and if $\mathcal{A}$ is a computable Boolean algebra, then $P(\mathcal{A})$ is a decidable $\Pi^0_1$ class.*

*Proof.* Suppose that $\mathcal{A} = \mathcal{B}/\equiv$ is a c. e. quotient Boolean algebra. We can represent the class $P(\mathcal{A})$ of prime ideals on $\mathcal{A}$ as follows.

$x \in P(\mathcal{A}) \iff$

(1) $(\forall a)(\forall b)[a \equiv b \rightarrow x(a) = x(b)]$ and

(2) $(\forall a)(\forall b)[x(a) = x(b) = 1 \rightarrow x(a \vee^B b) = 1]$ and

(3) $(\forall a)(\forall b)[x(a) = 1 \rightarrow x(a \wedge^B b) = 1]$ and

(4) $(\forall a)[x(a) = 1 \iff x(\neg^B a) = 0]$.

This clearly defines a $\Pi^0_1$ class. Observe that either $x(0^B) = 1$ or $x(1^B) = 1$ by (4) and hence $x(0^B) = 1$ by (3), so that $x(1^B) = 0$. Thus any $x \in P(\mathcal{A})$ represents a proper prime ideal. If $\mathcal{B}$ is actually a computable Boolean algebra, then we can omit clause (1) and define a computable tree $T$ with no dead ends such that $P(\mathcal{B}) = [T]$, as follows. $T$ is defined to be the set of finite sequences $x = (x(0), \ldots, x(n-1))$ which satisfy the following, where $lh(x) = n$.

(2)' $(\forall a < n)(\forall b < n)[(x(a) = x(b) = 1 a \vee^B b < n) \rightarrow x(a \vee^B b) = 1]$ and

(3)' $(\forall a < n)(\forall b < n)[(x(a) = 1 a \wedge^B b < n) \rightarrow x(a \wedge^B b) = 1]$ and

(4)' $(\forall a < n)(foralli < 2)[(x(a) = i \neg^B a < n) \rightarrow x(\neg^B a) = 1 - i]$.

$(5)'_k$ $(\forall a_1 < a_2 < \cdots < a_k < n)(x(a_1) = \cdots = x(a_k) = 0 \rightarrow a_1 \wedge^B a_2 \cdots \wedge^B a_k \neq 1^B)$.

Clause (5) is needed to establish the finite intersection property for $\{a < n : x(a) = 0\}$ which will ensure that any $\sigma \in T$ can be extended to a prime ideal in $P(\mathcal{A})$. This then implies that $T$ has no dead ends. □

We can now apply our general results about $\Pi^0_1$ classes to Boolean algebras. The following is a consequence of Theorems 2.2.15 and 3.1.4.

**Theorem 8.1.3.** *(i) For any c. e. Boolean algebra $\mathcal{B}$, $\mathcal{B}$ has a prime ideal $J$ of low c. e. degree (so that $J$ is computable in $\mathbf{0}'$).*

*(ii) For any computable Boolean algebra $\mathcal{B}$, $\mathcal{B}$ has a computable prime ideal.*

**Theorem 8.1.4.** *For any c. e. Boolean algebra $\mathcal{B}$ with no computable prime ideal, there exists a c. e. degree* $\mathbf{a}$ *such that $\mathcal{B}$ has no prime ideals of degree* $\leq \mathbf{a}$.

*Proof.* This follows from Theorem 3.thm:nla. □

**Theorem 8.1.5.** *For any For any c. e. Boolean algebra $\mathcal{B}$ with no computable prime ideals, there exists two prime ideals, $I$ and $J$, of $\mathcal{B}$ such that any set computable from $I$ and computable from $J$ is in fact computable.*

*Proof.* This follows from Theorem 3.3.2.12. □

The following theorem is a corollary of Theorems 4.4.2.3,4.4.4 and 4.2.2.

**Theorem 8.1.6.** *Let $\mathcal{B}$ be a c. e. Boolean algebra only countably many prime ideals. Then*

*(a) $\mathcal{B}$ has a computable prime ideal.*

*(b) If $\mathcal{B}$ has only finitely many prime ideals, then every prime ideal is computable.*

*(c) Every prime ideal of $\mathcal{B}$ is hyperarithmetic.*

Now we will briefly consider the notion of rank for ideals. It is an exercise below 5 that for any $\Pi^0_1$ class $P$, an element $U$ of $RC(P)$ is an atom if and only if $U \cap P$ is a singleton.

**Proposition 8.1.7.** *For any prime ideal $J$ of a c. e. Boolean algebra $\mathcal{B}$, $J$ is isolated in $P(\mathcal{B})$ if and only if $J$ is principal.*

*Proof.* Suppose that $J$ is a principal and prime ideal. Then for some $b$, we have $J = \{a : b \leq a\}$. It follows that $J$ is isolated in the interval $I(J \upharpoonright b+1)$.

Suppose that $J$ is isolated in the interval $I(\sigma)$ where $lh(\sigma) = n$. Let $b_\sigma = b_0 \wedge b_1 \wedge \ldots b_{n-1}$ where $b_i = i$ if $x(i) = 1$ and $b_i = \neg i$ if $x(i) = 0$. The isolation means that $J$ is the only prime ideal of $\mathcal{B}$ that contains $b_\sigma$. It follows that $J$ is generated by $b_\sigma$. $\square$

Note that a principal ideal is prime if and only if it is generated by an atom.

The following is a corollary of Theorem 4.4.4.3.

**Theorem 8.1.8.** *Let $\mathcal{B}$ be a c. e. Boolean algebra which has a unique non-principal prime ideal $J$. Then $J \leq_T \mathbf{0}''$ and if $\mathcal{B}$ is computable, then $J \leq_T \mathbf{0}'$.*

Next we consider the reverse direction of the correspondence between $\Pi^0_1$ classes and c. e. quotient Boolean algebras.

For an arbitrary $\Pi^0_1$ class $P$, let $RC(P)$ be the Boolean algebra of relatively clopen subsets of $P$, that is, $\{U \cap P : U \in \mathcal{Q}\}$ under the standard set operations. Let $\mathcal{B}(P)$ denote the c. e. Boolean algebra resulting from $\mathcal{Q}$ by taking the equivalence relation

$$U \equiv^P V \iff U \cap P = V \cap P.$$

(Note that the corresponding ideal $I(P) = \{U \in \mathcal{Q}(\{0,1\}^{\mathbb{N}}) : U \cap P = \emptyset\}$ is in fact a c. e. ideal and is computable if and only if $P$ is decidable.)

The notion of a perfect closed set corresponds to the notion of an atomless Boolean algebra in the following sense.

**Proposition 8.1.9.** *For any closed set $P$, $P$ is perfect if and only if $RC(P)$ is atomless.*

*Proof.* Assume first that $P$ is perfect and let $U \cap P \neq \emptyset$. Then $U \cap P$ contains at least two elements and hence $U$ can be partitioned into two clopen sets $U_1$ and $U_2$ such that $U_1 \cap P$ and $U_2 \cap P$ are distinct nonempty sets in $RC(P)$. It follows that $RC(P)$ is atomless.

Next assume that $P$ is not perfect and let $x$ be isolated in $P$. This means that there is a clopen set $U$ such that $U \cap P = \{x\}$. Clearly $U \cap P$ is an atom in $RC(P)$. $\square$

**Theorem 8.1.10.** *Let $P \subseteq \{0,1\}^{\mathbb{N}}$ be a $\Pi^0_1$ class. Then the quotient algebra $\mathcal{B}(P)$ is isomorphic to the Boolean algebra $RC(P)$, the equivalence relation $\equiv_P$ is computably enumerable and hence $RC(P)$ is a c. e. Boolean algebra. Furthermore, if $P$ is decidable, then $\equiv^P$ is computable and $\mathcal{B}(P$ is a computable Boolean algebra.*

*Proof.* The isomorphism mapping $U/equiv^P$ to $U \cap P$ is clearly a computable isomorphism from $\mathcal{B}(P)$ to $RC(P)$.

To see that $\equiv^P$ is a c. e. relation, let $P = [T]$ where $T$ is a computable tree and suppose that $U = I(\sigma_1) \cup \dots I(\sigma_k)$ and $V = I(\tau_1) \cup \dots I(\tau_m)$. Then

$$U \cap P \subseteq V \cap P \iff (\forall i \leq k) I(\sigma_i) \cap P \subseteq V \cap P.$$

But for any $\sigma$, we have

$$I(\sigma) \cap P \subset V \cap P \iff (\exists n)(\forall \tau)[(lh(\tau) = n \,\& \sigma \prec \tau \,\&\, \tau \in T) \rightarrow (\exists i \leq m)(\tau_i \prec \tau)].$$

Finally, $\equiv^P$ is c. e. , since
$U \equiv^P V \iff [U \cap P) \subset V \cap P V \cap P \subset U \cap P]$.

If $P$ is decidable, then $T$ has no dead ends, so we can take $k$ to be the maximum of $\{lh(\tau_i) : i \leq m\}$, so that $\equiv^P$ will be computable and hence each operation of $\mathcal{B}(P)$ also computable. $\square$

**Theorem 8.1.11.** *(a) For any $\Pi^0_1$ class $P \subseteq \{0,1\}^{\mathbb{N}}$, $P$ is computably homeomorphic to the set of prime ideals of $RC(P)$.*

*(b) For any Boolean algebra $\mathcal{B}$ with universe $B = \omega$, $RC(P(\mathcal{B}))$ is isomorphic to $\mathcal{B}$. For a c. e. Boolean algebra, this isomorphism is effective.*

*Proof.* (a) Map the element $x$ of $P$ to the prime ideal $J(x) = \{U \cap P : x \notin U\}$. If $x \neq y$, then there must be a clopen set $P$ such that $x \in U$ and $y \notin U$, so that the map is injective. Given a prime ideal $J$ of $RC(P)$, we claim that there must be a unique element $x_J$ of $P$ which belongs to every $U \cap P$ in $J$. Every finite subset of $J$ has nonempty intersection since $\emptyset \notin J$, hence by compactness $\bigcap J$ is nonempty. Let $x$ be an element of $\bigcap J$, suppose that $y \neq x$ and let $U$ be a clopen set such that $x \in U$ and $y \notin U$. Then $U \cap P \in J$ but $y \notin U$ and hence $y \notin \bigcap J$. Thus $\bigcap J$ is a singleton. We leave it to the exercises to show that $x_J$ may be computed effectively from $J$.

(b) Let $\mathcal{B}$ be a Boolean algebra with universe $B = \omega$. The isomorphism from $\mathcal{B}$ to $RC(P(\mathcal{B}))$ is given by mapping the element $b$ to $\{J :\ J$ is a prime ideal of $\mathcal{B} \,\&\, b \notin J\}$, that is, to $P(\mathcal{B}) \cap U(b)$, where $U(b)$ is the clopen set defined by $x \in U(b) \iff x(b) = 0$. $\square$

We now have the following corollaries.

**Theorem 8.1.12.** *For any c. e. degree* $\mathbf{c}$*, there is a c. e. Boolean algebra* $\mathcal{B}$ *such that the c. e. degrees of prime ideals of* $\mathcal{B}$ *are exactly the c. e. degrees above* $\mathbf{c}$.

*Proof.* This is an immediate consequence of Theorems 3.3.2.4 and 8.1.10. ☐

**Theorem 8.1.13.** *There is a c. e. Boolean algebra* $\mathcal{B}$ *such that any two prime ideals of* $\Gamma$ *are Turing incomparable.*

*Proof.* This follows from Theorem 3.3.2.10. ☐

Next we consider the translation of the notion of a thin $\Pi^0_1$ class to the corresponding notion for Boolean algebras. Let us say that a c. e. Boolean algebra is *thin* if every c. e. ideal of $\mathcal{B}$ is principal. Downey [54] defined a c. e. filter $M$ in $\mathcal{Q}$ to be *superthick* if, for every c. e. filter $W$ such that $M \subset W \subset \mathcal{Q}$, there exists $b \in \mathcal{Q}$ such that $W = \langle M, b\rangle$. Here $a \in \langle M, b\rangle$ if and only if there exists $x \in M$ such that $b \wedge x \leq a$.

**Lemma 8.1.14.** *Let* $\mathcal{A}$ *be the c. e. Boolean algebra defined as the quotient of* $\mathcal{Q}$ *modulo the ideal* $I$*. Then the following are equivalent.*

*(i) The* $\Pi^0_1$ *class* $P(\mathcal{A})$ *is thin.*

*(ii) The filter* $I^d$ *is superthick.*

*(iii)* $\mathcal{A}$ *is thin.*

*Proof.* We will show that (i) and (iii) are equivalent and leave the rest as an exercise. We may assume by Theorem 8.1.10 that $P \subseteq \{0,1\}^{\mathbb{N}}$ is a $\Pi^0_1$ class, that $I = \{V \in \mathcal{Q} : V \cap P = \emptyset\}$ is a c. e. ideal in $\mathcal{Q}$ and that $\mathcal{A} = \{[U] : U \in \mathcal{Q}\}$, where $[U]$ is the equivalence class in $\mathcal{Q}$ of $U$ under the equivalence relation defined by $U \equiv_P V \iff U \cap P = V \cap P$. Then we have, for any $x \in \{0,1\}^{\mathbb{N}}$,

$$x \in P \iff (\forall U \in I) x \notin U.$$

Suppose first that $P$ is thin and let $J$ be a c. e. ideal in $\mathcal{A}$. Define the $\Pi^0_1$ class $Q \subset P$ by

$$x \in Q \iff (\forall V \in \mathcal{Q})([V] \in J \implies x \notin V).$$

By assumption, $Q = P \cap U$ for some $U \in \mathcal{Q}$ and it follows that $J = \{[V] : [V] \subseteq [U]\}$ and is principal.

For the converse, suppose that $\mathcal{A}$ is thin and let $Q \subset P$ be a $\Pi^0_1$ class. Then $J = \{[V] : V \cap Q = \emptyset\}$ is a c. e. ideal in $\mathcal{A}$. By assumption, there exists $U$ such that, for all $V \in \mathcal{Q}$, $[V] \in J \iff [V] \subseteq [U]$. It is then easy to see that $Q = P \cap \overline{U}$. ☐

Here is an existence result concerning $\Pi^0_1$ classes of rank one. This follows from Theorem 4.4.4.5 and Corollary 4.4.5.2.

**Theorem 8.1.15.** *(a) For any degree* $\mathbf{b} \leq 0'$*, there is a computable Boolean algebra* $\mathcal{B}$ *with a unique non-principal prime ideal* $J$ *such that* $J$ *has degree* $\mathbf{b}$*.*

*(b) For any degree* $\mathbf{b}$ *such that* $\mathbf{0}' \leq \mathbf{b} \leq \mathbf{0}'$*, there is a c. e. Boolean algebra* $\mathcal{B}$ *with unique non-principal prime ideal* $J$ *of degree* $\mathbf{b}$*.*

**Exercises**

8.1.1. Show how to compute the partial ordering $\leq^B$ of a Boolean algebra $B$ from the $\vee^B$ and $\wedge^B$ operations.

8.1.2. Show that the $\approx$ relation in a c. e. Boolean algebra is preserved under the operations. That is, if $a \approx b$ then $\neg a \approx \neg b$ and similarly for the binary operations.

8.1.3. Show how to carefully define the Boolean algebra of clopen sets to see that it is in fact computable.

8.1.4. Show how to compute $x_J$ in the proof of Theorem 8.1.11. (Hint: for any clopen $U$, exactly one of $U$ and $\{0,1\}^{\mathbb{N}} - U$ belongs to $J$; to find $x(0)$, check which one of $I((0))$ and $I((1))$ belongs to $J$.)

8.1.5. Show that for any closed set $P$, $U$ is an atom in the Boolean algebra $RC(P)$ if and only if $U \cap P$ is a singleton.

8.1.6. Complete the proof of Lemma 8.1.14.

## 8.2 Groups and Rings

In this section, we consider Abelian groups and commutative rings with unity.

**Theorem 8.2.1.** *(a) For any c. e. commutative ring* $R$ *with unity, the set of all ideals of* $R$ *and the set of prime ideals of* $R$ *can be represented by c. b.* $\Pi^0_1$ *classes and the set of maximal ideals of* $R$ *can be represented by a* $\Pi^0_2$ *class.*

*(b) For any c. e. group* $G$*, the set of all subgroups of* $G$ *can be represented by a c. b.* $\Pi^0_1$ *class and the set of maximal subgroups of* $G$ *can be represented by a* $\Pi^0_2$ *class.*

*Proof.* (a): Let $A$ be the underlying recursive ring and $I$ the c. e. ideal such that $R$ is the quotient $A/I$. Then there is an effective one-to-one correspondence between the ideals of $R$ and the ideals of $A$ which extends $I$, defined as follows. For any ideal $J$ of $A$ which extends $I$, let $J_R = \{[a] \in R : a \in J\}$. Similarly, given an ideal $J_R$ of $R$, let $J = \{a \in A : [a] \in J_R\}$. Since $0 \in J$, it follows that $I \subset J$. It is easy to see that $J_R$ is prime if and only if $J$ is prime and is maximal if and only if $J$ is maximal. Thus we will actually consider the $\Pi^0_1$ class $Prime(A, I)$ of prime ideals of $A$ extending $I$. Since $A$ is recursive, we may assume that the universe of $A$ is $\omega$. Let the operations of $A$ be denoted by $+^A$ and $\cdot^A$ and assume that the additive identity $0^A = 0$ and the unity $1^A = 1$. Let $I^s$ be the set of elements of $I$ which have been enumerated into $I$ by stage $s$. Then the recursive tree $T$ is defined so that for any $\sigma = (x(0), \dots, x(n-1)) \in \{0,1\}^n$, $\sigma$ is in $T$ if and only if the following conditions all hold.

**(i)** For any $i, j, k < n$, if $i +^A j = k$, $x(i) = 1$ and $x(j) = 1$, then $x(k) = 1$.

**(ii)** For any $i, j, k < n$, if $i \cdot^A j = k$ and $x(i) = 1$, then $x(k) = 1$.

**(iii)** If $n > 1$, then $x(1) = 0$.

**(iv)** For any $i, j, k < n$, if $i \cdot^A j = k$ and $x(k) = 1$, then $x(i) = 1$ or $x(j) = 1$.

**(v)** For any $i < n$, if $i \in I^n$, then $x(i) = 1$.

Conditions (i),(ii) and (iii) ensure that any infinite path through $T$ will represent an ideal of $R$. Condition (iv) ensures that any infinite path through $T$ will represent a prime ideal. Condition (v) ensures that any infinite path through $T$ will represent an extension of $I$. Note that we can modify this construction to define the class of all ideals of $A$ which extend $I$ by simply omitting condition (iv).

To define the class of maximal ideals of $A$ which extend $I$, we note that any maximal ideal is prime and that an ideal $J$ is maximal in $A$ if and only if, for each $r \in A \setminus J$, the ideal $J(r)$ generated by $J \cup \{r\}$ equals $A$, which is if and only if $1 \in J(r)$, and we also note that $J(r) = \{i +^A r \cdot^A s : i \in I, s \in A\}$. Thus we let $P$ be the $\Pi^0_1$ class representing the set of prime ideals of $A$ extending $I$ and define $Q$ with $Q \subseteq P$ by

$$x \in Q \iff x \in P \;\&\; (\forall j)[x(j) = 0 \to (\exists i, r)(x(i) = 1 \;\&\; 1 = i +^A rj)].$$

Thus the set of all maximal ideals of $A$ extending $I$ is represented by the $\Pi^0_2$ class $Q$.

(b) The class representing all subgroups of the group $G = (\omega, +^G, 0, -^G)$ is defined as the set of all $x$ such that all of the following hold.

**(i)** For any $i, j, k < n$, if $i +^G j = k$ and $x(i) = x(j) = 1$, then $x(k) = 1$ and

**(ii)** For any $i, j, k < n$, if $i -^G j = k$ and $x(i) = x(j) = 1$, then $x(k) = 1$ and

**(iii)** $x(0) = 1$.

For the maximal subgroups, we note that $H$ is a maximal subgroup of $G$ if and only if, for each $g \in G \setminus H$, the subgroup $H(g)$ generated by $H \cup \{g\}$ equals $G$ and we also note that $H(g) = \{h +^G z \cdot g : h \in H, i \in \mathbb{Z}\}$. Thus we let $P$ be the $\Pi^0_1$ class representing the set of subgroups of $G$ and define $Q \subseteq P$ by $x \in Q$ if and only if

$$x \in P \;\&\; (\forall a, j)[x(j) = 0 \to (\exists i)(\exists z \in \mathbb{Z})(x(i) = 1 \;\&\; a = i +^G z \cdot j)].$$

Thus the set of all maximal subgroups of $G$ is represented by the $\Pi^0_2$ class $Q$. $\square$

We can now apply our general results about $\Pi^0_1$ classes to groups and rings.

**Theorem 8.2.2.** *(a) Any c. e. Abelian group which has a proper subgroup has a proper subgroup of c. e. degree.*

*(b) Any c. e. commutative ring with unity $R$ which is not a field has a prime ideal of c. e. degree.*

*(c) If a c. e. Abelian group $G$ has a maximal subgroup, then it has a maximal subgroup computable in some $\Sigma^1_1$ set.*

*(d) If a c. e. commutative ring $R$ with unity has a maximal ideal, then $R$ has a maximal ideal computable in some $\Sigma^1_1$ set.*

**Theorem 8.2.3.** *For any c. e. commutative ring $R$ with unity which has no computable prime ideal,*

1. *there exists a c. e. degree $\mathbf{a}$ such that $\mathcal{B}$ has no prime ideals of degree $\leq \mathbf{a}$.*
2. *there exists two prime ideals, $I$ and $J$, of $\mathcal{B}$ such that any set computable from $I$ and computable from $J$ is computable.*

The following theorem is a corollary of Theorems 4.4.2.3,4.4.4 and 4.2.2.

**Theorem 8.2.4.** *Let $R$ be a c. e. commutative ring with unity which has only countably many prime ideals. Then*

*(a) $R$ has a computable prime ideal.*

*(b) If $R$ has only finitely many prime ideals, then every prime ideal is computable.*

*(c) Every prime ideal of $R$ is hyperarithmetic.*

Next we turn to the other direction of our correspondences, that is, representing an arbitrary $\Pi^0_1$ class by the set of solutions to one of our problems. We say that a class $Q$ *weakly represents* the class $P$ if there is a computable functional $\phi$ such that for each $x \in Q$, $\phi(x) \in P$ and $\phi(x) \leq_T x$ and there is a computable functional $\psi$ such that for all $y \in Q$, $\psi(y) \in P$ and $\psi(y) \equiv_T y$.

**Theorem 8.2.5.** *For any pair of disjoint c. e. sets, the r.b. $\Pi^0_1$ class $S(A, B)$ can be weakly represented by the set of prime ideals and by the set of maximal ideals of some c. e. commutative ring $R$ with identity.*

*Proof.* We give the proof of Friedman-Simpson-Smith [67]. Let the infinite disjoint c. e. sets $A, B$ be given. The construction begins with the underlying ring $R = \mathbf{Q}[x_n : n \in \omega]$ (the ring of polynomials with rational coefficients in infinitely many variables). Let $A = \{f(n) : n \in \omega\}$ and $B = \{g(n) : n \in \omega\}$ and let $I$ be the ideal generated by $\{x^{n+1}_{f(n)}, x^{n+1}_{g(n)} - 1, n \in \omega\}$. We claim that

**(1)** $I$ is a proper recursive ideal;

**(2)** the set of prime ideals of $R/I$ represents $S(A, B)$; and

**(3)** the set of maximal ideals of $R/I$ represents $S(A, B)$.

To test whether a given $f \in R$ is in $I$, we first produce $f^* = f(\bmod I)$ by repeating the following reduction procedure. For any factor $x_m^{k+1}$ occurring in a term of $f$, enumerate the finite sets $F = \{f(0), \ldots, f(k)\}$ and $G = \{g(0), \ldots, g(k)\}$ and determine whether $m \in F \cup G$. If $m = f(i) \in F$, then replace $x_m^{k+1}$ with 0 (since $x_m^{i+1} \in I$); if $m = g(i) \in G$, then replace $x_m^{k+1}$ with $x^{k-i}$ (since $x_m^{i+1} - 1 \in I$). Each step in this process reduces the degree of some term and thus the process must terminate in $f^*$ after a finite number of steps. Then $f \in I$ if and only if this $f^* = 0$. It follows that $1 \notin I$, so that $I$ is proper.

For any set $C \in S(A, B)$, let $M_C$ be the ideal generated by the set of all $\{x_m : m \in C\} \cup \{x_n - 1 : x \notin C\}$. It is easy to see that, using the reduction procedure described in the previous paragraph, any polynomial will be equivalent to some $q \in \mathbf{Q}$. Thus $M_C$ is a maximal ideal and $M_C \equiv_T C$.

Now an ideal $J$ is said to be *radical* if $a^n \in I$ for any $n$ implies that $a \in I$. It is clear that any prime ideal is radical and it is easy to check that in fact any maximal ideal is radical. Suppose that $J$ is a radical ideal of $R$ which extends $I$. Then it follows that $x_{f(n)} \in J$ and that $x_{g(n)} \notin J$ for all $n$. Thus we can define the weak representation of $S(A, B)$ by a class of ideals, simply letting $\phi(J) = C$, where $m \in C \iff x_m \in J$. Clearly $C \leq_T J$ in this case. $\square$

.

The representation Theorem 8.2.5 has, as usual, a number of immediate corollaries.

**Theorem 8.2.6.** *(1) There is a c. e. commutative ring $R$ with unity which has a prime ideal but has no computable prime ideal.*

*(2) There is a c. e. commutative ring $R$ with unity such that that any two prime ideals of $R$ are Turing incomparable.*

*(3) If $\mathbf{a}$ is a Turing degree and $\mathbf{0} <_T \mathbf{a} \leq_T \mathbf{0}'$, then there is a computable society $s$ which has a prime ideal of degree $\mathbf{a}$ but has no computable prime ideal.*

## 8.3 Index sets for computable algebra

In this section, we examine the logical complexity of various properties of algebraic structures. For example, we consider the property that a c. e. Boolean algebra is atomless and the property that a computable commutative ring with unity possesses a computable prime ideal.

### 8.3.1 Index sets for Boolean algebras

In this section, we define index sets for c. e. Boolean algebras and consider the complexity of properties associated with Boolean algebras and their ideals.

Recall the computable Boolean algebra $\mathcal{Q}$ of clopen subsets of $\{0,1\}^{\mathbb{N}}$. It follows from Theorem 8.1.11 that every c. e. Boolean algebra is a c. e. quotient of $\mathcal{Q}$. The $e$'th ideal $J_e$ of $\mathcal{Q}$ may be defined as follows, as the ideal generated by the set $W_e$.

$$J_e = \{a \in \mathcal{Q} : (\exists x_1, \dots, x_k \in W_e) a \leq x_1 \ \vee x_2 \vee \cdots \vee x_k\}.$$

Then the $e$'th c. e. Boolean algebra $\mathcal{B}_e$ may be defined as the quotient Boolean algebra $\mathcal{Q}/J_e$. Note that the corresponding equivalence relation $\equiv_e$ is also c. e. .

Theorem 8.1.2 may be uniformized as follows.

**Lemma 8.3.1.** *There is a primitive recursive function $f$ such that, for all $e$, $P_{f(e)}$ represents the set of maximal ideals of the c. e. Boolean algebra $\mathcal{B}_e$. Furthermore, if $\mathcal{B}_e$ is a computable Boolean algebra, then $\P_{f(e)}$ is a decidable $\Pi^0_1$ class.*

For the reverse direction, Theorems 8.1.10 and 8.1.11 may be uniformized as follows.

**Lemma 8.3.2.** *There is a primitive recursive function $g$ such that, for all $e$, $P_e$ represents the set of maximal ideals of $\mathcal{B}_{g(e)}$. Furthermore, if $P_e$ is a decidable $\Pi^0_1$ class, then $\mathcal{B}_{g(e)}$ is a computable Boolean algebra.*

*Proof.* Given the $\Pi^0_1$ class $P_e$, we simply compute the c. e. set $J_{g(e)} = W_{g(e)} = \{U : U \cap P = \emptyset\}$ and then $P_e$ will represent the class of maximal ideals of the Boolean algebra $\mathcal{B}_{g(e)}$. $\square$

We can now apply the index set result sets of this chapter 5 to obtain some complexity results for maximal ideals of Boolean algebras.

In a Boolean algebra $\mathcal{B}$, we require that $0 \neq 1$, but if we take the quotient of $\mathcal{B}$ modulo itself, then we obtain the *trivial* structure with one element, so that in fact $0 = 1$. The proof of the following proposition is left as an exerecise.

**Proposition 8.3.3.** *$\{e : \mathcal{B}_e$ is trivial$\}$ is $\Pi^0_2$ complete.*

We can also classify the index sets of theories with a given number of complete consistent extensions (and similarly for computable complete consistent extensions). The next theorem follows from Theorems 5.2.8, 5.2.13 and 5.2.15 as above. Let us abbreviate "computable consistent extensions" by CCEs.

**Theorem 8.3.4.** *Let $c > 0$ be finite.*

1. *$(\{e : \mathcal{B}_e$ has $> c$ prime ideals$\}, \{e : \mathcal{B}_e$ has $\leq c$ prime ideals$\}$ is $(\Sigma^0_2, \Pi^0_2)$ complete.*

2. *$\{e : \mathcal{B}_e$ has exactly $c$ prime ideals$\}$ is $D^0_2$ complete.*

3. *$(\{e : \mathcal{B}_e$ has finitely many prime ideals$\}, \{e : \mathcal{B}_e$ has infinitely many prime ideals$\}$ is $(\Sigma^0_3, \Pi^0_3)$ complete.*

4. $\{e : \mathcal{B}_e$ *has exactly* $\aleph_0$ *prime ideals*$\}$ *is* $\Pi^1_1$ *complete.*

5. $(\{e : \mathcal{B}_e$ *has uncountably many prime ideals*$\}, \{e : \mathcal{B}_e$ *has countably many prime ideals*$\}$ *is* $(\Sigma^1_1, \Pi^1_1)$ *complete.*

For a given number of computable complete consistent extensions, we apply Theorems 5.3.5 and 5.3.9.

**Theorem 8.3.5.** *Let* $c > 0$ *be finite.*

1. $(\{e : \mathcal{B}_e$ *has* $> c$ *computable prime ideals*$\}, \{e : \mathcal{B}_e$ *has* $\leq c$ *computable prime ideals*$\}$ *is* $(\Sigma^0_3, \Pi^0_3)$ *complete.*

2. $\{e : \mathcal{B}_e$ *has exactly* $c$ *computable prime ideals*$\}$ *is* $D^0_3$ *complete.*

3. $(\{e : \mathcal{B}_e$ *has* $< \aleph_0$ *computable prime ideals*$\}, \{e : \mathcal{B}_e$ *has* $\geq \aleph_0$ *computable prime ideals*$\})$ *is* $(\Sigma^0_4, \Pi^0_4)$ *complete.*

The following result can be obtained by combining Proposition 8.1.9 with Theorem 5.6.1.

**Proposition 8.3.6.** *i*$\{e : \mathcal{B}_e$ *is atomless*$\}$ *is* $\Pi^0_3$ *complete.*

Finally, we consider thin Boolean algebras.

**Proposition 8.3.7.** $\{e : \mathcal{B}_e$ *is thin*$\}$ *is a* $\Pi^0_4$ *complete set.*

**Exercises**

8.3.1. Prove Lemma 8.3.3

8.3.2. Give details of the proof of Lemma 8.3.1.

8.3.3. Give details of the proof of Lemma 8.3.2.

8.3.4. A Boolean algebra $\mathcal{B}$ is said to be *atomic* if, for every $b \in \mathcal{B}$, there is an atom $a \leq b$. Show that i$\{e : \mathcal{B}_e$ is atomic$\}$ is $\Pi^0_4$ complete.

8.3.5. Let $R = \mathcal{Q}[x_n : n \in \mathbb{N}]$ and enumerate the elements of $R$ as $\{r_0, r_1, \dots\}$. Define the $e$'th c. e. ideal $J_e$ as the ideal generated by the set $\{r_i : i \in W_e\}$ and let $R_e$ be the quotient $R/J_e$. Show that $\{e : R_e$ has a computable prime ideal$\}$ is a $\Sigma^0_3$ set.

## 8.4 Reverse mathematics and computable algebra

In this section, we consider the proof-theoretic strength of the existence of prime ideals in Boolean algebras and, more generally, in commutative rings with unity.

It might seem that the close connection between countable Boolean algebras and propositional theories would carry over into the proof-theoretic content.

However, it is easy to see that any computable Boolean algebra has a computable maximal ideal, whereas we have seen that computably axiomatiable theories may not have computable complete consistent extensions. Thus c. e. Boolean algebras are different from computable Boolean algebras.

**Proposition 8.4.1.** *The statement 'Every countable Boolean algebra has a maximal (proper) ideal' is provable in* $RCA_0$.

*Proof.* Given a countable Boolean algebra $\mathcal{B} = \{b_0, b_1, \dots\}$, define the maximal ideal $I$ in stages $I_n$ as follows. Let $a_n$ be the join of the finite set $I_n$. We assume that $b_0 = 0$ and begin with $I_0 = \{0\}$. At each subsequent stage $n+1$, put $b_{n+1}$ into $I_{n+1}$ as long as $b_{n+1} \vee a_n \neq 1$. □

For countable commutative rings with unity, Friedman, Simpson and Smith [66] showed that the existence of prime ideals is equivalent with Weak König's Lemma.

**Theorem 8.4.2** (Friedman, Simpson, Smith)**.** *Weak König's Lemma is equivalent to the statement that every countable commutative ring with unity has a prime ideal*

*Proof.* Given a countable commutative ring $R$ with unity, the set of prime ideals of $R$ and the set of prime ideals of $R$ can be represented as the set of infinite paths through an infinite binary tree $T$. It then follows from Theorem 6.1.8 that $T$ has an infinite path and therefore $R$ has a prime ideal.

For the other direction, we follow the proof from the Addendum [68] to [66]. Let $R_0 = \mathcal{Q}[x_n : n \in \mathbb{N}\}$ be the polynomial ring over the rationals with countably many variables. We will use the last equivalent form of Weak König's Lemma from Theorem 6.1.8. Let $f, g : \mathbb{N} \to \mathbb{N}$ be one-to-one functions such that $f(i) \neq g(j)$ for any $i, j$. Then let $I$ be the ideal generated by $\{x_{f(n)}, x_{g(n)} - 1 : n \in \mathbb{N}\}$ and let $R = R_0/I$. By assumption, $R$ has a prime ideal $J$. Now let $J_0 = \{p \in R_0 : p + I \in J\}$ and let $X = \{m : x_m \in J_0\}$ to demonstrate Weak König's Lemma. □

# Chapter 9

# Computer Science

Non-monotonic logics arose in attempts to formalize several notions of "common-sense" reasoning. These systems include the default logic of Reiter [158] and the stable semantics of general logic programs [73] due to Gelfond and Lifschitz. Classical logic is monotonic in that a deduction from a set of premises remains valid for any larger set of premises. Minsky [140] suggested that there is another form of reasoning which is not monotonic. That is, common sense and even scientific reasoning forces one to make assumptions in the absence of complete information. Thus new information may naturally lead to the rejection of previous beliefs. The set of stable models of a logic program is in some sense a non-monotonic generalization of the set of complete consistent extensions of a set of premises. Marek, Nerode and Remmel [124] showed that different versions of a logic program may be used to represent c. b., bounded and unbounded $\Pi^0_1$ classes.

Another area of theoretical computer science where $\Pi^0_1$ classes have application is the study of $\omega$-languages, that is, sets of infinite words. Here an $\omega$-language is the set of infinite words which are *accepted*, in some fashion, by a program. In particular, a $\Pi^0_1$ class may be viewed as the set of infinite words which are accepted by a deterministic automata $M$ in the sense that an infinite sequence $x = (x(0), x(1), \dots$ is accepted by $M$ if $M$ is always in an accepting state after reading each initial segment $(x(0), \dots, x(k))$ of $x$. L. Staiger and K. Wagner [190, 187, 188, 189] have examined several other widely studied notions of acceptance which produce different classes. The relation between these notions and $\Pi^0_1$ classes is developed in [39].

## 9.1 Non-monotonic Logic

In this section, we shall show how $\Pi^0_1$ classes arise naturally in the setting of nonmonotonic logics. In fact, nonmonotonic logic is one of the few areas where all three types of $\Pi^0_1$ classes, arbitrary, bounded, and computably bounded, arise in a natural manner. Nonmonotonic logics arose in attempts to formalize several forms of common sense reasoning. These systems include the default logic of Reiter [158], the nonmonotonic modal logics of McDermott and Doyle [135, 134], the stable semantics of general logic programs [73], and the answer sets semantics for logic programs with classical negation [74].

Classical logic, which we considered in the previous section, is monotonic in that a deduction from a set of premises remains valid for any larger set of premises. Minsky [140] suggested that there is another sort of reasoning which is not monotonic. This is the reasoning in which we deduce a statement based on the absence of any evidence against the statement. Such statements are in the category of beliefs rather than in the category of truths. Common sense or even scientific reasoning forces one to make assumptions without complete information. New information may naturally lead to the rejection of previous beliefs.

Tarski [191] characterized monotonic formal systems by means of monotonic rules of inference. Such systems include intuitionistic logic, classical logics, modal logics, and many others. Suppose that a nonempty set $U$ is given. In a particular application $U$ may be the collection of all formulas of a propositional or first order logic, of all legal strings of a formal system, or of all atomic statements as in logic programming.

A **monotonic rule of inference** is a tuple $r = (P, \varphi)$ where $P = (\alpha_1, \ldots, \alpha_n)$ is a finite (possibly empty) list of objects from $U$ and $\varphi$ is an element of $U$. Such a rule $r$ is usually written in the suggestive form

$$r = \frac{\alpha_1, \ldots, \alpha_n}{\varphi}.$$

We call $\alpha_1, \ldots, \alpha_n$ the premises of $r$ and $\varphi$ the conclusion of $r$.

**Definition 9.1.1.** *(a) A monotonic formal system is a pair $(U, M)$ where $U$ is a nonempty set and $M$ is a collection of monotonic rules.*
*(b) A subset $S \subseteq U$ is called* **deductively closed** *over $(U, M)$ if for all rules $r = \frac{\alpha_1,\ldots,\alpha_n}{\varphi} \in M$, $\alpha_1, \ldots, \alpha_n \in S$ implies $\varphi \in S$.*

Inspired by Reiter [158], and Apt [3], Marek, Nerode, and Remmel developed a theory of nonmonotonic rule systems [123, 125, 124, 126, 127, 128]. Nonmonotonic rule systems are simple algebraic structures which formalize the notion of nonmonotonic reasoning. Moreover there are simple translations between nonmonotonic rule systems and each of the nonomonotonic formalisms listed above which show that theorems established for nonmonotonic rule systems immediately transfer to corresponding results for each of these nonmonotonic logics.

A *nonmonotonic rule system* is a pair $(U, N)$ where a nonempty set $U$ and a set $N$ of nonmonotonic rules. A nonmonotonic rule of inference is a triple $P, R, c)$, where $P = \{a_1, \ldots, a_n\}$, $R = \{b_1, \ldots, b_m\}$ are finite lists of objects from $U$ and $c \in U$. Each such rule is written in form

$$r = \frac{a_1, \ldots, a_n \colon b_1, \ldots, b_m}{c}.$$

Here $a_1, \ldots, a_n$ are called the *premises* of rule $r$, $b_1, \ldots, b_m$ are called the *restraints* of rule $r$. Either $P$, or $R$, or both may be empty. If $R$ is empty, then the rule $r$ is *monotonic*. If $P = G = \emptyset$, then the rule $r$ is called an axiom. The set $\{a_1, \ldots, a_n\}$ is denoted by $p(r)$, the set $\{b_1, \ldots, b_m\}$ is denoted by $R(r)$, and $c$ is denoted by $c(r)$. The intuitive idea of the non-monotonic rule $r$ is that $c$ is supposed to hold if we have established that $a_1, \ldots, a_n$ are true and there is no evidence that any of $b_1, \ldots, b_n$ have been established.

A subset $S \subseteq U$ is called *deductively closed* if for every rule $r$ of $N$, whenever all premises $a_1, \ldots, a_n$ of $r$ are in $S$ and no restraint $b_1, \ldots, b_m$ of $r$ is in $S$, then the conclusion $c$ of $r$ belongs to $S$.

In a monotonic rule system, the family of deductively closed sets is closed under arbitrary intersections, so that for every $I \subseteq U$ there is the least set $T(I)$ such that $I \subseteq T(I)$ and $T(I)$ is deductively closed. The operator $T$ is monotone, meaning that $I \subseteq J$ implies that $T(I) \subseteq T(J)$. For first order logic, $T(I) = Con(I)$. In nonmonotonic systems, the deductively closed sets are not generally closed under arbitrary intersections. But the deductively closed sets are closed under intersections of *descending* chains. Since $U$ is deductively closed, by the Kuratowski-Zorn Lemma, any $I \subseteq U$ is contained in at least one *minimal* deductively closed set.

Given a set $S$ and an $I \subseteq U$, an *$S$-deduction* of $c$ from $I$ in $(U, N)$ is a finite sequence $< c_1, \ldots, c_k >$ such that $c_k = c$ and, for all $i \leq k$, each $c_i$ is in $I$, or is an axiom, or is the conclusion of a rule $r \in N$ such that all the premises of $r$ are included in $\{c_1, \ldots, c_{i-1}\}$ and all restraints of $r$ are in $U \setminus S$. An *$S$-consequence* of $I$ is an element of $U$ occurring in some $S$-deduction from $I$. $C_S(I)$ is defined to be the set of all $S$-consequences of $I$ in $(U, N)$.

Note that a monotonic rule system can be viewed as a special case of a nonmonotonic rule systems where all the rules are monotonic. In a monotonic system, $C_S(I) = T(I)$ for any subset $S$ of $U$.

For a fixed $S$, the operator $C_S(\cdot)$ is monotonic. That is, if $I \subseteq J$, then $C_S(I) \subseteq C_S(J)$. Also, $C_S(C_S(I)) = C_S(I)$. However, $C_S(\cdot)$ is antimonotonic in $S$, that is, for fixed $I$, $S' \subseteq S$ implies that $C_S(I) \subseteq C_{S'}(I)$.

Generally, $C_S(I)$ is not deductively closed in $(U, N)$. It is perfectly possible to have all the premises of a rule be in $C_S(I)$, all the restraints of that rule be outside $C_S(I)$, but a restraint of that rule be in $S$, preventing the conclusion from being put into $C_S(I)$.

A set $S$ is said to be an *extension* of $I$ in $(U, N)$ if $C_S(I) = S$. Thus in a monotonic rule system, the only extension of $I$ is $T(I)$.

We list below some basic properties of extensions.

**Proposition 9.1.2.** *(a) If $S$ is an extension of $I$, then:*

**(i)** *$S$ is a minimal deductively closed superset of $I$.*

**(ii)** *For every $I'$ such that $I \subseteq I' \subseteq S$, $C_S(I') = S$.*

*(b) The set of extensions of $I$ forms an antichain. That is, if $S_1, S_2$ are extensions of $I$ and $S_1 \subseteq S_2$, then $S_1 = S_2$.*

With each non-monotonic rule $r$, we associate a monotonic rule obtained from $r$ by dropping all the restraints. The rule $r'$ is called the *projection* of rule $r$. We write $M(S)$ for the collection of all projections of all rules from $N(S)$. The projection $(U, N) \mid_S$ is the monotonic rule system $(U, M(S))$. Thus $(U, N) \mid_S$ is obtained as follows: First, non-$S$-applicable rules are eliminated. Then, the restraints are dropped altogether. We have the following characterization theorem.

**Theorem 9.1.3.** *A subset $S \subseteq U$ is an extension of $I$ in $(U, N)$ if and only if $S$ is the deductive closure of $I$ in $(U, N) \mid_S$.*

Based on Theorem 9.1.3, we can give an intuitive explanation of the notion of extension for a nonmonotonic rule system. That is, one way to view an extension is that it represents a justifiable internally consistent set of beliefs given the rules of the system. The idea is to view the nonmonotonic rules of the systems as rules of thumb. One then asserts a certain set of beliefs $B$. Given $B$ we eliminate all the rules which are not consistent with $B$, i.e all those rules $r$ such that $R(r) \cap B \neq \emptyset$. Then $B$ is a justifiable internally consistent set of beliefs if we can derive everything in $B$ from the rules that are left. On a more practical level, Theorem 9.1.3 tells us how to test if a collection $S \subseteq U$ is an extension of $I$ in $(U, N)$.

A simple construction allows us to consider only extensions of the empty set. In fact, if $\mathcal{S}$ is a nonmonotonic rule system, and $I \subseteq U$, then the system $\mathcal{S}(I)$ arises from $\mathcal{S}$ and $I$ by adding to $N$ all the rules of the form $\frac{:}{t}$ for all $t \in I$. We then have:

**Proposition 9.1.4.** *$T$ is an extension of $I$ in $\mathcal{S}$ if and only if $T$ is an extension of $\emptyset$ in $\mathcal{S}(I)$.*

We next introduce briefly some of the nonmonotonic logical systems mentioned above and show how each can be coded into nonmonotonic rule systems.

### 9.1.1 Default Logic

Default Logic is a system based on the language $\mathcal{L}$ of propositional logic. A default rule has the form

$$r = \frac{\alpha : M\beta_1, \ldots, M\beta_k}{\gamma},$$

where $\alpha, \beta_1, \ldots, \beta_k, \gamma$ are formulas of $\mathcal{L}$. Following Reiter [158], a default theory is defined as a pair $(D, W)$ where $D$ is a set of default rules and $W$ is a set of formulas of $\mathcal{L}$.

A theory $S$ is said to be an *extension* of $(D, W)$ if for all rules $r \in D$ as above, if $\alpha \in S$ and $\neg\beta_i \notin S$ for all $i$, then $\gamma \in S$.

$(D, W)$ may be interpreted as a nonmonotonic rule system as follows. For every default rule $r$ as above, construct the following nonmonotonic rule $d_r$:

$$d_r = \frac{\alpha : \neg\beta_1, \ldots, \neg\beta_k}{\gamma}.$$

Next, for every formula $\psi \in \mathcal{L}$, define the rule

$$d_\psi = \frac{:}{\psi}$$

and for all pairs of formulas $\chi_1, \chi_2$ define

$$mp_{\chi_1,\chi_2} = \frac{\chi_1, \chi_1 \supset \chi_2 :}{\chi_2}.$$

Now define the set of rules $N_{D,W}$ as follows:
$N_{D,W} = \{d_r : r \in D\} \cup \{d_\psi : \psi \in W \text{ or } \psi \text{ is a tautology}\} \cup \{mp_{\chi_1,\chi_2} : \chi_1, \chi_2 \in \mathcal{L}\}$.

It was shown in [123] that a set of formulas $S$ is a default extension of $(D, W)$ if and only if $S$ is an extension of nonmonotonic rule system $\langle U, N_{D,W}\rangle$.

### 9.1.2 Nonmonotonic modal logics

McDermott [134] introduced a technique which allows one to create nonmonotonic counterparts of various modal logics. The modal language $\mathcal{L}_L$ has one modal operator $L$, interpreted as necessitation, knowledge, or belief. Given a modal logic $\mathcal{S}$, McDermott defined a consequence operator $Cn_\mathcal{S}$ which allows for application of necessitation to all previously proven formulas, not only to the the axioms of $\mathcal{S}$.

Now, given a set of formulas $T \subseteq \mathcal{L}$ and another set of formulas $I$, interpreted to be the initial assumptions of the reasoning agent, a theory $T$ is called an **$\mathcal{S}$-expansion of $I$** if

$$T = Cn_\mathcal{S}(I \cup \{\neg L\psi : \psi \notin T\}).$$

The set of formulas $\{\neg L\psi : \psi \notin T\}$ represents the so-called "negative introspection with respect to $T$". The modal logic $S$ may be simulated as a nonmonotonic rule system as follows. The universe $U$ will be as before the set of all well formed formulas of the language $\mathcal{L}_L$.

For every formula $\psi \in \mathcal{L}_L$ we consider a rule

$$e_\psi = \frac{: \psi}{\neg L\psi}.$$

Now, given a theory $I$ (the set of initial assumptions) in a modal logic $\mathcal{S}$, and a theory $T \subseteq \mathcal{L}_L$ consider the following set of rules

$$\begin{aligned} N_{I,\mathcal{S}} = \quad & \{d_\psi : \psi \in \mathcal{S}\} \cup \{e_\psi : \psi \in \mathcal{L}_L\} \cup \\ & \{mp_{\chi_1,\chi_2} : \chi_1, \chi_2 \in \mathcal{L}_L\} \cup \\ & \{d_\psi : \psi \in I\} \cup \{d_\psi : \psi \text{ is a tautology of } \mathcal{L}_L\} \end{aligned}$$

It may then be seen that $T$ is an $\mathcal{S}$-expansion of $I$ if and only if $T$ is an extension of the nonmonotonic rule system $(U, N_{I,\mathcal{S}})$.

### 9.1.3 General logic programming

General logic programs extend the usual syntax of (Horn) logic programs by admitting negated atoms in the body of clauses. Specifically, a general clause is an expression of the form:

$$C = p \leftarrow q_1, \ldots, q_m, \neg r_1, \ldots, \neg r_n.$$

Here we only assume that $m \geq 0$ and $n \geq 0$ so that the usual logic programming clauses are special cases of general clauses. General clauses possess the *logical interpretation*:

$$q_1 \wedge \ldots \wedge q_m \wedge \neg r_1 \wedge \ldots \wedge \neg r_n \supset p.$$

As long as we are interested in Herbrand models of general programs, we can consider the propositional theory $ground(P)$ consisting of all ground substitutions of clauses of $P$. While $P$ is usually finite, $ground(P)$ may be infinite if $P$ contains function symbols. There is of course a one-to-one correspondence between Herbrand models of $P$ and propositional models of $ground(P)$.

As is the case for (Horn) logic programming, not every model of the program has a clear computational meaning. Some models of a general program provide a computationally sound meaning to negation. We shall discuss here only the stable models of Gelfond and Lifschitz [73] since stable models are most naturally modeled by extensions of nonmonotonic rule systems.

Given a subset $M$ of the Herbrand universe, and a clause $C$ as above in $ground(P)$, we define $C^M$ as **nil** if $r_j \in M$ for some $1 \leq j \leq n$. Otherwise $C^M = p \leftarrow q_1, \ldots, q_m$. Then we put

$$P^M = \{C^M : C \in ground(P)\}.$$

Since $P^M$ is a Horn program, it possesses a least Herbrand model, $N_M$. Then we call $M$ a *stable model of* $P$ if $M = N_M$. It is easy to see that a stable model of $P$ is a model of $P$. The stable models of logic programs may be encoded as extensions of nonmonotonic rule systems as follows. The universe of all our system, $U$, will be the Herbrand base of the program. Next, to every general propositional clause $C$ as above, assign the rule

$$r_C = \frac{q_1, \ldots, q_m : r_1, \ldots, r_n}{p}.$$

Now, given the program $P$, define

$$N_P = \{r_C : C \in ground(P)\}.$$

Then $M$ is a stable model of $P$ if and only if $M$ is an extension of the nonmonotonic rule system $(U, N_P)$.

### 9.1.4 Proof Schemes

We now return to studying the complexity of the set of extensions of nonmonotonic rule systems. The basic notion used to analyze the Turing complexity of the set of extensions of nonmonotonic rule systems is that of a proof scheme.

A *proof scheme* $s$ is a finite sequence of triples $((c_1, r_1, Z_1), \dots, (c_n, r_n, Z_n))$ where
$c_1, \dots, c_n \in U$, $r_1, \dots, r_n \in N$, $Z_1, \dots, Z_n$ are finite subsets of $U$ such that for all $1 \le j \le n$,

**(1)** $c_1 = c(r_1)$, $Z_1 = R(r_1)$ and $p(r_1) = \emptyset$

**(2)** For $j > 1$, $p(r_j) \subseteq \{c_1, \dots, c_{j-1}\}$, $c(r_j) = c_j$, and $Z_j = Z_{j-1} \cup R(r_j)$.

**(3)** $c_n$ is the conclusion of $s$ and is denoted by $cln(s)$. $Z_n$ is called the support of $s$ and is denoted by $supp(s)$.

Clearly an initial segment of a proof scheme is also a proof scheme.

Notice that the support of a proof scheme $s$, $Z_n$, has the property that for every set $S$ such that $S \cap Z_n = \emptyset$, the sequence $(c_1, \dots, c_n)$ is an $S$-derivation. Conversely if $(c_1, \dots, c_n)$ is an $S$-derivation, then there is a proof scheme

$$s = ((c_1, r_1, Z_1), \dots, (c_n, r_n, Z_n)$$

such that $Z_n \cap S = \emptyset$.

There is a natural preordering on proof schemes. Namely, $s_1 \prec s_2$ if and only if every rule appearing in $s_1$ appears in $s_2$ as well. The relation $\prec$ is not a partial ordering but it is well-founded. We can thus talk about minimal proof schemes for a given element $c \in U$. Intuitively, a minimal proof scheme carries the minimal information necessary to derive its conclusion. Since the support of every proof scheme is finite, the negative information carried in such a proof scheme is finite.

Proof schemes can be used to characterize extensions. We say that a set $S$ *admits* a proof scheme $s$ if $supp(s) \cap S = \emptyset$. We then have the following characterization of extensions.

**Proposition 9.1.5.** *Let $\mathcal{S} = (U, N)$ be a nonmonotonic rule system. Let $S \subseteq U$. Then $S$ is an extension of $\mathcal{S}$ if and only if the following conditions are met:*

**(a)** *If $s$ is a proof scheme and $S$ admits $s$, then $c(s) \in S$.*

**(b)** *Whenever $a \in S$ then there exists a proof scheme $s$ such that $S$ admits $s$.*

It is easy to see that we can restrict to minimal proof schemes in Proposition 9.1.5.

### 9.1.5 $\Pi^0_1$ Classes and extensions

We now give the basic results from [123, 125, 124, 127] on the complexity of extensions in computable nonmonotonic rule systems.

The canonical index, $can(X)$, of the finite set $X = \{x_1 < \ldots < x_n\} \subseteq \omega$ is defined as $2^{x_1} + \ldots + 2^{x_n}$ and the canonical index of $\emptyset$ is defined as 0. Let $D_k$ be the finite set whose canonical index is $k$, i.e., $can(D_k) = k$.

Let $(U, N)$ be a nonmonotonic rule system where $U \subseteq \omega$. We shall identify a rule $r$ with the code of a triple $\langle k, l, \varphi \rangle$ where $D_k = p(r)$, and $D_l = R(r)$,

$\varphi = c(r)$. In this way we can think about $N$ as a subset of $\omega$ as well. This given, we then say that a NRS $\mathcal{S} = (U, N)$ is *computable* if $U$ and $N$ are computable subsets of $\omega$.

There are two important subclasses of computable NRS's introduced in [124], namely locally finite and highly computablee nonmonotonic rules systems. We say that the system $(U, N)$ is *locally finite* if for each $c \in U$, there are only finitely many $\prec$-minimal proof schemes with conclusion $c$. Given a proof scheme for $c$, $s = (c_1, r_1, Z_1), \ldots, (c_n, r_n, Z_n))$, the code of $s$, $c(s)$, is defined by

$$c(s) = \langle\langle c_1, r_1, Z_1\rangle, \ldots, \langle c_n, r_n, Z_n\rangle.$$

If $\langle U, N\rangle$ is a locally finite computable nonmonotonic rule system and $c \in U$, we let $Dr_c$ denote the set of codes of all $\prec$-minimal proof schemes for $c$. We say that $(U, N)$ is *highly computable* if $(U, N)$ is computable, locally finite, and the map $c \mapsto can(Dr_c)$ is partial computable. The latter means that there is an effective procedure which, when applied to any $c \in U$, produces a canonical index of the set of all codes of $\prec$-minimal proof schemes with conclusion $c$. The following results are due to Marek, Nerode, and Remmel [124].

**Theorem 9.1.6.** *For any highly computable NRS system $\mathcal{S} = (U, N)$, there is a highly computable tree $T_{\mathcal{S}}$ such that there is an effective 1:1 degree preserving correspondence between $[T_{\mathcal{S}}]$ and $\mathcal{E}(\mathcal{S})$. Vice versa, for any highly computable tree $T$, there is a highly computable NRS system $\mathcal{S}_T = (U, N)$ such that there is an effective 1:1 degree preserving correspondence between $[T]$ and $\mathcal{E}(\mathcal{S}_T)$.*

**Theorem 9.1.7.** *For any locally finite computable NRS system $\mathcal{S} = (U, N)$, there is a finitely branching computable tree $T_{\mathcal{S}}$ such that there is an effective 1:1 degree preserving correspondence between $[T_{\mathcal{S}}]$ and $\mathcal{E}(\mathcal{S})$. Vice versa, for any highly computable tree $T$ in $\mathbf{0}'$, there is a locally finite computable NRS system $\mathcal{S}_T = (U, N)$ such that there is an effective 1:1 degree preserving correspondence between $[T]$ and $\mathcal{E}(\mathcal{S}_T)$.*

**Theorem 9.1.8.** *For any computable NRS system $\mathcal{S} = (U, N)$, there is a computable tree $T_{\mathcal{S}}$ such that there is an effective 1:1 degree preserving correspondence between $[T_{\mathcal{S}}]$ and $\mathcal{E}(\mathcal{S})$. Vice versa, for any computable tree $T$, there is a computable NRS system $\mathcal{S}_T = (U, N)$ such that there is an effective 1:1 degree preserving correspondence between $[T]$ and $\mathcal{E}(\mathcal{S}_T)$.*

As usual, we can immediately derive a number of corollaries about the complexity of the set of extensions of a computable nonmonotonic rule systems by transferring known results about $\Pi^0_1$-classes. For example, for computable nonmonotonic rule systems, we have the following results, see [124].

**Corollary 9.1.9.** **(a)** *Every computable NRS system $\mathcal{S} = (U, N)$ which has an extension has an extension $E$ such that $E \leq_T B$ where $B$ is a complete $\Pi^1_1$-set.*

**(b)** *If $\mathcal{S} = (U, N)$ is a computable NRS system with a unique extension $E$, then $E$ is hyperarithmetic.*

**Corollary 9.1.10.** **(a)** *There is a computable NRS system $\mathcal{S} = (U, N)$ such that $\mathcal{S}$ has an extension but $\mathcal{S}$ has no extension which is hyperarithmetic.*

**(b)** *For each computable ordinal $\alpha$, there exists a computable NRS system $\mathcal{S} = (U, N)$ possessing a unique extension $E$ such that $E \equiv_T \mathbf{0}^{(\alpha)}$.*

**Corollary 9.1.11.** *Let $\mathcal{S} = (U, N)$ be a highly computable nonmonotonic rule system such that $\mathcal{E}(\mathcal{S}) \neq \emptyset$. Then*

**(a)** *There exists an extension $E$ of $\mathcal{S}$ such that $E$ is low.*

**(b)** *If $\mathcal{S}$ has only finitely many extensions, then every extension $E$ of $\mathcal{S}$ is computable.*

In the other directions, there are a number of corollaries of Theorem 9.1.6 which allow us to show that there are highly computable NRS systems $\mathcal{S}$ such that the set of degrees realized by elements of $\mathcal{E}(\mathcal{S})$ are still quite complex. Again all these corollaries follow by transferring results of Section 3.

**Corollary 9.1.12.** **(a)** *There is a highly computable nonmonotonic rule system*
*$(U, N)$ which has $2^{\aleph_0}$ extensions but has no computable extensions.*

**(b)** *There is a highly computable nonmonotonic rule system $(U, N)$ such that $(U, N)$ has $2^{\aleph_0}$ extensions and any two extensions $E_1 \neq E_2$ of $(U, N)$ are Turing incomparable.*

**(c)** *There is a highly computable nonmonotonic rule system $(U, N)$ such that $(U, N)$ has $2^{\aleph_0}$ extensions and if $\mathbf{a}$ is the degree of any extension $E$ of $(U, N)$ and $\mathbf{b}$ is any computably enumerable degree such that $\mathbf{a} <_T \mathbf{b}$, then $\mathbf{b} \equiv_T \mathbf{0}'$.*

**(d)** *If $\mathbf{a}$ is any computably enumerable Turing degree, then there is a highly computable nonmonotonic rule system $(U, N)$ such that $(U, N)$ has $2^{\aleph_0}$ extensions and the set of computably enumerable degrees $\mathbf{b}$ which contain an extension of $(U, N)$ is precisely the set of all computablly enumerable degrees $\mathbf{b} \geq_T \mathbf{a}$.*

Finally, we note that there are analogues of Corollaries 9.1.11 and 9.1.12 which hold for computable locally finite nonmonotonic rule systems. That is, one can replace highly computable nonmonotonic rule systems by computable locally finite nonmonotonic rule systems if one replaces all the statements about degrees of extensions by the corresponding statement relative to a $\mathbf{0}'$ oracle. For example, the analogue of part (1) of Corollary 9.1.11 is that every computabe locally finite nonmonotonic rule system $\mathcal{S}$ such that $\mathcal{E}(\mathcal{S}) \neq \emptyset$ has an extension $E$ such that the jump of $E$ is computable in $\mathbf{0}''$, while the analogue of (1) of Corollary 9.1.12 is that there exists a computable locally finite nonmonotonic rule system $(U, N)$ which has $2^{\aleph_0}$ extensions but which has no extension which is computable in $\mathbf{0}'$. Moreover, we can weaken the hypothesis of locally finite and highly computable slightly and still derive the same theorems. That is, we say that a computable nonmonotonic rule system $(U, N)$ has the *finite support property* if for each $c \in U$, the set of supports of all $\prec$-minimal proofs schemes of $c$ is finite. It is possible for a $c \in U$ to have infinitely many $\prec$-minimal proof schemes with the same support so that not every computable nonmonotonic rule system with the finite support property is locally finite. Similarly, we say that a computable nonmonotonic rule system $(U, N)$ which has the finite support property has the *computable finite support property* if there is an effective algorithm which given any $c \in U$, produces the canonical index of the set of canonical indices of the supports of all the $\prec$-minimal proof schemes of $c$. See [127] for further details. Finally there are complete analogues of all the results of this section which apply to finite predicate logic programs or finite predicate logic default theories.

### 9.1.6 Predicate Logic Programs

We end this section with an extension of the results of the previous section to finite predicate logic programs. In this setting, we get a perfect correspondence between $\Pi^0_1$ classes and the set of stable models of finite predicate logic program. That is, given any finite predicate logic program $P$, there is a computable tree $T_P$ such that there is an effective one-to-one correspondence between the set of stable models of $P$ and the paths through $T_P$. Vice versa, given any computable tree $T$, there is a computable program $P_T$ such that there is an effective one to one correspondence between the set of stable models of $P_T$ and the paths through $T$. Moreover under these correspondences, bounded trees correspond to a natural set of finite predicate logic programs called finite support property programs $FSP$ and $r.b$ programs correspond to computably $FSP$ programs. These correspondences can be found in [127] and they essentially allow us to translate all the results on index sets for trees to results on index sets for finite predicate logic programs.

For an introductory treatment of Predicate Logic Programs, see [119]. Here is a brief self-contained account of their stable models [73]. Assume as given a fixed first order language based on predicate letters, constants, and function symbols. The Herbrand base of the language is defined as the set $B_{\mathcal{L}}$ of all ground atoms (atomic statements) of the language. A literal is an atomic formula or its negation, a ground literal is an atomic statement or its negation. A Logic Program $P$ is a set of "program clauses", that is, an expression of the form:

$$p \leftarrow l_1, \ldots, l_k$$

where $p$ is an atomic formula, and $l_1, \ldots, l_k$ is a list of literals.

Then $p$ is called the conclusion of the clause, the list $l_1, \ldots, l_k$ is called the body of the clause. Ground clauses are clauses without variables. Horn clauses are clauses with no negated literals, that is, with atomic formulas only in the body. Horn clause programs are programs $P$ consisting of Horn clauses. Each such program has a least model in the Herbrand base determined as the least fixed point of a continuous operator $T_P$ representing 1-step Horn clause logic deduction ([119]).

A ground instance of a clause is a clause obtained by substituting ground terms (terms without variables) for all variables of the clause. The set of all ground instances of the program $P$ is called *ground*$(P)$.

Let $M$ be any subset of the Herbrand base. A ground clause is said to be $M$-applicable if the atoms whose negations are literals in the body are not members of $M$. Such clause is then *reduced* by eliminating remaining negative literals. This monotonization $GL(P, M)$ of $P$ with respect to $M$ is the propositional Horn clause program consisting of reducts of $M$-applicable clauses of *ground*$(P)$ (see Gelfond-Lifschitz [73]). Then $M$ is called a *stable model* for $P$ if $M$ is the least model of the Horn clause program $GL(M, P)$. We denote this least model as $N_{M,P}$. It is easy to see that a stable model for $P$ is a minimal model of $P$ ([73]). We denote by $Stab(P)$ the set of all stable models of $P$. There may be no, one, or many stable models of $P$.

A *proof scheme* for $p$ with respect to $P$ is a sequence of triples

$$((p_l, C_l, S_l))_{1 \leq l \leq n},$$

with $n$ a natural number, such that the following conditions all hold.

**(1)** Each $p_l$ is in $B_{\mathcal{L}}$. Each $C_l$ is in *ground*$(P)$. Each $S_l$ is a finite subset of $B_{\mathcal{L}}$.

**(2)** $p_n$ is $p$.

The $S_l$, $C_l$ satisfy the following conditions. For all $1 \leq l \leq n$, one of **(a)**, **(b)**, **(c)** below holds.

- **(a)** $C_l$ is $p_l \leftarrow$, and $S_l$ is $S_{l-1}$,
- **(b)** $C_l$ is $p_l \leftarrow \neg s_1, \ldots, \neg s_r$ and $S_l$ is $S_{l-1} \cup \{s_1, \ldots, s_r\}$, or
- **(c)** $C_l$ is $p_l \leftarrow p_{m_1}, \ldots, p_{m_k}, \neg s_1, \ldots, \neg s_r$, $m_1 < l, \ldots, m_k < l$, and $S_l$ is $S_{l-1} \cup \{s_1, \ldots, s_r\}$.

(We put $S_0 = \emptyset$). Suppose that $\varphi = ((p_l, C_l, S_l))_{1 \leq l \leq n}$ is a proof scheme. Then *conc*$(\varphi)$ denotes atom $p_n$ and is called the *conclusion* of $\varphi$. Also, *supp*$(\varphi)$ is the set $S_n$ and is called the *support* of $\varphi$.

Condition (3) tells us how to construct the $S_l$ inductively, from the $S_{l-1}$ and the $C_l$. The set $S_n$ consists of the negative information of the proof scheme.

Formally, preorder proof schemes $\varphi$, $\psi$ by $\varphi \prec \psi$ if

**(1)** $\varphi, \psi$ have same conclusion,

**(2)** Every clause in $\varphi$ is also a clause of $\psi$.

The relation $\prec$ is reflexive, transitive, and well-founded. Minimal elements of $\prec$ are minimal proof schemes.

We can characterize stable models via proof schemes as follows.

**Proposition 9.1.13.** *Let $P$ be a program. Also, suppose that $M$ is a subset of the Herbrand universe $B_{\mathcal{L}}$. Then $M$ is a stable model of $P$ if, and only if, for every $p \in B_{\mathcal{L}}$, it is true that $p$ is in $M$ if and only if there exists a proof scheme $\varphi$ with conclusion $p$ such that the support of $\varphi$ is disjoint from $M$.*

A *finitary support* program (*FSP* program) is a Logic Program such that for every atom $p$, there is a finite set of finite sets S, which are exactly the inclusion-minimal supports of all those minimal proof schemes with conclusion $p$.

A *recursively* FSP program is an FSP recursive program such that we can uniformly compute the finite family of supports of proof schemes with conclusion $p$ from $p$. The meaning of this is obvious, but we need a technical notation for the proofs. Start by listing the whole Herbrand base of the program, $B_{\mathcal{L}}$ as a countable sequence in one of the usual effective ways. This assigns an integer (Gödel number) to each element of the base, its place in this sequence. This encodes finite subsets of the base as finite sets of natural numbers, all that is left is to code each finite set of natural numbers as a single natural number, its *canonical index*. We also set $can(\emptyset) = 0$. If program $P$ is FSP, and the list, in order of magnitude, of Gödel numbers of all minimal support of schemes with conclusion $p$ is

$$Z_1^p, \ldots, Z_{l_r}^p,$$

then define a function $su^P : B_{\mathcal{L}} \to \omega$ as below.

$$p \mapsto can(\{can(Z_1^p), \ldots, can(Z_{l_r}^p)\})$$

We call a Logic Program $P$ a *computably* FSP program if it is FSP and the function $su^P$ is computable.

In [127], Marek, Nerode, and Remmel proved the following two results.

**Theorem 9.1.14.** *We suppose that the first order language $\mathcal{L}$ has infinitely many ground atoms.*

**(a)** *Then for any computable program $P$ in $\mathcal{L}$, there exists a computable tree $T \subseteq \omega^{<\omega}$ and an effective one-to-one degree preserving correspondence between the set of all stable models of $P$, $Stab(P)$ and $[T]$, the set of all infinite paths through $T$.*

**(b)** *If, in addition to the hypothesis of (1), program $P$ is FSP, then the tree $T$ is bounded.*

**(c)** *If, in addition to the hypothesis of (2), program $P$ is computably FSP, then the tree $T$ is a highly computable tree.*

**Theorem 9.1.15.** *Let $C$ be any $\Pi_1^0$-class. Then*

**(a)** *There is a finite program, $P$, and an effective one-to-one degree preserving correspondence between the elements of $C$ and the set of all stable models of $P$, $Stab(P)$.*

**(b)** *If in addition $C$ is of the form $[T]$ for a bounded computable tree $T$, then $P$ can be chosen FSP.*

**(c)** *If in addition $T$ is a highly computable tree, then $P$ can be chosen recursively FSP.*

These two results were strengthened by Cenzer, Marek, and Remmel [25] to prove the following.

**Theorem 9.1.16.** *We suppose that the first order language $\mathcal{L}$ has infinitely many ground atoms.*

**(a)** *Then for any finite predicate logic program $P$ in $\mathcal{L}$, there exists a primitive recursive tree $T \subseteq \omega^{<\omega}$ and an effective one-to-one degree preserving correspondence between the set of all stable models of $P$, $Stab(P)$ and $[T]$, the set of all infinite paths through $T$.*

**(b)** *If, in addition to the hypothesis of (1), the program $P$ is FSP, then the tree $T$ is bounded.*

**(c)** *If, in addition to the hypothesis of (2), the program $P$ is computably FSP, then the tree $T$ is a highly computable tree.*

**Theorem 9.1.17.** *Let $T$ be any primitive recursive tree. Then*

**(a)** *There is a finite program, $P$, and an effective one-to-one degree preserving correspondence between the elements of $[T]$ and the set of all stable models of $P$, $Stab(P)$.*

**(b)** *If in addition $T$ is bounded, then $P$ is FSP.*

**(c)** *If in addition $T$ is a highly computable tree, then $P$ is computably FSP.*

The crucial point about the proof of Theorems 9.1.16 and 9.1.17 is that they are completely uniform. For example, given a finite predicate logic program $P$, we can uniformly find the index of a primitive recursive tree $T_P$ such that there is an effective one-to-one degree preserving correspondence between the stable models of $P$ and the elements of $[T_P]$. Vice versa, given any primitive recursive tree $T$, we can uniformly find a finite predicate logic program $P_T$ such that there is an effective one-to-one degree preserving correspondence between the stable models of $P_T$ and the elements of $[T]$. This means that one can transfer all the index set results about trees and $\Pi^1_0$ classes to index sets about finite predicate logic programs. Thus if we fix some computable first order language $\mathcal{L}$ with infinitely many ground atoms, then we can effectively list all finite predicate logic programs $P_0, P_1, \ldots$. Then for any property $\mathcal{R}$, we can define an index set

$$Prog(\mathcal{R}) = \{e : P_e \text{ has property } \mathcal{R}\}.$$

We can then transfer all the index set results to index set results for finite predicate logic programs. For example, from Theorems 5.1.6 and 5.1.7 we obtain the following.

**Theorem 9.1.18.** **(a)** *$Prog(FSP) = \{e : P_e$ has the FSP property$\}$ is $\Sigma^0_3$ complete set.*

**(b)** *$Prog($ computably $FSP) = \{e : P_e$ has the computably FSP property$\}$ is $\Pi^0_3$ complete set.*

Similarly, Theorem 5.2.13 implies the following.

**Theorem 9.1.19.** **(a)** *$Prog($computably FSP and has $\geq \aleph_0$ stable models$)$ is $D^0_3$ complete and $Prog($computably FSP and has $< \aleph_0$ stable models$)$ is $\Sigma^0_3$ complete.*

**(b)** *$Prog($is FSP and has $\geq \aleph_0$ stable models$)$ is $\Pi^0_4$ complete and $Prog($is FSP and has $< \aleph_0$ stable models$)$ is $\Sigma^0_4$ complete.*

**(c)** *$Prog($has $\geq \aleph_0$ stable models$), Prog($has $< \aleph_0$ stable models$)$ is $(\Sigma^1_1, \Pi^1_1)$ complete.*

See [25] for further details.

## 9.2 $\omega$ languages

An $\omega$-language is a set of infinite sequences (words) on a countable language, and corresponds to a set of real numbers in a natural way. Languages may be described by logical formulas in the arithmetical hierarchy and also may be described as the set of words accepted by some type of automata or Turing machine. Certain families of languages, such as the $\Sigma^0_2$ languages, may enumerated as $P_0, P_1, \ldots$ and then an index set associated to a given property $R$ (such as finiteness) of languages is just the set of $e$ such that $P_e$ has the property. The complexity of index sets for 7 types of languages is determined for various properties related to the size of the language.

This section is concerned with the connections between index sets for $\omega$-languages and index sets for computable analysis. Let $X$ be a finite or infinite alphabet and let $X^\omega$ denote the set of all infinite sequences $(x_0, x_1, \ldots)$ of elements of $X$. An automaton $M$ over $X$ is a quadruple $[Z, z_0, R, Z_f]$ where $Z$ is a nonempty set of states, $z_0 \in Z$ is the initial state, $Z_f \subseteq Z$ is a set of accepting states and $R \subset Z \times X \times Z$ is transition relation. An automaton $M$ is said to be *computable* provided $Z$ is an initial segment of the natural numbers $\omega$ and $Z_f$ and $R$ are computably enumerable (c.e). $M$ is *strictly computable* if $Z_f$ is a computable set. An $\omega$-language is just a subset $W$ of $X^\omega$ which consists of the strings accepted by some computable automaton. Such languages arise naturally in several areas of computer science such as temporal logic, model checking, automata theory, and fair terminations, see [**?**] and [**?**].

Such languages also naturally arise in the study of computable analysis and the study of $\Pi^0_1$ classes. For example, in computable analysis, we refer to an effectively closed subset of $\{0,1\}^\omega$ as a $\Pi^0_1$ class. Such a class may be presented as the $\omega$-language of infinite strings which are accepted by a computable automaton

$M = [Z, z_0, R, Z_f]$, in the following sense. Suppose that $w = (w(0), w(1), \ldots) \in \{0,1\}^\omega$. A *run* of $w$ for $M$ is a sequence $(\langle z_0, w(0), z_1\rangle, \langle z_1, w(1), z_2\rangle, \ldots)$ such that for all $i$, $\langle z_i, w(i), z_{i+1}\rangle \in R$. Then the set of all $w \in \{0,1\}^\omega$ such that there exists a run $(\langle z_0, w(0), z(1)\rangle, \langle z_1, w(1), z_2\rangle, \ldots)$ of $w$ for $M$ such that for all $i$, $z_i \in Z_f$ is a $\Pi^0_1$ class. This provides a topological context for the study of $\omega$-languages. Similarly real numbers may be represented in various ways as infinite words and thus computable analysis may be viewed as the study of effectively Borel sets (that is, $\omega$-languages) and effectively Borel measurable functions on infinite words. (See [198] for a full introduction to computable analysis.)

There are other notions of acceptance for $\omega$-languages relative to computable automata which correspond with other natural topological notions. Several of these different types of acceptance conditions will be described in section 3. A presentation of the topological aspects of $\omega$-languages and their relation to the effective Borel hierarchy can be found in the survey article by L. Staiger [189]. We will show how one can translate between the index sets used by Staiger [188] and the index sets used here in our book. That is, an index for a formal language may be obtained directly from an index for the machine which accepts the language or it may be obtained from the representation of the language in the effective Borel hierarchy. We will demonstrate the connection between these two approaches and show how, in most cases, one can reduce the index set result for $\omega$-languages to an index set result for some type of $\Pi^0_1$ class. This will allow us to apply the results of **??** to determine the complexity of a vast array of index sets for $\omega$-languages. Second, we shall study index sets for several other families of $\omega$-languages corresponding to various properties. For example, Staiger [188] classified the index sets for *pairs* $\langle V, W\rangle$ such that $V \subset W$ which he called *verification properties*. We will consider various *approximate verification properties* such as considering index sets for pair of languages $\langle V, W\rangle$ such that $W - V$ is finite, is a set of measure zero or contains only finitely many computable sequences.

e shall introduce 7 classes that we shall study in this paper and develop an natural indexing scheme for each type of class. The advantage of our indexing schemes is that we can easily connect them to known indexing schemes for $\omega$-languages and indexing schemes for $\Pi^0_1$-classes. In section 3, we shall define various notions of acceptance for $\omega$-languages. In sections 4, 5, and 6 we shall derive a number of new index set results for our seven types of classes and for $\omega$-languages. Section 4 is devoted to index sets for various cardinality conditions, section 5 is devoted to index sets for various measure conditions and section 6 is devoted to various weak verification conditions.

We begin with the notion of an effective topological space $\mathcal{X}$. Our notion of effective topological space is closely related to the notion of effective topological space defined in Kalantari and Retzlaff [**?**] except that we require primitive recursive intersection and inclusion relations instead of just computable intersection and inclusion relations. That is, suppose $\mathcal{X}$ is some separable metric space such that there is an effective enumeration $U_0, U_1, \ldots$ of a basis $\Delta$ for the space $X$ such that the following hold.

**(I)** $\Delta$ is closed under finite intersections.

**(II)** $\emptyset$ and $\mathcal{X}$ is in $\Delta$.

We then say that $\mathcal{X}$ is an *effective topological space* if, in addition, the following hold.

**(i)** The operations of union and intersection and the inclusion relation are all primitive recursive. This means that there are primitive recursive functions $\pi_{ou}$, $\pi_{cu}$, and $\pi_i$ such that
(a) $U_{m_0}\cup\cdots\cup U_{m_k} \subseteq U_{n_0}\cup\cdots\cup U_{n_l} \iff \pi_{ou}(\langle m_0,\ldots,m_k\rangle,\langle n_0,\ldots n_l\rangle) = 1$;
(b) $\overline{U_{m_0}\cup\cdots\cup U_{m_k}} \subseteq U_{n_0}\cup\cdots\cup U_{n_l} \iff \pi_{cu}(\langle m_0,\ldots,m_k\rangle,\langle n_0,\ldots n_l\rangle) = 1$;
(c) $U_m \cap U_n = U_{\pi_i(m,n)}$.

**(ii)** For any $m$, $s$, and $x \in U_m$, there is an $n > s$ such that $x \in \overline{U_n} \subseteq U_m$.

Given such an enumeration, we can also define an enumeration of basic open sets for the product space $\mathcal{X}\otimes\mathcal{X} = \mathcal{X}^2$ and for the disjoint union $2\mathcal{X} = \mathcal{X}\oplus\mathcal{X} = \{(i,x) : i \in \{0,1\} \ \& \ x \in \mathcal{X}\}$ of two copies of $\mathcal{X}$. Then for $\mathcal{X}^2$, the basic open set $V_{\langle m,n\rangle} = U_m \times U_n$. For $\mathcal{X}\oplus\mathcal{X}$, let $V_{2m} = \{0\}\times U_m$ and $V_{2m+1} = \{1\}\times U_m$.

Here are some specific examples.

**(A)** For the space $\mathcal{X} = \{0,1\}^\omega$, we have a basis of sets of the form $I(\sigma) = \{x : \sigma \prec x\}$, where $\sigma \in \{0,1\}^{<\omega}$. The finite sequences $\sigma \in \{0,1\}^{<\omega}$ may be enumerated as $\emptyset, (0), (1), \ldots$, so that in general $bin(n+1) = 1^\frown\sigma_n$. Then we simply let $U_0 = \emptyset$ and for all $n \geq 1$, $U_n = I(\sigma_{n-1})$.

**(B)** For the real line $\Re$, there is a basis which consists of $\emptyset$ and $\Re$ plus all open intervals $(q,r)$ where $q < r$ are rationals.

**(C)** For the space $[0,1]$ (the real interval), there is a basis consisting of $\emptyset$ and $[0,1]$ plus all open intervals $(q,r)$ where $0 \leq q < r \leq 1$ are rationals, together with the half-open intervals $[0,r)$ and $(q,1]$. See [37] for a specific enumeration.

We are now ready to define our enumerations. We start with the enumerations for $\Sigma^0_1$ and $\Pi^0_1$ classes. Let $W_e$ be the $e^{th}$ computably enumerable subset of $\mathbb{N}$, that is, the domain of the $e^{th}$ partial computable function, $\phi_e$.

**(1)** We define the $\Sigma^0_1$ class $S_{1,e}$ with index $e$ to be

$$S_{1,e} = \bigcup_{n\in W_e} U_n.$$

**(2)** We define the $\Pi^0_1$ class $P_{1,e}$ with index $e$ to be

$$P_{1,e} = \mathcal{X} \setminus S_{1,e} = \mathcal{X} \setminus \bigcup_{n\in W_e} U_n = \bigcap_{n\in W_e} \mathcal{X}\setminus U_n.$$

As described above, in the paper [35], we gave an alternative definition of the $e^{th}$ $\Pi^0_1$ class $P'_{1,e}$ for the space $\omega^\omega$ as the set of paths through the $e^{th}$ primitive recursive tree $T_e$. In [37], we observed that the Kleene normal form theorem can be used to show that these definitions are equivalent in a certain sense. That is, there are primitive recursive functions $\phi$ and $\psi$ such that $P'_{1,e} = P_{1,\phi(e)}$ and $P_{1,e} = P'_{1,\psi(e)}$. This easily follows from Theorem 4.1 of [37] which proves that for any c.e. tree $T$, there is a primitive recursive (and indeed a polynomial-time) tree $S$ such that $S$ and $T$ have the same set of infinite extensions. However, we could just as well have given an alternate version of enumeration for the $\Sigma^0_1$ classes in terms of primitive recursive unions of basic open sets. In fact, these two versions of effectively closed sets are still equivalent in the more general setting.

**Lemma 9.2.1.** *For any effective topological space $\mathcal{X}$ and any $\Sigma^0_1$ subset $W$ of $\mathbb{N}$, there exists a primitive recursive set $V$ such that $\bigcup_{n\in W} U_n = \bigcup_{n\in V} U_n$.*

*Proof.* Let $W = W_e$ and let $W_{e,s}$ be the numbers enumerated into $W_e$ by stage $s$. Thus $\{\langle n,e,s\rangle : n \in W_{e,s}\}$ is primitive recursive. Now let

$$m \in V \iff (\exists n, s < m)[U_m \subset U_n \ \&\ n \in W_{e,s}].$$

We claim that $\bigcup_{m\in V} U_m = \bigcup_{n\in W_e} U_n$. If $x \in U_m$ for some $m \in V$, then there is $n \in W_e$ such that $U_m \subset U_n$ so that $\bigcup_{m\in V} \subseteq \bigcup_{n\in W_e} U_n$. On the other hand, suppose that $x \in U_n$ for some $n \in W$ and let $s > n$ be large enough so that $n \in W_{e,s}$. By the definition of an effective topological space, there must be some $m > s$ such that $x \in U_m$ and $U_m \subset U_n$ so that $m \in V$. □

A similar phenomenon will occur for the strong $\Sigma^0_2$ and $\Pi^0_2$ classes defined below. That is, a $\Sigma^0_2$ class is an effective union of $\Pi^0_1$ classes so that every $\Sigma^0_2$ class is of the form $\bigcup_{m=0}^\infty \bigcap_{n=0}^\infty \{\mathcal{X} - U_n : \langle m,n\rangle \in W\}$ for some c.e. set $W$. A $\Pi^0_2$ class is the complement of a $\Sigma^0_2$ class. Thus we can use the following enumerations for $\Sigma^0_2$ and $\Pi^0_2$ classes.

We define the $\Sigma^0_2$ class with index $e$ to be

$$S_{2,e} = \bigcup_{m=0}^{\infty} \bigcap_{n=0}^{\infty} \{\mathcal{X} - U_n : \langle m,n\rangle \in W_e\}.$$

**(4)** We define the $\Pi^0_2$ class with index $e$ to be $P_{2,e} = \mathcal{X} - S_{2,e}$.

Note that by the S-M-N theorem, there is a primitive recursive function $\pi$ such that $W_{\pi(e,m)} = \{n : \langle m,n\rangle \in W_e\}$. Thus for each $m$,

$$P_{1,\pi(e,m)} = \mathcal{X} \setminus \bigcup_{n=0}^{\infty} \{U_n : \langle m,n\rangle \in W_e\},$$

so that

$$S_{2,e} = \bigcup_{m=0}^{\infty} P_{1,\pi(e,m)}.$$

It follows that

$$P_{2,e} = \bigcap_{m=0}^{\infty} S_{1,\pi(e,m)}.$$

Next we shall turn our attention to strong $\Sigma_2^0$ and strong $\Pi_2^0$ classes. Given any oracle $C$, we can generalize the definitions of $\Sigma_1^0$ classes and $\Pi_1^0$ classes relative to the oracle $C$ in a natural way by replacing the $e^{th}$ c.e. set $W_e$ by the $e^{th}$ c.e. set relative to $C$, $W_e^C$, to give rise to the notion of $\Sigma_1^{0,C}$ and $\Pi_1^{0,C}$ classes. Let $\mathbf{0}'$ denote the jump of the empty set, then we define a *strong* $\Sigma_2^0$ class to be a $\Sigma_1^{0,\mathbf{0}'}$ class and a *strong* $\Pi_2^0$ class to be a $\Pi_1^{0,\mathbf{0}'}$ class. Our next lemma, a generalization of Theorem 4.1 of [**?**], will help us define natural enumerations for strong $\Sigma_2^0$ and strong $\Pi_2^0$ classes.

**Lemma 9.2.2.** *For any effective topological space $\mathcal{X}$, any oracle $C$ and any $\Sigma_2^{0,C}$ subset $W$ of $\mathbb{N}$, there exists a $\Pi_1^{0,C}$ subset $Y$ such that $\bigcup_{n \in W} U_n = \bigcup_{p \in Y} U_p$.*

*Proof.* Let $C'$ denote the jump of $C$ so that $W$ is c.e. in $C'$. By the relativized version of Lemma 9.2.1, we can obtain a set $V$ which is primitive recursive in $C'$ such that $\bigcup_{n \in W} U_n = \bigcup_{m \in V} U_m$. By the relativized version of the Schoenfield limit lemma, see [181], there is a sequence $\{V_n\}_{n \in \omega}$ which is uniformly computable in $C$ such that $V = lim_s V_s$. That is, for any $m$, there is some modulus of convergence $p_m$ such that $m \in V \iff m \in V_s$ for all $s > p_m$. Hence we define our set $Y$ as follows:

$$p \in Y \iff (\exists m < p)[U_p \subset U_m \ \& \ (\forall s > p)(m \in V_s)].$$

The desired equality can be checked as in Lemma 9.2.1. □

For the space $\{0,1\}^\omega$, this result is essentially due to Jockush, Lewis and Remmel [86] who showed that for any finitely branching tree $T \subseteq \omega^{<\omega}$ which is highly computable in $\mathbf{0}'$, there is a finitely branching recursive tree $T'$ with the same set of infinite paths.

In the case where $C = \emptyset$, a $\Sigma_2^{0,C}$ set is just a set which is $\Sigma_1^0$ in $\mathbf{0}'$ and a $\Pi_1^{0,C}$ set is just the complement of a c.e. open set. Thus Lemma 9.2.2 says that for any $e$, there is an $f$ such that

$$\bigcup_{n \in W_e^{\mathbf{0}'}} U_n = \bigcup_{m \notin W_f} U_m.$$

This leads to following natural enumerations of strong $\Sigma_2^0$ and strong $\Pi_2^0$ classes.

**(1)*** We define the strong $\Sigma_2^0$ class with index $e$ to be

$$S_{2,e}^* = \bigcup_{m \notin W_e} U_m.$$

**(2)*** We define the strong $\Pi_2^0$ class with index $e$ to be

$$P_{2,e}^* = \mathcal{X} \setminus S_{2,e}^* = \bigcap_{n \notin W_e} X \setminus U_n.$$

It is easy to see from our previous lemmas that for each $e$, we can find an index $f$ such that $V_f$ is the $f^{th}$ primitive recursive set in $\mathbf{0}'$ and

$$S^*_{2,e} = \bigcup_{n \in V_f} U_n.$$

This means that most results about $\Sigma^0_1$ and $\Pi^0_1$ classes can be lifted to results about strong $\Sigma^0_2$ and strong $\Pi^0_2$ classes by relativizing the proofs to the oracle $\mathbf{0}'$.

Next we consider a special class of $\Sigma^0_3$ classes that are the effective unions of sets of strong $\Pi^0_2$ classes. These classes are said to be $\Sigma^\sigma_1$ in the notation of $\omega$-languages, see section three. Thus we define the $\Sigma^\sigma_1$ class with index $e$ to be

$$S^*_{3,e} = \bigcup_{m=0}^{\infty} \bigcap_n \{\mathcal{X} - U_n : \langle m, n\rangle \notin W_e\}.$$

As for $\Pi^0_1$ and $\Sigma^0_2$ classes,

$$S^*_{3,e} = \bigcup_{m=0}^{\infty} P^*_{2,\pi(e,m)}$$

where $W_{\pi(e,m)} = \{n : \langle n, m\rangle \in W_e\}$.

For an enumeration of $\Pi^\sigma_1$ classes, we define $P^*_{3,e} = \mathcal{X} - S^*_{3,e}$. However, we shall not pursue index set results for $\Pi^\sigma_1$ classes since almost all natural index sets we would consider are either $\Sigma^1_1$ complete or $\Pi^1_1$ complete and follow easily from previous results. Thus, in this paper, we shall develop index set results for 7 types of classes: $\{S_{1,e}\}_{e\in\omega}$, $\{P_{1,e}\}_{e\in\omega}$, $\{S_{2,e}\}_{e\in\omega}$, $\{P_{2,e}\}_{e\in\omega}$, $\{S^*_{2,e}\}_{e\in\omega}$, $\{P^*_{2,e}\}_{e\in\omega}$ and $\{S^*_{3,e}\}_{e\in\omega}$.

Next we need to consider the notions of unions, intersections, sums and products of the various classes defined above as well as develop the connections between simpler and more complicated classes. In any of the spaces $\mathcal{X}$ considered above, unions and intersections exist. However we will only define sums and products in the space $\omega^\omega$. Given $x = (x(0), x(1), \ldots)$ and $y = (y(0), y(1), \ldots)$ in $\omega^\omega$, we let $x \otimes y = (x(0), y(0), x(1), y(1), \ldots)$ and we let $0^\frown x = (0, x(0), x(1), \ldots)$ and $1^\frown y = (1, y(0), y(1), \ldots)$. Then given $A, B \subseteq \omega^\omega$, we define

$$A \otimes B = \{x \otimes y : x \in A \;\&\; y \in B\} \text{ and} \tag{9.1}$$
$$A \oplus B = \{0^\frown x : x \in A\} \cup \{1^\frown y : y \in B\}. \tag{9.2}$$

**Lemma 9.2.3.** *Let $\mathcal{X}$ be an effective topological space. For each of the seven types of classes $C_e$ discussed above, there are primitive recursive functions $\psi_u$, $\psi_i$, $\psi_s$ and $\psi_p$ such that for all $a$ and $b$,*

*(a)* $C_a \cup C_b = C_{\psi_u(a,b)}$. *(b)* $C_a \cap C_b = C_{\psi_i(a,b)}$.

*(c)* $C_a \oplus C_b = C_{\psi_s(a,b)}$. *(d)* $C_a \otimes C_b = C_{\psi_p(a,b)}$.

*Here for (c) and (d), we assume that $\mathcal{X}$ is either $\{0,1\}^\omega$ or $\omega^\omega$.*

*Proof.* We will just give the proofs in a few cases and leave the proofs of the remaining cases to the reader. To define $C_a \oplus C_b$ or $C_a \otimes C_b$ in the cases where $\mathcal{X} = \{0,1\}^\omega$ or $\mathcal{X} = \omega^\omega$, suppose we are given $U_n = I(\sigma)$ and $U_m = I(\tau)$ where $\sigma = (\sigma(0), \dots, \sigma(k))$ and $\tau = (\tau(0), \dots, \tau(l))$. Then we define $\pi_s(m,n) = \{U_r, U_t\}$ where $U_r = I((0, \sigma(0), \dots, \sigma(k)))$ and $U_t = I((1, \tau(0), \dots, \tau(l)))$. We let $\pi_p(m,n)$ equal the set of all $U_s$ such that $U_s = I(\gamma)$ where $\gamma = (\gamma(0), \dots, \gamma(2q+1))$ is such that $\forall i \leq k(\sigma(i) = \gamma(2i))$ and $\forall j \leq l(\gamma(2i+1) = \tau(i))$, where $q = max(k,l)$ .

(1) Then for the $\Sigma^0_1$ classes, we have the following:

$W_{\psi_u(a,b)} = W_a \cup W_b$;

$W_{\psi_i(a,b)} = \{\pi_i(m,n) : m \in W_a \ \& \ n \in W_b\}$;

$W_{\psi_s(a,b)} = \{U_x : \exists m \in W_a \ \& \ n \in W_b(x \in \pi_s(m,n))\}$;

$W_{\psi_p(a,b)} = \{U_y : \exists m \in W_a \ \& \ n \in W_b(y \in \pi_p(m,n)\}$.

Clearly $\psi_u$ and $\psi_i$ are primitive recursive for our space $\mathcal{X}$. Similarly if $\mathcal{X}$ is either $\{0,1\}^\omega$ or $\omega^\omega$, it is easy to see $\psi_s$ and $\psi_p$ are primitive recursive.

(2) For the $\Pi^0_1$ classes, we can use DeMorgan's laws. That is, for example, $P_{1,a} \cup P_{1,b} = \mathcal{X} - (S_{1,a} \cap S_{1,b})$, so that we can use the function $\psi_i$ from (1) as our $\psi_u$ in that case.

(3) For the $\Sigma^0_2$ classes, we can let

$$W_{\psi_u(a,b)} = \{\langle 2m, n\rangle : \langle m, n\rangle \in W_a\} \cup \{\langle 2m+1, n\rangle : \langle m, n\rangle \in W_b\}.$$

Again it is easy to see that $\psi_u$ is primitive recursive.

(4) For the $\Pi^0_2$ classes, we can use the corresponding functions for $\Sigma^0_2$ classes along with DeMorgan's laws as in (2).

(1)* For the strong $\Sigma^0_2$ classes,

$$W_{\psi_u(a,b)} = W_a \cap W_b \text{ and } W_{\psi_i(a,b)} = \{\pi_i(m,n) : m \in W_a \ \& \ n \in W_b\}$$

(2)* For the strong $\Pi^0_2$ classes, proceed as in (2).

(3)* For the $\Sigma^\sigma_1$ classes, proceed as in (3). □

There are many inclusions between the seven definability notions. For example, every $\Sigma^0_1$ class is also a $\Sigma^0_2$, a $\Pi^0_2$, a strong $\Sigma^0_2$ and a $\Sigma^\sigma_1$ class. Similarly, every $\Pi_1^0$ class is also $\Sigma_2^0$, $\Pi_2^0$, strong $\Pi_2^0$ and $\Sigma_1^\sigma$. Every $\Sigma_2^0$ class is also $\Sigma_1^\sigma$. Every strong $\Sigma_2^0$ class is also $\Sigma_2^0$ and every strong $\Pi_2^0$ class is also $\Pi_2^0$. Every strong $\Sigma_2^0$ class and every strong $\Pi_2^0$ class is also $\Sigma_1^\sigma$. Each of these inclusions is effective in the following sense.

**Lemma 9.2.4.** *For each inclusion $C \subset D$ given above, there is a primitive recursive function $\theta$ such that $C_e = D_{\theta(e)}$ for all $e$.*

*Proof.* We will just indicate how to proceed in a few cases.

(i) For the inclusion of $\Sigma_1^0$ classes in strong $\Sigma_2^0$ classes, uniformize the proof of Lemma 9.2.1. That is, let

$$m \in W_{\theta(e)} \iff \neg(\exists n, s < m)[U_m \subset U_n \ \&\ n \in W_{e,s}]$$

Then by the Lemma 9.2.2, $S_{2,\theta(e)} = \bigcup_{m \notin W_{\theta(e)}} U_m = S_{1,e}$.

(ii) For the inclusion of strong $\Sigma_2^0$ classes in $\Sigma_2^0$ classes, we will use the extra property of our effective topological spaces. That is, for any $j$ and $n$,

$$X - \overline{U_n} = \bigcup_{m \in \omega} \{U_m : U_m \cap U_n = \emptyset\}$$

and

$$U_j = \bigcup_{n \in \omega} \{\overline{U_n} : \overline{U_n} \subset U_j\}.$$

Hence, for any $e$,

$$S^*_{2,e} = \bigcup_{j \notin W_e} U_j = \bigcup_{j \notin W_e} \ \bigcup_{\substack{i \\ \overline{U_i} \subset U_j}} \ \bigcap_{\substack{n \\ U_i \cap U_n = \emptyset}} X - U_n$$

It can be checked that

$$S_{1,e} = S_{2,\theta(e)} = \bigcup_{m=0}^{\infty} \ \bigcap_{\substack{n \\ \langle m,n\rangle \in W_{\theta(e)}}} X - U_n,$$

where $\langle m,n\rangle \in W_{\theta(e)}$ if and only if

$$(\exists i)(\exists j)[m = \langle i,j\rangle \ \&\ (j \in W_{e,n} \vee \neg(\overline{U_i} \subset U_j) \ \vee \ (U_i \cap U_n \neq \emptyset))].$$

(ii) For the inclusion of $\Pi_1^0$ in $\Sigma_2^0$, just let $W_{\theta(e)} = \{\langle m,n\rangle : n \in W_e\}$.

The other inclusions follow by taking complements or have similar proofs. $\square$

## 9.3 Formal $\omega$-languages

In this section, we explain the connection between the seven types of classes which we will analyze in this paper and the notion of an $\omega$-language as the set of infinite words accepted by some computable automaton. For our purposes, a language will be a set of infinite strings based over a finite alphabet $X$. There is no loss in generality in assuming that $X$ is just $\{0,1\}$.

Recall that a computable automaton $M = [Z, z_0, Z_f, R]$ over $X$ consists of a nonempty initial segment $Z$ of $\omega$ (the *states* of $M$), a designated initial state $z_0 \in Z$, a computably enumerable set $Z_f \subset Z$ of accepting states, and a computably enumerable subset $R$ if $Z \times X \times Z$ of *transitions*. Given $M$, we shall consider the function $\Theta_R : Z \times X \to \omega \cup \{\infty\}$ where $\Theta_R(z,x) = card(\{z' : (z,x,z') \in R\})$. $M$ is *finitely branching* if $\Theta_R(z,x) \in \omega$ for all $z \in Z$ and $x \in X$. $M$ is *deterministic* if $\Theta_R(z,x) \in \{0,1\}$ for all $z \in Z$ and $x \in X$ so that $R$ may be considered as a partial function mapping $Z \times X$ into $Z$. Finally, a finitely branching computable automaton $M$ is *strictly computable* if $Z_f$ is computable and $\Theta_R$ is a computable function. Note that if $M$ is a deterministic computable automaton, then $R$ can be viewed a partial computable function. More generally, if $M$ is a deterministic computable automaton, we may consider the computably continuous mapping $\Phi_M : X^\omega \to Z^\omega$ defined by the machine $M$. Note that the graph of $\Phi_M$ is a strong $\Pi^0_2$ class if $M$ is computable and is a $\Pi^0_1$ class if $M$ is strictly computable.

We will say that a word $x \in X^\omega$ is $\alpha$*-accepted* according to a certain condition $(\alpha)$ provided that some run corresponding to the input word $x$ satisfies condition $(\alpha)$. There are four important types of acceptance conditions that have appeared in the literature that we will consider in this paper. Recall that a run of $M$ on input $x$ determines an infinite sequence $z_0, z_1, \ldots$ of states of $M$.

1. $x$ is 1-accepted by the run $r$ if at least one $z_i \in Z_f$. This notion is due to Hartmanis and Stearns [**?**].

2. $x$ is $e$-accepted by the run $r$ if every $z_i \in Z_f$. This notion is due to Landweber [**?**].

3. $x$ is *io*-accepted by the run $r$ if infinitely many $z_i$ are in $Z_f$. This notion is due to Büchi [14].

4. $x$ is *ae*-accepted by the run $r$ if all but finitely many $z_i \in Z_f$. This notion is due to Landweber [**?**].

Let $T_\alpha(M)$ be the set of infinite words $\alpha$-accepted by $M$. The following results are due to Cohen and Gold [**?**] and to Staiger [188].

**Theorem 9.3.1.** *For a strictly computable deterministic automaton M':*

*(i)* $T_e(M)$ *is a* $\Pi^0_1$ *class (and every* $\Pi^0_1$ *class is so represented.)*

*(ii)* $T_1(M)$ *is a* $\Sigma^0_1$ *class (and every* $\Sigma^0_1$ *class is so represented.)*

*(iii)* $T_{io}(M)$ *is a* $\Pi^0_2$ *class (and every* $\Pi^0_2$ *class is so represented.)*

*(iv)* $T_{ae}(M)$ *is a* $\Sigma^0_2$ *class (and every* $\Sigma^0_2$ *class is so represented.)*

We should point out that this theorem holds for the more general class of Turing machines, where one can move forward or backward on the input tape and do some calculations on work tapes. See [190, **?**] for details.

If $M$ is not strictly computable, then cases (ii) and (iv) are unchanged, but for cases (i) and (iii), we have two other notions of definability.

**Definition 9.3.2.** *For any language* $W \subset X^*$*,*

*(i)* $lim\ W = \{x : (\forall n)(x\lceil n \in W)\}$*;*

*(ii)* $W^\sigma = \{x : (\exists m)(\forall n > m)(x\lceil n \in W)\}$*;*

A class is in $lim\ \Sigma^0_1$ if and only if it is a strong $\Pi^0_2$ class. A class is in ${\Sigma^0_1}^\sigma$ if and only if it is an effective union of strong $\Pi^0_2$ classes.

We have the following theorem, see [188].

**Theorem 9.3.3.** *(i)** $W \in lim\ \Sigma^0_1$ *if and only if* $W = T_e(M)$ *for some deterministic computable automaton* $M$*.*

*(ii)** $W \in {\Sigma^0_1}^\sigma$ *if and only if* $W = T_{ae}(M)$ *for some deterministic computable automaton* $M$*.*

It is then easy to see that index set results for any of the six classes of $\omega$-languages considered above will follow from index sets results for one of our seven families of classes defined in section 2. Thus we shall only state our index set results for the seven families of classes in section 2 and leave it to the reader to translate these results to index set results for $\omega$-languages.

## 9.4 Index sets for cardinality

In this section, we determine the complexity of index sets for various classes with constraints on the cardinality of the class. For the rest of this paper, we shall restrict our attention to the spaces $\mathcal{X} = \{0,1\}^\omega$ and $\mathcal{X} = [0,1]$. Of course, the results for $\{0,1\}^\omega$ here will extend to the space $X^\omega$ for an arbitrary finite alphabet $X$.

The fundamental index set results for the property of being nonempty were given by Staiger [188]. Proofs of (i) and (iv) can also be found in [35] and a version of (i) for the classes in the space $[0,1]$ are given in Theorem 4.5 of [37].

**Theorem 9.4.1.** *(i)* $\{e : S_{1,e}$ *is nonempty*$\}$ *is* $\Sigma^0_1$ *complete.*

*(ii)* $\{e : P_{1,e}$ *is nonempty*$\}$ *is* $\Pi^0_1$ *complete.*

*(iii)* $\{e : S_{2,e}$ *is nonempty*$\}$ *is* $\Sigma^0_2$ *complete.*

*(iv)* $\{e : P_{2,e}$ *is nonempty*$\}$ *is* $\Sigma^1_1$ *complete.*

*(v)* $\{e : S^*_{2,e}$ *is nonempty*$\}$ *is* $\Sigma^0_2$ *complete.*

*(vi)* $\{e : P^*_{2,e}$ *is nonempty*$\}$ *is* $\Pi^0_2$ *complete.*

*(vii)* $\{e : S^*_{3,e}$ *is nonempty*$\}$ *is* $\Sigma^0_3$ *complete.*

*Proof.* It is not hard to see that each family has a definition of the appropriate complexity. For example,

$$S_{1,e} \neq \emptyset \iff (\exists n \in W_e)(U_n \neq \emptyset)$$

so that $\{e : S_{1,e}$ is nonempty$\}$ is $\Sigma^0_1$. (Recall that for an effective topological space, we assume that the $\{\langle m, n\rangle : U_m = U_n\}$ is primitive recursive. For the spaces of this theorem, each basic open set has a unique representation, by a finite string or by a pair of rationals, so that this problem is trivial.) Similarly, by the compactness of $\{0,1\}^\omega$ or $[0,1]$,

$$S_{1,e} = \mathcal{X} \iff (\exists n_1, \ldots, n_k \in W_e)(\mathcal{X} \subseteq U_{n_1} \cup \cdots \cup U_{n_k})$$

so $\{e : S_{1,e} = \mathcal{X}\}$ is $\Sigma^0_1$ and hence $\{e : P_{1,e}$ is nonempty$\}$ is $\Pi^0_1$. Relativizing these results to a $\mathbf{0}'$-oracle then establishes the complexity bounds for strong $\Sigma^0_2$ and strong $\Pi^0_2$ classes. We can use the effective union properties to see that

$$S_{2,e} \neq \emptyset \iff (\exists n) P_{1,\pi(e,m)} \neq \emptyset$$

and similarly

$$S^*_{3,e} \neq \emptyset \iff (\exists n) P^*_{2,\pi(e,m)} \neq \emptyset.$$

The completeness results for (i), (ii), and (iv) were established in [35]. We note also that completeness results for $\{0,1\}^\omega$ can be extended to the real interval using Lemma 4.4 of [37] which shows that each space can be effectively embedded in the other by means of a primitive recursive function on indices.

To introduce a technique which we will use later, we shall show the completeness of (iii) follows from (ii). Let $A$ be $\Sigma^0_2$. Then there is a uniformly $\Pi^0_1$ sequence of sets $B_e$ such that

$$a \in A \iff (\exists e)(a \in B_e).$$

It follows from (ii) that there is computable function $\phi$ such that

$$a \in B_e \iff P_{1,\phi(a,e)} \neq \emptyset.$$

But then

$$a \in A \iff (\exists e)(P_{1,\phi(a,e)} \neq \emptyset) \iff \bigcup_e P_{1,\phi(a,e)} \neq \emptyset \iff S_{2,\psi(a)} \neq \emptyset$$

where

$$W_{\psi(a)} = \{\langle m, n\rangle : n \in W_{\phi(a,n)}\}.$$

This shows that the family in (iii) is $\Sigma_2^0$ complete as desired. One can establish the completeness result for (vii) from the completeness result for (vi) in a similar manner.

The completeness results for (v), (vi) and (vii) easily follow from the completeness results of (i),(ii), and (iii) respectively by the process of relativization. That is, as noted above, strong $\Sigma_2^0$, strong $\Pi_2^0$ and $\Sigma_1^\sigma$ classes can be viewed as relativized versions with respect to an $\mathbf{0}'$ oracle of $\Sigma_1^0$, $\Pi_1^0$ and $\Sigma_2^0$ classes respectively. Hence a relativization with respect to an $\mathbf{0}'$ oracle of the proofs of parts (i), (ii) and (iii) will demonstrate that the complexity of the index sets in parts (v), (vi) and (vii) are just the complexity of the classes (i), (ii) and (iii) relative to oracle $\mathbf{0}'$. It easily follows that the index set in (v), (vi) and (vii) are respectively $\Sigma_2^0$, $\Pi_2^0$ and $\Sigma_3^0$ complete. □

Next we consider index sets for finite cardinality.

**Theorem 9.4.2.** *Let $\mathcal{X}$ be either $\{0,1\}^\omega$ or $[0,1]$. For any $k > 1$,*

*(i)* $\{e : card(S_{1,e}) \geq k\}$ *is* $\Sigma_1^0$ *complete.*

*(ii)* $\{e : card(P_{1,e}) \geq k\}$ *is* $\Sigma_2^0$ *complete.*

*(iii)* $\{e : card(S_{2,e}) \geq k\}$ *is* $\Sigma_2^0$ *complete.*

*(iv)* $\{e : card(P_{2,e}) \geq k\}$ *is* $\Sigma_1^1$ *complete.*

*(v)* $\{e : card(S^*_{2,e}) \geq k\}$ *is* $\Sigma_2^0$ *complete.*

*(vi)* $\{e : card(P^*_{2,e}) \geq k\}$ *is* $\Sigma_3^0$ *complete.*

*(vii)* $\{e : card(S^*_{3,e}) \geq k\}$ *is* $\Sigma_3^0$ *complete.*

*Proof.* (i) This follows from Theorem 9.4.1 (i) since any open set has cardinality of the continuum if and only if it is nonempty. Part (ii) was part proved in [37].

(iii) First observe that

$$card(S_{2,e}) \geq k \iff (\exists n) card(\cup_{m=0}^n P_{1,\pi(e,m)}) \geq k.$$

However by Lemma 9.2.3, there is a computable function $g$ such that $P_{1,g(e,n)} = \cup_{m=0}^n P_{1,\pi(e,m)}$ so that

$$card(S_{2,e}) \geq k \iff (\exists n) card(P_{1,\pi(g(e,n))}) \geq k.$$

Thus (iii) is $\Sigma_2^0$ because (ii) is $\Sigma_2^0$. By Lemma 9.2.4, there is a computable function $f$ such that for all $n$, $P_{1,n} = S_{2,f(n)}$. By (ii), for any $\Sigma_2^0$ set $A$, there is a computable $h$ such that $x \in A \iff h(x) \in \{e : card(P_{1,e}) \geq k\}$. Hence $x \in A \iff f(h(x)) \in \{e : card(S_{2,e}) \geq k\}$ so that (iii) is $\Sigma_2^0$ complete. We note the one can establish (vii) from (vi) by a similar argument.

(iv) This set is certainly $\Sigma_1^1$ and the completeness follows from the fact that

$$P \neq \emptyset \iff card(kP) \geq k$$

where $kP$ is just the disjoint union of $k$ copies of $P$.

The remaining three cases are just relativizations of (i), (ii), and (iii). □

We note that the complexity results of Theorem 9.4.2 are uniform in $k$. That is, for example, $\{\langle e,k\rangle : card(S_{1,e}) \geq k\}$ is $\Sigma^0_1$. We can use this in our next result, on infinite cardinality.

**Theorem 9.4.3.** *Let $\mathcal{X}$ be either $\{0,1\}^\omega$ or $[0,1]$.*

*(i) $\{e : card(S_{1,e}) \geq \aleph_0\}$ is $\Sigma^0_1$ complete.*

*(ii) $\{e : card(P_{1,e}) \geq \aleph_0\}$ is $\Pi^0_3$ complete.*

*(iii) $\{e : card(S_{2,e}) \geq \aleph_0\}$ is $\Pi^0_3$ complete.*

*(iv) $\{e : card(P_{2,e}) \geq \aleph_0\}$ is $\Sigma^1_1$ complete.*

*(v) $\{e : card(S^*_{2,e}) \geq \aleph_0\}$ is $\Sigma^0_2$ complete.*

*(vi) $\{e : card(P^*_{2,e}) \geq \aleph_0\}$ is $\Pi^0_4$ complete.*

*(vii) $\{e : card(S^*_{3,e}) \geq \aleph_0\}$ is $\Pi^0_4$ complete.*

*Proof.* Part (i) follows as in Theorem 9.4.2 and part (ii) is proved in [37].

(iii) We show that $\{e : card(S_{2,e}) < \aleph_0\}$ is $\Sigma^0_3$ complete. Let $g$ and $f$ be the computable functions defined in Theorem 9.4.2 such that $P_{1,g(e,n)} = \cup^n_{m=0} P_{1,\pi(e,m)}$ and $P_{1,e} = S_{2,f(e)}$. To see that $\{e : card(S_{2,e}) < \aleph_0\}$ is $\Sigma^0_3$ observe that $card(S_{2,e}) < \aleph_0$ if and only if

$$(\exists n)[(\forall m \leq n)(P_{1,\pi(m,e)} < \aleph_0) \mathbin{\&} (\forall p > n)(P_{1,\pi(p,e)} \subseteq P_{1,g(e,n)} = \bigcup_{m=0}^{n} P_{1,\pi(m,e)})]. \tag{9.3}$$

It is easy to see that the predicate $S_{1,e} \supset S_{1,f}$ is in $\Pi^0_2$ since $S_{1,e} \supseteq S_{1,f}$ if and only if

$$(\forall m)(\forall n)[(n \in W_f \mathbin{\&} \overline{U_m} \subset U_m) \Rightarrow (\exists n_1, \ldots, n_k \in W_e)(U_n \subseteq U_{n_1} \cup \cdots \cup U_{n_k})].$$

By taking complements it is easy to see that the predicate $P_{1,e} \subseteq P_{1,f}$ is also $\Pi^0_2$. It then easily follows that the expression in (9.3) is $\Sigma^0_3$. (Note that we could also have appealed here to the uniformity of Theorem 9.4.2.) For completeness, it follows from (ii) that for any $\Sigma^0_3$ set $A$, there is a computable function $k$ such that $x \in A \iff k(x) \in \{e : card(P_{1,e}) < \aleph_0\}$ and hence $x \in A \iff f(k(x)) \in \{e : card(S_{2,e}) < \aleph_0\}$. Thus $\{e : card(S_{2,e}) < \aleph_0\}$ is complete for $\Sigma^0_3$ sets.

(iv) The completeness follows from the fact that $P_{2,e} \otimes \{0,1\}^\omega$ is infinite if and only if $P_{2,e}$ is nonempty.

The remaining cases follow by relativization. □

The complexity results for uncountable cardinality are the same for each type of class expect for $\Sigma^0_1$ classes and strong $\Sigma^0_2$ classes.

**Theorem 9.4.4.** *Let $\mathcal{X}$ be either $\{0,1\}^\omega$ or $[0,1]$.*

*(i) $\{e : card(S_{1,e}) > \aleph_0\}$ is $\Sigma^0_1$ complete and $\{e : card(S_{1,e}) = \aleph_0\} = \emptyset$.*

*(ii)* $\{e : card(S^*_{2,e}) > \aleph_0\}$ *is* $\Sigma^0_2$ *complete and* $\{e : card(S^*_{2,e}) = \aleph_0\} = \emptyset$.

*(iii) For each of the remaining five types of classes* $C_e$*,* $\{e : card(C_e) > \aleph_0\}$ *is* $\Sigma^1_1$ *complete and* $\{e : card(C_e) = \aleph_0\}$ *is* $\Pi^1_1$ *complete.*

*Proof.* Part (i) holds by the proof of Theorem 9.4.1 since $S_{1,e}$ is nonempty if and only if $S_{1,e}$ has the cardinality of the continuum. Part (ii) follows from part (i) by relativization.

Theorem 4.5 of [37] proves that $\{e : card(P_{1,e}) > \aleph_0\}$ is $\Sigma^1_1$ complete and that $\{e : card(P_{1,e}) = \aleph_0\}$ is $\Pi^1_1$ complete. The remaining cases can be proved from these completeness results by the same type of argument that we used to prove part (iii) from part (ii) in Theorem 9.4.2. That is, in general, the upper bound on the complexity is given by the fact that any Borel set $K$ is uncountable if and only if it has a perfect subset. This means that $K$ is uncountable if and only if there exists an embedding of $\mathcal{X}$ into $K$, which can be coded by a map from basic open sets to basic open sets. Thus $K$ is uncountable if and only if

$$(\exists f)(\forall m)(\forall n)[U_{f(n)} \subset K \ \& \ (U_m \subset U_n \to U_{f(m)} \subset U_{f(n)}) \ \& \ (U_m \neq U_n \to U_{f(m)} \neq U_{f(n)})].$$

Finally, the completeness in the remaining cases easily follows form the completness results for $\{e : card(P_{1,e}) > \aleph_0\}$ and $\{e : card(P_{1,e}) = \aleph_0\}$. □

Next we consider the complexity of having a given number of computable members.

**Theorem 9.4.5.** *Let* $\mathcal{X}$ *be either* $\{0,1\}^\omega$ *or* $[0,1]$ *and let* $k \geq 1$.

*(i)* $\{e : S_{1,e}$ *has* $\geq k$ *computable members*$\}$ *is* $\Sigma^0_1$ *complete.*

*(ii)* $\{e : S^*_{2,e}$ *has* $\geq k$ *computable members*$\}$ *is* $\Sigma^0_2$ *complete.*

*(iii) For any* $k \geq 1$ *and for each of the remaining five types of classes* $C_e$*,* $\{e : C_e$ *has* $\geq k$ *computable members*$\}$ *is* $\Sigma^0_3$ *complete.*

*Proof.* Part (i) holds by the proof of Theorem 9.4.2 since any nonempty open set contains infinitely many recursive members. Then part (ii) follows from part (i) by relativization.

The fact that $\{e : P_{1,e}$ has $\geq k$ computable members$\}$ is $\Sigma^0_3$ complete is in Theorem 4.7 of [37]. Theorem 4.8 of [37] proves that $\{e : P^*_{2,e}$ has $\geq k$ computable members$\}$ is $\Sigma^0_3$ complete when $k = 1$ and the proof can easily be extended to cover the case when $k > 1$. One can then use these completeness results to prove that $\{e : S_{2,e}$ has $\geq k$ computable members$\}$ and $\{e : S_{3,e}$ has $\geq k$ computable members$\}$ are $\Sigma^0_3$ complete by the same type of argument that we used to prove part (iii) of Theorem 9.4.2 from part (ii) of Theorem 9.4.2.

Observe that in $\{0,1\}^\omega$, $P_{2,e}$ has a computable member if and only if

$$(\exists e)[e \in Tot \ \& \ (\forall m)(\exists n)(\langle m, n\rangle \in W_e \ \& \ \phi_e \in U_n)].$$

Here $Tot = \{e : \phi_e$ is total$\}$ which is a complete $\Pi^0_2$ set. Of course, for $n > 0$, $U_n = I(\sigma_{n-1})$ for the finite string $\sigma_{n-1}$ and $\phi_e \in U_n$ just means that $(\forall i <$

$lh(\sigma_{n-1}))(\phi_e(i) = \sigma_n(i))$, which is $\Sigma^0_1$. This is easily modified to obtain the result for $\geq k$ recursive members. Finally, the completeness follows from the completeness of (ii). □

**Theorem 9.4.6.** *Let $\mathcal{X}$ be either $\{0,1\}^\omega$ or $[0,1]$.*

*(i) $\{e : S_{1,e}$ has $\aleph_0$ computable members$\}$ is $\Sigma^0_1$ complete.*

*(ii) $\{e : S^*_{2,e}$ has $\aleph_0$ computable members$\}$ is $\Sigma^0_2$ complete.*

*(iii) For each of the remaining five types of classes $C_e$,*
*$\{e : C_e$ has $\aleph_0$ recursive members$\}$ is $\Pi^0_4$ complete.*

*Proof.* Part (i) holds by the proof of Theorem 9.4.2 since every nonempty open set has $\aleph_0$ computable members. Part (ii) follows from part (i) by relativization.

The fact that $\{e : P_{1,e}$ has $\aleph_0$ computable members$\}$ is $\Pi^0_4$ complete is proved in Theorem 4.7 of [37]. In general, we have

$$C_e \text{ has } \aleph_0 \text{ recursive members} \iff (\forall k)(C_e \text{ has } \geq k \text{ recursive members})$$

so that each index set is $\Pi^0_4$. Then the completeness result for the remaining cases follows from Theorem 9.2.4 and the $\Pi^0_4$ completeness of $\{e : P_{1,e}$ has $\aleph_0$ computable members$\}$. □

## 9.5 Index sets for measure

In this section, we consider the complexity of having a certain Lebesgue measure. Let $\mu(P)$ denote the measure of a class $P$ in either $\{0,1\}^\omega$ or in $[0,1]$. We begin with the results from [37] on the measure of $\Pi^0_1$ classes.

**Theorem 9.5.1.** *Let $\mathcal{X}$ be either $\{0,1\}^\omega$ or $[0,1]$ and let $r$ be a recursive real number.*

*(i) For $r > 0$, $\{e : \mu(P_{1,e}) < r\}$ is $\Sigma^0_1$ complete.*

*(ii) For $r < 1$, $\{e : \mu(P_{1,e}) \leq r\}$ is $\Pi^0_2$ complete.*

Relativizing this with oracle $\mathbf{0}'$, we obtain the following.

**Theorem 9.5.2.** *Let $\mathcal{X}$ be either $\{0,1\}^\omega$ or $[0,1]$ and let $r$ be a recursive real number.*

*(i) For $r > 0$, $\{e : \mu(P^*_{2,e}) < r\}$ is $\Sigma^0_2$ complete.*

*(ii) For $r < 1$, $\{e : \mu(P^*_{2,e}) \leq r\}$ is $\Pi^0_3$ complete.*

From the two previous results, we obtain the following results for $\Sigma^0_1$ and strong $\Sigma^0_2$ classes.

**Theorem 9.5.3.** *Let $\mathcal{X}$ be either $\{0,1\}^\omega$ or $[0,1]$ and let $r$ be a recursive real number.*

(i) *For* $r > 0$, $\{e : \mu(S_{1,e}) < r\}$ *is* $\Sigma^0_2$ *complete.*

(ii) *For* $r < 1$, $\{e : \mu(S_{1,e}) \leq r\}$ *is* $\Pi^0_1$ *complete.*

(iii) *For* $r > 0$, $\{e : \mu(S^*_{2,e}) < r\}$ *is* $\Sigma^0_3$ *complete.*

(iv) *For* $r < 1$, $\{e : \mu(S^*_{2,e}) \leq r\}$ *is* $\Pi^0_2$ *complete.*

We next consider $\Pi^0_2$ and $\Sigma^0_2$ classes.

**Theorem 9.5.4.** *Let* $\mathcal{X}$ *be either* $\{0,1\}^\omega$ *or* $[0,1]$ *and let* $r$ *be a recursive real number.*

(i) *For* $r > 0$, $\{e : \mu(P_{2,e}) < r\}$ *is* $\Sigma^0_2$ *complete.*

(ii) *For* $r < 1$, $\{e : \mu(P_{2,e}) \leq r\}$ *is* $\Pi^0_3$ *complete.*

*Proof.* (i) For the upper bound on the complexity, use the fact that $P_{2,e} = \cap_m S_{1,\pi(m,e)}$ for some recursive function $\pi$. Then

$$\mu(P_{2,e}) < r \iff (\exists m)\mu(S_{1,\pi(m,e)}) < r$$

The completeness result follows from Theorem 9.5.3.

(ii) The complexity bound follows from the fact that

$$\mu(P_{2,e}) \leq r \iff (\forall q > r)\mu(P_{2,e}) < q$$

where $q$ denotes a rational number. The required completeness follows from Theorem 9.5.2. $\square$

This immediately gives the corresponding result for $\Sigma^0_2$ classes.

**Theorem 9.5.5.** *Let* $\mathcal{X}$ *be either* $\{0,1\}^\omega$ *or* $[0,1]$ *and let* $r$ *be a recursive real number.*

(i) *For* $r > 0$, $\{e : \mu(S_{2,e}) < r\}$ *is* $\Sigma^0_3$ *complete.*

(ii) *For* $r < 1$, $\{e : \mu(S_{2,e}) \leq r\}$ *is* $\Pi^0_2$ *complete.*

Finally, the result for $\Sigma^\sigma_1$ classes follows by relativization from Theorem 9.5.5.

**Theorem 9.5.6.** *Let* $\mathcal{X}$ *be either* $\{0,1\}^\omega$ *or* $[0,1]$ *and let* $r$ *be a recursive real number.*

(i) *For* $r > 0$, $\{e : \mu(S^*_{3,e}) < r\}$ *is* $\Sigma^0_4$ *complete.*

(ii) *For* $r < 1$, $\{e : \mu(S^*_{3,e}) \leq r\}$ *is* $\Pi^0_3$ *complete.*

## 9.6 Verification

The *Verification Problem* for two sets of classes $\{K_i : i \in I\}$ and $\{M_j : j \in J\}$ is $\{\langle i,j\rangle : K_i \subset M_j\}$. The verification problem has been studied by Klarlund [96], Staiger [188] and others. Staiger solved the verification problem for all combinations of the seven types of classes considered in this paper. More generally, one can ask questions about the difference between two given classes $K$ and $M$. For example, "Does $K - M$ contain any recursive elements?", "What is the cardinality of $K - M$?", "What is the measure of $K - M$?", and so on. One goal of this paper is to address these general verification-type questions for the various classes defined above.

In this section, we will analyze various properties of the difference set $P-Q$. Of course $P - Q$ is empty if and only $P \subset Q$, but it is also interesting to see whether $P-Q$ might have cardinality $\leq k$ for some $k$, or might have no recursive members. We will generally restrict our study to four cases: (1) differences of $\Pi^0_1$ classes; (2) differences of $\Sigma^0_2$ classes; (3) differences of strong $\Pi^0_2$ classes; and (4) differences of $\Sigma^\sigma_1$ classes. Notice that the difference of $\Sigma^0_1$ classes is the same thing as the difference of $\Pi^0_1$ classes, and similarly for $\Sigma^0_2$ and $\Pi^0_2$. The first theorem of this section gives the complexity of the verification problem for all combinations of the classes studied. We begin with a fundamental theorem of Staiger [188].

**Theorem 9.6.1** (Staiger)**.** *(i)* $\{\langle a,b\rangle : S_{1,a} \subset P_{1,b}\}$ *is* $\Sigma^0_1$ *complete;*

*(ii)* $\{\langle a,b\rangle : P_{1,a} \subset S_{1,b}\}$ *is* $\Pi^0_1$ *complete;*

*(iii)* $\{\langle a,b\rangle : P_{1,a} \subset P_{1,b}\}$ *and* $\{\langle a,b\rangle : S_{2,a} \subset P_{2,b}\}$ *are* $\Pi^0_2$ *complete;*

*(iv)* $\{\langle a,b\rangle : P^*_{2,a} \subset S_{1,b}\}$ *is* $\Sigma^0_2$ *complete;*

*(v)* $\{\langle a,b\rangle : P^*_{2,a} \subset P_{1,b}\}$ *and* $\{\langle a,b\rangle : S^*_{3,a} \subset P_{2,b}\}$ *are* $\Pi^0_3$ *complete; and*

*(vi)* $\{\langle a,b\rangle : P_{2,a} \subset P_{1,b}\}$ *and* $\{\langle a,b\rangle : P_{2,a} \subset S_{1,b}\}$ *are* $\Pi^1_1$ *complete.*

Next we consider the complexity of having a finite difference.

**Theorem 9.6.2.** *Let* $\mathcal{X}$ *be either* $\{0,1\}^\omega$ *or* $[0,1]$*. For any* $k \geq 1$*,*

*(i)* $\{\langle a,b\rangle : card(P_{1,a} - P_{1,b}) \leq k\}$ *is* $\Pi^0_2$ *complete.*

*(ii)* $\{\langle a,b\rangle : card(P_{2,a} - P_{2,b})) \leq k\}$ *is* $\Pi^1_1$ *complete.*

*(iii)* $\{\langle a,b\rangle : card(P^*_{2,a} - P^*_{2,b}) \leq k\}$ *is* $\Pi^0_3$ *complete.*

*(iv)* $\{\langle a,b\rangle : card(S^*_{3,a} - S^*_{3,b}) \leq k\}$ *is* $\Pi^0_3$ *complete.*

*Proof.* In each case, the completeness follows from Theorem 9.4.2. Thus we need only see that index sets have the appropriate complexity.

(i) To see that (i) is $\Pi^0_2$, we claim that

$$card(P_{1,a} - P_{1,b}) \leq k \iff (\forall e)[P_{1,b} \cap P_{1,e} = \emptyset \;\rightarrow\; card(P_{1,a} \cap P_{1,e}) \leq k].$$

Certainly if the condition is false, then $card(P_{1,a}-P_{1,b})>k$. On the other hand, suppose that $card(P_{1,a}-P_{1,b})>k$. Then there are $k+1$ elements $x_0,x_1,\ldots,x_k$ in $P_{1,a}-P_{1,b}$. For each $i$, there is a basic open set $U_i$ such that $x_i\in U_i$ and $U_i\cap P_{1,b}=\emptyset$. Then $P_e=U_0\cup\cdots\cup U_k$ contradicts the condition.

(ii) As above, we claim that

$$card(P_{2,a}-P_{2,b})\le k) \iff (\forall e)[P_{2,b}\cap P_{2,e}=\emptyset \ \rightarrow \ card(P_{2,a}\cap P_{2,e})\le k].$$

The key here is that $\mathcal{X}-P_{2,b}=S_{2,b}=\cup_m P_{1,\pi(e,m)}$ so that if $x\in P_{2,a}-P_{2,b}$, then $x\in P_{1,\pi(e,m)}$ for some $m$ and $P_{1,\pi(e,m)}\cap P_{2,b}=\emptyset$. Hence if $P_{2,a}-P_{2,b}$ has $k+1$ elements, we can obtain a finite union $P$ of $\Pi^0_1$ classes such that $P\cap P_{2,b}=\emptyset$ and $card(P\cap P_{2,a})>k$.

(iii) This is similar to (i) since $P^*_{2,b}$ is a closed set.

(iv) This is similar to (ii) since $S^*_{3,a}$ is an effective union of strong $\Pi^0_2$ classes. □

From this, we can go to an arbitrary finite difference.

**Theorem 9.6.3.** *Let $\mathcal{X}$ be either $\{0,1\}^\omega$ or $[0,1]$.*

*(i) $\{\langle a,b\rangle : card(P_{1,a}-P_{1,b})<\aleph_0\}$ is $\Sigma^0_3$ complete.*

*(ii) $\{\langle a,b\rangle : card(P_{2,a}-P_{2,b}))<\aleph_0\}$ is $\Pi^1_1$ complete.*

*(iii) $\{\langle a,b\rangle : card(P^*_{2,a}-P^*_{2,b})<\aleph_0\}$ is $\Sigma^0_4$ complete.*

*(iii) $\{\langle a,b\rangle : card(S^*_{3,a}-S^*_{3,b})<\aleph_0\}$ is $\Sigma^0_4$ complete.*

*Proof.* In each case, the completeness follows from Theorem 9.4.3 and the upper bound on the complexity follows from the uniformity of the proof of 9.6.2. □

Next we consider countable differences.

**Theorem 9.6.4.** *Let $\mathcal{X}$ be either $\{0,1\}^\omega$ or $[0,1]$. For each of the seven types of classes $C_e$, $\{\langle a,b\rangle : card(C_a-C_b))\le\aleph_0\}$ is $\Pi^1_1$ complete.*

*Proof.* This follows easily from Theorem 9.4.4. □

Finally, we look at the cardinality of computable elements.

**Theorem 9.6.5.** *Let $\mathcal{X}$ be either $\{0,1\}^\omega$ or $[0,1]$. For any $k\ge 0$ and for each of the seven types of classes $C_e$,*

*(i) $\{\langle a,b\rangle : C_a-C_b$ has $\le k$ computable members$\}$ is $\Pi^0_3$ complete.*

*(ii) $\{\langle a,b\rangle : C_a-C_b$ has $<\aleph_0$ computable members$\}$ is $\Sigma^0_4$ complete.*

*Proof.* (i) In each case, the completeness follows from Theorem 9.4.5. For the upper bounds on the complexity, we claim that in general, for a given type of classes $C_e$, $C_a - C_b$ has $\leq k$ computable members if and only if

$$(\forall e)[C_b \cap P_{1,e} = \emptyset \ \rightarrow \ card(C_a \cap P_{1,e}) \leq k].$$

The key here is that if we have $k+1$ recursive elements $x_0, \ldots, x_k$ in the difference, then they compose a $\Pi^0_1$ class. Here we use Lemmas 9.2.3 and 9.2.4 to put each set $P_{1,e}$ in the same definability family and to compute the intersection and test whether a set is empty.

(ii) The upper bounds on the complexity follow from the uniformity of (i) and the completeness follows from Theorem 9.4.6. □

%bibitemSW77 K. Wagner and L. Staiger, Springer-Verlag (1977), 532-537.

# Chapter 10

# Graphs

There are several combinatorial problems associated with computable graphs. These include the graph coloring problem, the problems of Hamiltonian and Euler circuits, the vertex partition problem, and various matching or marriage problems. In each case, the set of solutions to any such problem may be represented by a $\Pi_1^0$ class. To obtain a bounded $\Pi_1^0$ class, it is sometimes necessary to assume that each vertex of the computable graph has finite degree and to obtain a c. b. $\Pi_1^0$ class, it is sometimes necessary to assume that the graph is *highly computable*, that is, the set of vertices joined to vertex $v$ can be computed from $v$.

For the reverse direction, there are a variety of results. In each case, the set of solutions can represent an arbitrary $\Pi_1^0$ class of separating sets. For the graph-coloring problem, Remmel [161] showed that the 3-coloring problem for highly computable graphs can represent an arbitrary c. b. $\Pi_1^0$ class. Manaster and Rosenstein [120] showed that the set of surjective marriages in a symmetically highly computable society can likewise represent an arbitrary c. b. $\Pi_1^0$ class. On the other hand, Remmel [160] showed that this last result does not hold when each person knows at most two other people; this problem is related to the the Schroder-Bernstein theorem, where one tries to construct an isomorphism between two sets given injections in each direction.

For each section, we begin by giving a list of the problems and the required definitions together with some of the history of each problem. Next we explain (in varying detail) how to prove that the set of solutions to any such problem can be represented by a computably bounded $\Pi_1^0$ class. Then we apply the results of Chapters IV and V to obtain corollaries which apply to the set of solutions of any such problem. Conversely we also consider for each problem, whether the set of solutions to such a problem can represent any *c.b.* $\Pi_1^0$ class. In each case, we show that the set of solutions to such a problem can represent the class of separating sets of any two disjoint *c.e.* sets. Then we apply the results of Chapters IV and V to obtain corollaries which give the existence of "pathological" problems of each type. Next we consider index sets for such problems using the methods of Chapter VI. Then we examine the reverse mathematics of such problems as in

Chapter VII. Finally, we look at complexity-theoretic versions of some of the problems.

## 10.1 Matching problems

A *computable society* $S = (B, G, K)$ consists of disjoint computable sets $B$, the set of boys, and $G$, the set of girls, and a computable binary relation $K \subseteq B \times G$. Here $K(b, g)$ means $b$ knows $g$. The solutions in this case are the set of *marriages*, or *matchings*, that is, 1:1 maps $f : B \to G$ such that $K(b, f(b))$ holds for all $b$. For any subset $B'$ of $B$, let $K(B') = \{g : (\exists b \in B')K(b, g)\}$. Marshall Hall [76] extended the classical Philip Hall Theorem to infinite societies and proved that, for any countable society $S = (B, G, K)$, if every boy knows only finitely many girls and, for any finite subset $B' \subseteq B$, $|B'| \leq |K(B')|$, then there is a marriage for $S$. We say that a computable society $S = (B, G, K)$ is *highly computable* if there is a partial computable function $k : B \to \omega$ such that, for each $b \in B$, $k(b)$ equals the cardinality of $K(b)$. We say that $S$ is *symmetrically highly computable* if there is also a partial computable function $\overline{k}$ such that, for each $g \in G$, $\overline{k}(g)$ is the cardinality of the set of boys which know $g$.

The problems which we consider are:

**(i)** The general problem of finding a marriage in a highly computable society $S$,

**(ii)** the surjective matching problem, that is, finding a marriage $f : B \to G$ which is both 1:1 and onto in a symmetrically highly computable society $S$, and

**(iii)** the surjective matching problem, where each person knows at most two other people in a symmetrically highly computable society $S$.

Problems (i) and (ii) were analyzed by Manaster and Rosenstein in [120, 121], who showed that the set of marriages in case (i) and (ii) is always a *c.b.* $\Pi^0_1$ class, but does not always contain a computable element. Moreover, Manaster, Rosenstein showed that in case (ii), the set of surjective marriages can represent an arbitrary *c.b.* $\Pi^0_1$ class. We note that problem (iii) contains a computable version of Banach's strengthening of the Schroder-Bernstein theorem, which was shown to be noneffective by Remmel [160]. That is, suppose we take 1:1 computable functions with computable ranges $f : B \to G$ and $g : G \to B$ where $B$ and $G$ are computable sets. Then we can form a highly computable society $S = (B, G, K)$, where $K(x, y)$ holds if and only if $f(x) = y$ or $g(y) = x$. For such a society $S$, the only surjective marriages $h$ arise from some partition $B = B_1 \cup B_2$, where $h = f\lceil B_1 \cup g^{-1}\lceil B_2$, and the existence of such marriages are guaranteed by Banach's result. (See [160] for details.) It was shown by Remmel in [161] that the set of surjective marriages in case (iii) cannot represent an arbitrary *c.b.* $\Pi^0_1$ class in contrast to the Manaster-Rosenstein result for case (ii).

In each case, the set of solutions to such a problem can be represented by a $\Pi^0_1$ class [120, 121].

**Theorem 10.1.1.** *For any computable instance of each of the three matching problems described above, the set of solutions can be represented by a $\Pi^0_1$ class. If the given graph is highly computable, then the class is computably bounded.*

*Proof.* We may assume that $B$ is the set of even numbers and $G$ is the set of odd numbers. In (1), a marriage is simply a 1:1 map $g : B \to G$ such that $(b, g(b)) \in K$ for all $b \in B$. We can represent $g$ by a map $x_g : \mathbb{N} \to \mathbb{N}$ by letting $x_g(i) = 2g(2i) + 1$. Thus the $\Pi^0_1$ class $P \subset \mathbb{N}^{\mathbb{N}}$ which represents the set of solutions is given by

$$x \in P \iff (\forall i)(2i, 2x(i) + 1) \in K \;\&\; (\forall i, k)(x(i) = x(k) \to i = k).$$

If $S$ is highly computable, then given $i$, we can compute the finite set $G_i = \{j : (2i, 2j + 1) \in K\}$. Since $x_g(i) \in G_i$, this shows that $P$ is computably bounded.

For problems (ii) and (iii), the solution is a pair of functions, one from $B$ into $G$ and one from $G$ into $B$, which are inverses of each other. This matching can be represented by a single function from $\mathbb{N}$ to $\mathbb{N}$ and the set of solutions will again be a $\Pi^0_1$ class, and will be c. b. if $S$ is highly computable. □

We can derive a number of immediate corollaries to Theorem 10.1.1.

**Theorem 10.1.2.** *For each highly computable society $S$ and matching problem of type (i), (ii), or (iii), the following hold.*

- *(a) If $S$ has a solution, then $S$ has a solution in some c. e. degree.*
- *(b) If $S$ has a solution, then $S$ has solutions $s_1$ and $s_2$ such that any function computable in both $s_1$ and $s_2$ is recursive.*
- *(c) If $S$ has a solution but only has countably many solutions, then $S$ has a computable solution.*
- *(d) If $S$ has only finitely many solutions, then each solution is computable.*
- *(e) If $S$ has a solution but has no computable solution, then for any countable sequence of nonzero degrees $\{\mathbf{a}_i\}$, $S$ has continuum many solutions $s$ which are mutually Turing incomparable and such that the degree of $s$ is incomparable with each $\mathbf{a}_i$.*

Next we consider the reverse direction of this correspondence. That is, given an arbitrary c. b. $\Pi^0_1$ class $P$, is there a matching problem of a given type such that $P$ represents the set of matchings?

**Theorem 10.1.3.** *[[120]] The problem of finding a surjective marriage in a computable society can represent an arbitrary bounded $\Pi^0_1$ class and the problem of finding a surjective marriage in a symmetrically highly computable society can represent an arbitrary c. b. $\Pi^0_1$ class.*

*Proof.* Let $P$ be the set of infinite paths through a computable tree $T$. Let $B = \{2 < \sigma >: \sigma \in T\ \{\emptyset\}\}$ and $G = \{2 < \sigma > +1 : \sigma \in T\}$. $K$ consists of all pairs $(2 < \sigma >, 2 < \sigma > +1)$ for $\sigma \in T$ as well as the pairs $2 < \sigma >, 2 < \sigma \upharpoonright n > +1)$ for any $\sigma \in T$ with length $n+1$. An infinite path $x \in T$ corresponds to the matching which assigns girl $2 < x \upharpoonright n > +1$ to boy $2 < x \upharpoonright n+1 >$ and assigns girl $2 < \sigma > +1$ to boy $2 < \sigma >$ if $\sigma$ is not an initial segment of $x$. It is clear that if $T$ is highly computable, then $K$ will also be highly computable. □

**Theorem 10.1.4.** *The following problems can represent the c. b. $\Pi^0_1$ class of separating sets for any pair of disjoint infinite c. e. sets.*

*(i) The problem of finding a marriage in a highly recursive society.*

*(ii) The problem of finding a surjective marriage in a symmetrically highly recursive society where each person knows at most two other people.*

*Proof.* (i) For each $i \in \omega$, we will specify a boy $b_i$ and two girls $g_{0,i}$ and $g_{1,i}$ so that $b_i$ knows both $g_{0,i}$ and $g_{1,i}$ and no other. Our highly computable society $S = (B, G, K)$ will be such that $G = \{g_{0,i}, g_{1,i} : i \in \omega\}$ and $B = R \cup \{b_i : i \in \omega\}$, where $R = \{r_s : (A^s \cup B^s) - (A^{s-1} \cup B^{s-1}) \neq \emptyset\}$ is some infinite set of boys held in reserve. A marriage $f$ for S will code a set $C_f$ by specifying that $i \in C_f$ if and only if $f(b_i) = g_{1,i}$. We then determine who the boys in $R$ know in stages in such a way that

**(a)** if $i \in A$, then one boy in $R$ knows $g_{1,i}$ and no others and no boy in $R$ knows $g_{0,i}$;

**(b)** if $i \in B$, then one boy in $R$ knows $g_{0,i}$ and no others and no boy in $R$ knows $g_{1,i}$;

**(c)** if $i \notin A \cup B$, then no boy in $R$ knows $g_{0,i}$ or $g_{1,i}$.

Then if $i$ enters $A \cup B$ at stage $s$, we put $r_s \in B$ and we put $(r_s, g_{1,i})$ in K if $i \in A$ and $(r_s, g_{0,i})$ in K if $i \in B$. It is clear that this defines a highly computable society $S$ and that there is a one-to-one degree-preserving correspondence between the marriages $f$ for $S$ and the separating sets $C$ of $A$ and $B$, given by mapping $f$ to $C_f$.

(ii) Fix a pair $A$ and $B$ of infinite disjoint c. e. sets and recursive enumerations $\{A^s\}_{s\in\omega}$ and $\{B^s\}_{s\in\omega}$ such that, for all $s$, $A^s, B^s \subseteq \{0, 1, \ldots, s\}$ and there is at most one element of $A \cup B$ which comes into $A \cup B$ at stage $s$.

We first partition $\omega$ into a computable sequence $(G_0, B_0, G_1, B_1, \ldots)$ of infinite computable sets . For any fixed $i$, let $g^0_i < g^1_i < \ldots$ and $b^0_i < b^1_i < \ldots$ list the elements of $G_i$ and $B_i$ in increasing order. Our symmetrically highly computable society $S = (B, G, K)$ will be thought of as a bipartite graph with $B = \cup_i B_i$ and $G = \cup_i G_i$. The idea is to construct a connected component of $S$ with vertex set $G_i \cup B_i$ for each $i$. We construct the $i$-th component in stages, so that at stage $s$, we determine the edges out of $g^k_i$ and $b^k_i$ for $k \leq 2s$. We begin as if we are going to construct the two-way infinite chain in which $b^0_i$ is joined to $g_i^0$ and $g_i^1$ and such that, for each $n > 0$, $b_i^{2n}$ is joined to $g_i^{2n-2}$ and $g_i^{2n}$ and $b_i^{2n-1}$ is joined to $g_i^{2n-1}$ and $g_i^{2n+1}$. See Figure 10.1

Observe that there are exactly two possible surjective marriages $f$ for such a component depending on whether $f(b_i^0) = g_i^0$ or $f(b_i^0) = g_i^1$. A marriage $f : B \to G$ for $S$ will code a separating set $C_f$ for $A$ and $B$ by letting $i \in C_f$ if and only if $f(b_i^0) = g_i^1$. Then it is easy to see that all we need to do to ensure that each marriage $f$ of $S$ corresponds to a separating set $C_f$ for $A$ and $B$ is to construct the $i$-th component so that it is a one-way chain starting in $B_i$ if $i \in A$, a one-way chain starting in $G_i$ if $i \in B$, and the full two-way infinite chain if $i \notin A^s \cup B^s$. Thus we build the chain until we see that $i \in A \cup B$ at some stage $s$. That is, at each stage $t$, we add $b_i^k$ and $g_i^k$ for $k \in \{2t, 2t+1\}$ as pictured in Figure 10.1. Then if $i \in B^s$ omit $b_i^{2n}$ and $g_i^{2n}$ from the chain for all $n \geq s$ so that the chain will be a one-way infinite starting a girl $g_i^{2s-2}$. If $i \in A^s$, then add $b_i^{2s}$ and we omit $g_i^{2s}$ plus all boys and girls of the form $b_i^{2n}$ and $g_i^{2n}$ for $n > s$ from the chain so that the chain will be a one-way infinite chain starting at $b_i^{2s}$.

We note that we can consider this example as a computable version of problem (ii) by simply directing the edges of the graph down the left hand side of the graph and up the right hand side of the graph. That is, we can define the function $f : B \to G$ by saying that $f(b^*) = g^*$ is there is a directed edge from $b^*$ to $g^*$ in some component and define the function $g : G \to B$ by saying that $g(g^*) = b^*$ if there is a directed edge from $g^*$ to $b^*$ in some component. $\square$

Given these representation results, we have the usual corollaries.

**Theorem 10.1.5.** *(a) For each one of the three matching problems,*

*(1) There is a computable society $S$ which has a matching but has no computable matching.*

*(2) There is a computable society $S$ such that that any two distinct matchings are Turing incomparable.*

*(3) If $\mathbf{a}$ is a Turing degree and $\mathbf{0} <_T \mathbf{a} \leq_T \mathbf{0}'$, then there is a computable society s which has a matching of degree $\mathbf{a}$ but has no computable matching.*

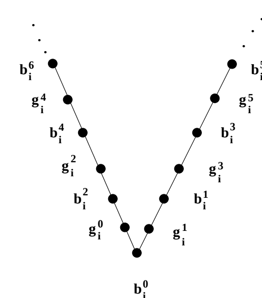
*(b) For the surjective matching problem, the following also hold.*

*(4) There is a computable society $S$ such that if $\mathbf{a}$ is the degree of any matching and $\mathbf{b}$ is a c. e. degree with $\mathbf{a} \leq_T \mathbf{b}$, then $\mathbf{b} = \mathbf{0}'$.*

*(5) If $\mathbf{c}$ is any c. e. degree, then there exists a computable society $S$ such that the set of c. e. degrees which contain matchings equals the set of c. e. degrees $\geq_T \mathbf{c}$.*

## 10.2 Graph-coloring problems

A countable infinite graph $G = (V, E)$ consists of a subset $V$ of the natural numbers called *vertices* together with a symmetric subset $E$ of $V \times V$, called



$b_i^6$
$g_i^4$
$b_i^4$
$g_i^2$
$b_i^2$
$g_i^0$
$b_i^0$
$g_i^1$
$b_i^1$
$g_i^3$
$b_i^3$
$g_i^5$
$b_i^5$

the *edges*. $G$ is said to be computable if the sets $V$ and $E$ are computable. We say that vertices $u, v$ are joined by an edge $(u, v)$. The *degree* of a vertex $u$ of $G$ is the cardinality of the set of vertices joined to $u$. A *$k$-coloring* of the graph $G$ is a map $g$ from $V$ into $\{1, 2, \ldots, k\}$ such that $g(u) \neq g(v)$ whenever $(u, v) \in E$. The *$k$-coloring problem* for a graph $G$ is to determine whether $G$ has any $k$-colorings. The set of solutions to this problem is the set of $k$-colorings of $G$. We make the convention that, unless stated otherwise, the graphs we shall discuss are assumed to be connected, have no loops or multiple edges, and have the property that each vertex $v$ of $G$ is of finite degree.

The graph coloring problem has been studied in combinatorics for over a century. Two classical results for finite graphs are Brooks' Theorem [13] that every graph with all vertices of degree $\leq k$ and with no $k+1$-cliques is $k$-colorable and the Four Color Theorem of Haaken and Appel [2] that every planar graph is 4-colorable. These results are easily extended to infinite graphs by a compactness argument. A natural question is whether such results can be effectivized. The answer to this question is yes for Brooks' Theorem, that is, Schmerl showed in [169] that every computable graph with all vertices of degree $\leq k$ and with no $k+1$-cliques has a computable $k$-coloring. On the other hand, the Four Color Theorem cannot be effectivized. Bean constructed in [7] a 3-colorable, computable, planar connected graph which has no computable $k$-coloring for any $k$.

A computable graph $G = (V, E)$ is said to be *highly computable* if there is a partial computable function $f : V \to \omega$ such that, for each $v \in V$, $f(v)$ is the degree of $v$. Highly computable graphs are of interest for several reasons. One reason is the result of Bean [7] that any highly computable $k$-colorable graph has a computable $2k$-coloring, in contrast to the result cited above for arbitrary computable graphs. This result was improved by Schmerl [168] from $2k$ to $2k-1$, who also showed that $2k-1$ is the best possible result. It follows from the work of Bean and Schmerl that every highly computable planar graph has a computable 6-coloring. This result was improved by Carstens [16] from 6 to 5, but the highly computable four color problem remains open.

Bean showed in [7] that the set of $k$-colorings of a highly computable graph is always a computably bounded $\Pi^0_1$ class. (See Exercise 1.)

Conversely, Remmel [161] showed that every c. b. $\Pi^0_1$ class can actually be strongly represented by a highly computable $k$-coloring problem.

The problem of feasible graphs and colorings has been studied by Cenzer and Remmel in [33].

**Exercises**

10.2.1. Show that for any highly computable graph $G = (V, E)$ and any finite $k$, the set of $k$-colorings of $G$ may be represented by a computably bounded $\Pi^0_1$ class $P \subseteq \{0, 1, \ldots, k-1\}^{\mathbb{N}}$.

10.2.2. Show that if $G$ is a planar graph, then the set of 5-colorings of $G$ always has cardinality $2^{\aleph_0}$ and hence not every c.b. $\Pi^0_1$ class may be represented as the set of 5-colorings of a planar graph. (Hint: every planar graph $G$ is 4-colorable, by the theorem of Appel and Haken [2].)

## 10.3 The Hamiltonian circuit problem

Let $G = (V, E)$ be a countably infinite graph. Two vertices $u, v$ of $G$ are *adjacent* if $(u, v) \in E$ and two edges $(u_1, v_1)$ and $(u_2, v_2)$ are *adjacent* if either $v_1 = u_2$ or $u_1 = v_2$. A *one-way* (respectively *two-way*) *Hamiltonian circuit* (or *Hamiltonian path*) for $G$ is a one-to-one correspondence $f$ between the natural numbers $\omega$ (resp. the integers $Z$) and $V$ such that consecutive vertices are adjacent, i.e. $(f(i), f(i+1)) \in E$ for all $i$. The dual concepts are the *one-way* (respectively *two-way*) *Euler path*, which is a one-to-one correspondence between the natural numbers $\omega$ (resp. the integers $Z$) and $E$ such that consecutive edges are adjacent. For each of these four notions, let us also define the associated notion of being such a path for a subgraph. That is, a one-way Hamiltonian sub-path for $G$ will be a one-to-one embedding of the natural numbers into $V$ such that consecutive vertices are adjacent. The other three definitions are similar.

In each case, the problem here is whether a given graph has such a path. We will focus on the sub-path problems.

**Theorem 10.3.1.** *For each of the following problems, the set of solutions can be represented as a $\Pi^0_1$ class. In cases (a) and (b), the class is bounded if the each vertex has finite degree and is c. b. if the graph is highly computable.*

*(a) The one-way Hamiltonian (Euler) sub-paths starting from a fixed vertex in a recursive graph.*

*(b) The two-way Hamiltonian (Euler) sub-paths through a fixed vertex in a recursive graph.*

*(c) The one-way Hamiltonian (Euler) paths starting from a fixed vertex in a recursive graph.*

*(d) The two-way Euler paths through a fixed vertex in a recursive graph.*

*Proof.* (a) Let the computable graph $G = (V, E)$ with fixed vertex $v_0$ be given. Then a one-way Hamiltonian (Euler) sub-path is a function $f$ from $\omega$ into $V$ with $f(0) = v_0$ such that $(f(n), f(n+1) \in E$ for all $n$ and such that, for the Hamiltonian path, $m \neq n$ implies that $f(m) \neq f(n)$ and, for the Euler path, $m \neq n$ implies that the edges $(f(m), f(m+1))$ and $(f(n), f(n+1))$ are different. In each case, this clearly defines a $\Pi^0_1$ class $P$. If each vertex $v$ has finite degree, then there is a function $g$ such that all vertices joined to vertex $v$ are $\leq g(v)$. It follows that we can compute a bound $h(m)$ for the possible value of $f(m)$ by letting $h(0) = v_0$ and in general $h(m+1) = sup\{g(v) : v \leq h(m)\}$. This shows that $P$ is bounded. If $G$ is highly computablee, then the function $g$ may be taken to be computable, so that $P$ is computably bounded.

(b) Again let the computable graph $G = (V, E)$ with fixed vertex $v_0$ be given. Then a two-way Hamiltonian (Euler) sub-path

$$\ldots, \pi(-1), \pi(0) = v_0, \pi(1), \ldots$$

can be represented as a function $f$ from $\omega$ into $V$ with $f(0) = v_0$ such that $(v_0, f(1)) \in E$, such that $(f(n), f(n+2)) \in E$ for all $n$ and such that, for the Hamiltonian path, the function $f$ is one-to-one, and, for the Euler path, no edge occurs twice in the list

$$\ldots, (f(3), f(1)), (f(1), f(0)), ((f(0), f(2)), (f(2), f(4)), \ldots.$$

It follows as in (a) that the class $P$ of two-way Hamiltonian (Euler) sub-paths is a $\Pi^0_1$ class, is bounded if each vertex of $G$ has finite degree, and is c. b. if $G$ is highly computable.

(c) We first give the proof for one-way Hamiltonian paths. Recall that $V = \omega$ and represent a one-way Hamiltonian path

$$\pi = (\pi(0) = v_0, \pi(1), \pi(2), \ldots)$$

by a function $f$ such that $f(2n) = \pi(n)$ and $f(2v+1) = n$ such that $v = \pi(n)$. This is clearly a one-to-one degree-preserving correspondence between the one-way Hamiltonian paths of $G$ and the $\Pi^0_1$ class $P$. Then the $\Pi^0_1$ class $P$ of solutions is the set of functions $f$ such that $f(0) = v_0$, such that $(f(2n), f(2n+2)) \in E$ for all $n$, and such that, for all $v$ and $n$, $f(2n) = v$ if and only if $f(2v+1) = n$. For the one-way Euler paths $\pi$, we take $f(2n) = \pi(n)$ and let $f(2[u,v]+1) = n+1$ such that $\pi(n) = u$ and $\pi(n+1) = v$ if $(u,v) \in E$ and otherwise $f(2[u,v]+1) = 0$. In either case, the assumption that $G$ is highly computable does not necessarily imply that $P$ is even bounded.

(d) Represent a two-way Hamiltonian path by a function $f$ so that the path is given by $\ldots, f(4), f(1), f(0) = v_0, f(3), f(6), \ldots$ and such that $f(3v+2) = n$ such that $n \neq 2 \; mod \; 3$ and $f(n) = v$. Represent a two-way Euler path $\pi$ again by a function $f$ so that $\pi = \ldots, f(4), f(1), f(0) = v_0, f(3), f(6), \ldots$ and now such that $f(3[u,v]+2) = n$ such that $n \neq 2 \; mod \; 3$ and $f(n) = u$ and $f(n+3) = v$. □

It follows that if each vertex of $G$ has finite degree and $G$ has a one-way or two-way Hamiltonian (Euler) sub-path, then it has such a sub-path which is computable in $\mathbf{0}''$. In the cases (c) and (d) of the Hamiltonian and Euler paths, we can only conclude, even for a highly computable graph $G$, that $G$ has a solution recursive in some $\Sigma^1_1$ set. We leave the other usual corollaries for the reader.

Bean [8] showed that if $G$ is highly computable and has an Euler path, then $G$ will actually have a computable Euler path. This is not the case for Hamiltonian paths, by the following reasoning. If every highly computable graph $G$ with a Hamiltonian path had a hyperarithmetic Hamiltonian path, then the set of highly computable graphs with Hamiltonian paths would be $\Pi^1_1$, by the Spector-Gandy theorem 1.14.5. However, Harel [77] showed that the problem of the existence of (one-way or two-way) Hamiltonian paths in a highly computable graph is $\Sigma^1_1$-complete and therefore not $\Pi^1_1$. It follows that the set of Hamiltonian paths of a highly computable graph is not always a c. b. $\Pi^0_1$ class. That is, if it were always a c. b. class, then by Theorem every highly computable graph with a Hamiltonian path would have a Hamiltonian path computable in $\mathbf{0}'$ and hence a hyperarithmetic Hamiltonian path. This applies to one-way and two-way paths.

For the reverse direction, we have the following result of Bean [8].

**Theorem 10.3.2.** *For any c. b. $\Pi^0_1$ class $P$, there is a highly computable planar graph $G$ and a one-to-one computable isomorphism between $P$ and the set of Hamiltonian paths for $G$.*

*Proof.* Let $P$ be the set of infinite paths through the highly computable tree $T$. $G$ is constructed in stages, beginning with vertices 0 and $0^{out}$ and edge $(0, 0^{out})$. At stage $n$, let $\sigma_0, \sigma_1, \ldots, \sigma_m$ be the nodes of $T$ at level $n$ and introduce a circuit of $3(m+1)$ vertice in $G$ given by

$$(\sigma_0^{out}, \sigma_0^{in}, \sigma_{0,1}, \sigma_1^{out}, \sigma_{1,2}, \ldots, \sigma_m^{out}, \sigma_m^{in}, \sigma_{m,0}.\sigma_0^{out}).$$

For every node $\tau_i$ at level $n-1$ and every successor $\sigma_j$ at level $n$ of $\tau_i$, also add an edge $(\tau_i^{out}, \sigma_j^{in})$ to $G$. (For the two-way circuit, also add a vertex $v_n$ and edge $(v_{n-1}, v_n)$, where $v_0 = 0$. It is clear that $G$ is a highly computable planar graph. The desired correspondence between $P$ and the Hamiltonian paths of $G$ is given as follows. The node $\sigma_j$ of $T$ follows the node $\tau_i$ on the infinite path through $T$ if and only if the vertex $\sigma_j^{in}$ immediately follows the vertex $\tau_j^{out}$ on the Hamiltonian path. $\square$

It follows that there is a highly computable graph $G$ which has Hamiltonian paths but has no computable Hamiltonian paths. Other corollaries are left to the reader.

This problem, posed by S. Ulam, is to show that for each partition of the vertex set $V$ of a graph $G = (V, E)$ into sets of uniformly bounded cardinality, there is at least one set of the partition which is adjacent to $m$ (or more) other sets of the partition. Here we say that two sets $S_1$ and $S_2$ are adjacent if there exist vertices $v_1 \in S_1$ and $v_2 \in S_2$ such that $(v_1, v_2) \in E$. The partition number $m$ of a graph $G$ is the least number $m$ for which the statement is true. The vertex partition problem was studied by Cenzer and E. Howorka [24], who computed the vertex partition numbers of various well-known graphs, including the $m$-regular trees $T_m$ and the planar mosaic graphs $M_3$, $M_4$ and $M_6$. The tree $T_m$ may be viewed as $\{1, 2, \ldots, m\}^*$. The graphs $M_3$, $M_4$ and $M_6$ may be viewed as tilings of the plane by regular hexagons, squares and equilateral triangles. In each case, the partition number of the graph turns out to be the degree of the graph. In this situation, the $\Pi^0_1$ class arises from the dual problem. That is, given the graph $G$ and numbers $k$ and $m$, to find a $k$-partition $P$ of the graph such that no set has $m$ neighbors. Here a $k$-partition is a partition of $V$ into sets of cardinality $\leq k$. The solution to such a problem may be represented as a function $f$ from $V \times V$ into $\{0, 1\}$ which is to be the characteristic function of the equivalence relation with equivalence classes being the sets of the partition.

**Theorem 10.3.3.** *For any highly computable graph $G = (V, E)$, and any finite $k$ and $m$, the set $C$ of $k$-partitions of $V$ such that no set in the partition is adjacent to $m$ other sets may be represented by a $\Pi^0_1$ class in $\{0,1\}^{\mathbb{N}}$.*

*Proof.* Let $G = (V, E)$ be a highly recursive graph and let $k, m$ be positive integers. Let $C$ be the set of $k$-partitions of $V$ such that no set in the partition is adjacent to $m$ other sets. As indicated above, we may represent a partition by the characteristic function $f$ of the corresponding equivalence relation. Let us assume that $V = \omega$ for simplicity and let $C$ be the class of all such functions for which there is no set in the partition represented by $f$ which has $m$ neighbors. Now a function $f \in \{0,1\}^\omega$ will be in the class $C$ if it satisfies the following conditions:

**(i)** $(\forall u)[f(u,u) = 1]$.

**(ii)** $(\forall u, v)[f(u,v) = f(v,u)]$.

**(iii)** $(\forall u, v, w)[f(u,v) = f(v,w) = 1 \rightarrow f(u,w) = 1]$.

**(iv)** $(\forall u_1, u_2, \ldots, u_{k+1})(\exists i, j \leq k+1)[f(u_i, u_j) = 0]$.

**(v)** $(\forall u_1, v_1, u_2, v_2, \ldots, u_m, v_m)[((\forall i, j \leq m)[f(u_i, u_j) = 1]$ &
$(\forall i \leq m)[E(u_i, v_i)]) \rightarrow (\exists i, j \leq m)[f(v_i, v_j) = 1]]$.

The first three clauses are the requirement that $f$ is the characteristic function of an equivalence relation. The fourth clause is the requirement that each set in the corresponding partition has cardinality $\leq k$ and the final clause is the requirement that no set in the partition is adjacent to $m$ other sets. $\square$

# Chapter 11

# Orderings

There are several problems associated with partially ordered sets (posets) and also with computable linear orderings and ordered structures.

Problems to be considered include the decomposition of a poset into chains and into antichains, as well as the problem of expressing a partial ordering as the intersection of finitely many linear orderings. For each computable instance of these problems, the set of solutions can be represented as a c. b. $\Pi_1^0$ class and can represent an arbitrary $\Pi_1^0$ class of separating sets.

For a computable linear ordering $\mathcal{A}$, we consider the problem of finding suborderings of type $\omega$ or $\omega^*$, the problem of finding an $\omega$-successivity or $\omega^*$-successivity, and the problem of finding a self-embedding of $\mathcal{A}$.

Finally, we consider the problem of finding an ordering of a computable Abelian group or formally real field. As usual, the set of orderings can always be represented by a c. b. $\Pi_1^0$ class and Metakides and Nerode [138] showed that any c. b. class. On the other hand, Solomon [182] showed that not every c. b. $\Pi_1^0$ class can be represented as the set of orderings of a computable abelian group.

## 11.1 Partial orderings

In this section we consider three problems associated with partially ordered sets (posets). Two of these are the dual problems of covering a poset with chains or with antichains. The third problem is the dimension problem, that is, expressing a poset as the intersection of linear orderings.

We first describe the problems and show that the solution set to a computable problem always forms a c. b. $\Pi_1^0$ class, and then apply the results of Part One to obtain corollaries which apply to the set of solutions of any such problem. We also consider for each problem, whether, conversely, the set of solutions to such a problem can represent any c. b. $\Pi_1^0$ class. For each problem, we show that the set of solutions to such a problem can represent the class of separating sets of any two disjoint c. e. sets and we apply the results of Part One to obtain corollaries which give the existence of "pathological" problems of each type.

**Decomposition problems for posets**

Here we start with a computable poset $\mathcal{A} = (A, \leq^A)$, which consists of a computable subset $A$ of $\mathbb{N}$ and a computable partial ordering $\leq^A$. The *width* of $\mathcal{A}$ is the maximum cardinality of an antichain in $\mathcal{A}$ and the *height* of $\mathcal{A}$ is the maximum cardinality of a chain in $\mathcal{A}$.

(a) The first decomposition theorem we consider is Dilworth's theorem [53], which states that any poset $\mathcal{A}$ of width $n$ can be covered by $n$ chains. The problem here is to find such a covering of $\mathcal{A}$ by $n$ chains and the set of solutions corresponds to the various coverings of $\mathcal{A}$ by $n$ chains. The effective version of Dilworth's theorem has been analyzed by Kierstead in [94], where he showed that every computable poset $\mathcal{A}$ of width $n$ can be covered by $(5^n - 1)/4$ computable chains, while for each $n \geq 2$, there are computable posets of width $n$ which cannot be covered by $4(n-1)$ chains. See Kierstead's article [93] in this volume for details and more results.

Thus, the set of solutions of this problem for a computable poset $\mathcal{A}$ can be represented as the set of maps $f : A \to \{1, 2, \ldots, n\}$ such that $f^{-1}(\{i\}) = \{x \in A : f(x) = i\}$ is a chain for each $i$, which is clearly a c. b. $\Pi^0_1$ class.

(b) There is a natural dual to Dilworth's theorem which says that every poset of height $n$ can be covered by $n$ antichains. The problem again is to find such a covering. The effective version of the latter theorem was analyzed by Schmerl, who showed that every computable poset of height $n$ can be covered by $(n^2 + n)/2$ computable antichains while for each $n \geq 2$, there is a computable poset of height $n$ which cannot be covered by $(n^2 + n)/2 - 1$ computable antichains. Furthermore, Szeméredi and Trotter showed that there exist computabl partial orders of height $n$ and computable dimension 2 which still cannot be covered by $(n^2 + n)/2 - 1$ computable antichains. These results are reported by Kierstead in [94].

**(2) Dimension of posets problem** The poset $\mathcal{A} = (A, R)$ is defined to be $n$-*dimensional* if there are $n$ linear orderings of $A$, $(A, L_1), \ldots, (A, L_n)$, such that $R = L_1 \cap \cdots \cap L_n$. The notion of the dimensionality of posets is due to Dushnik and Miller, who showed in [62] that a countable poset $(A, R)$ is $n$-dimensional if and only if it can be embedded as a subordering in the product ordering $\mathbb{Q}^n$, where $\mathbb{Q}$ is the set of rational numbers under the usual ordering. A (computable) poset $(A, R)$ has *(computable) dimension* equal to $d$, for $d$ finite, if there are $d$ (computable) linear orderings $(A, L_1), \ldots, (A, L_d)$ such that $R = L_1 \cap \cdots \cap L_d$, but there are not $d-1$ (computable) linear orderings $(A, L'_1), \ldots, (A, L'_{d-1})$ such that $R = L'_1 \cap \cdots \cap L'_{d-1}$. Kierstead, McNulty and Trotter have analyzed in [95], the computable dimension of computable posets and have shown that in general, the computable dimension of a poset is not equal to its computable dimension.

Given a countable poset $(A, R)$ with $A \subseteq \omega$, we can code a set of $d$ linear orderings of $A$, $(A, L_1), \ldots, (A, L_d)$ as follows. Let $a_0 < a_1 < \cdots$ be an increasing enumeration of $A$. Then given $d$ linear orderings of $\{a_0, \ldots, a_{n-1}\}$, there clearly are $(n+1)^d$ ways to extend the $d$ linear orderings to $d$ linear orderings on $\{a_0, \ldots, a_n\}$. One can fix some effective enumeration of these extensions for each n, so that it then becomes possible to code each $d$-tuple of linear orderings by a function $f : A \to \omega$ where $f(a_n) \leq (n+1)^d - 1$ for all $n$. Thus the set of solutions for the $n$-dimensionality problem of a computable poset $(A, R)$ can be represented as the set of all $f : A \to \omega$ such that $f$ codes an $n$-tuple, $(A, L_1), \ldots, (A, L_n)$, of linear orderings on $A$ such that $R = L_1 \cap \cdots L_n$, which is a c. b. $\Pi^0_1$ class.

We state the first theorem and leave the details of the representation to the reader.

**Theorem 11.1.1.** *For each specific computably presented instance of one of the poset problems $P$ listed above, the set of solutions can be represented as a c. b. $\Pi^0_1$ class.*

As usual, we can now derive a number of immediate corollaries from the results of Part One. We state only a few of these and leave the rest to the reader. For example, the following is true.

**Theorem 11.1.2.** *(a) If a computable poset $\mathcal{A}$ has a covering by $n$ chains, then $\mathcal{A}$ can be covered by $n$ chains $C_1, \ldots C_n$ such that $C_1 \oplus \cdots \oplus C_n$ has c. e. degree.*

*(b) If $\mathcal{A} = (A, R)$ is a computable poset such that the family of sets $\{(A, L_1), (A, L_2), \ldots, (A, L_n)\}$ of $n$ linear orderings such that $R = L_1 \cap L_2 \cap \cdots \cap L_n$ is countably infinite, then $\mathcal{A}$ has computable dimension $\leq n$.*

*(c) If a computable poset $\mathcal{A}$ has a covering by $n$ antichains, but has no covering by $n$ computable antichains, then for any countable sequence of nonzero degrees $\{\mathbf{a}_i\}$, $\mathcal{A}$ has a continuum of coverings $\{A_1, A_2, \ldots, A_n\}$ by $n$ antichains, which are pairwise Turing incomparable and such that the degree of $\{A_1, A_2, \ldots, A_n\}$ is incomparable with each $\mathbf{a_i}$.*

Next we consider the reverse direction of this correspondence.

**Theorem 11.1.3.** *Each of the three problems described above can strongly represent the c. b. $\Pi^0_1$ class of separating sets for any pair of disjoint infinite c. e. sets.*

*Proof.* Fix a pair $A$ and $B$ of infinite disjoint c. e. sets and computable enumerations $\{A^s\}_{s\in\omega}$ and $\{B^s\}_{s\in\omega}$ such that, for all $s$, $A^s, B^s \subseteq \{0, 1, \ldots, s\}$ and there is at most one element of $A \cup B$ which comes into $A \cup B$ at stage $s$.

(1) *The problem of covering a computable poset of width $k$ by $k$ chains.*
First consider the case $k = 2$. We begin with the poset $\mathcal{D}_0$ consisting of two one-way chains $\{a_{i,j} : i = 0, 1 \wedge j \in \omega\}$ and $\{b_{i,j} : i = 0, 1 \wedge j \in \omega\}$ where we have $a_{i,j} \leq a_{i,k}$ whenever $j < k$ and $a_{0,j} \leq a_{1,j}$ as well, and similarly for the

$b_{i,j}$. The two chains are linked by having $a_{0,j} \leq b_{1,j}$ and similarly $b_{0,j} \leq a_{1,j}$. Let us call the posets $\{a_{0,i}, a_{1,i}, b_{0,i}, b_{1,i}\}$ the *i-th block* of the poset $\mathcal{D}_0$. The $i$-th block of $\mathcal{D}_0$ is pictured in Figure 11.1(A).

Our final poset $\mathcal{D} = (D, \leq_D)$ will consist of the poset $\mathcal{D}_0$ together with an infinite computable set $E$ whose relations to the elements of $\mathcal{D}_0$ and among themselves is to be specified in stages. Now it is clear that a decomposition of this poset, up to renaming the chains, is completely determined by the choice, for each $i$, of either

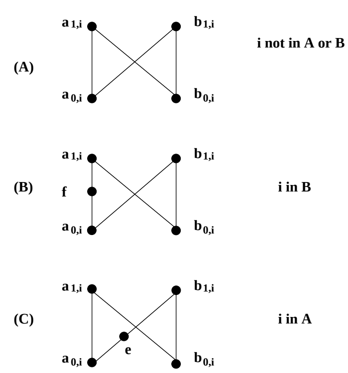
**(a)** putting $a_{0,i}$ and $a_{1,i}$ in one chain and $b_{0,i}$ and $b_{1,i}$ in the other, or

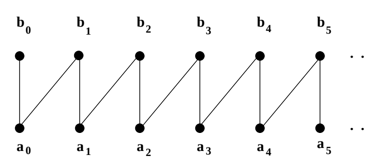
**(b)** putting $a_{0,i}$ and $b_{1,i}$ in one chain and $a_{1,i}$ and $b_{0,i}$ in the other.

Thus we can think of a chain decomposition $f : D \to \{1, 2\}$ as coding up a set $C_f$, where $i \in C_f$ if and only if we use choice (b) for the $i$-th component, that is, if and only if $f(a_{0,i}) = f(b_{1,i})$. Now the idea is to define the relations between the remaining computable set $E$ so that we introduce an element $e$ in the $i$-th component between $a_{0,i}$ and $a_{1,i}$ if $i \in B$, see Figure 11.1(B). This will force $e$, $a_{0,i}$ and $a_{1,i}$ to be in the same chain. We introduce an element $f$ in the $i$-th component between $b_{0,i}$ and $a_{1,i}$ if $i \in A$, see Figure 11.1(C). This will force $f$, $b_{0,i}$ and $a_{1,i}$ to be in the same chain. Finally we have no new element in the $i$-th component if $i \notin A \cup B$. It is not difficult to see that this can be accomplished so as to ensure that $\mathcal{D}$ is a computable poset of width 2 and that such actions will ensure that the correspondence $f \to C_f$ will be a one-to-one degree-preserving correspondence between the decompositions of $\mathcal{D}$ into two chains and the separating sets of $A \cup B$. We leave the details to the reader. For the case where $k > 2$, one simply adds to the poset described a set of $k - 2$ computable infinite one-way chains, all of whose elements are incomparable with $\mathcal{D}$ and so that elements from different chains are also incomparable.

(2) *The problem of covering a computable poset of width $k$ by $k$ antichains.* Again we shall initially consider the case $k = 2$. The poset $\mathcal{D} = (D, \leq_D)$ will consist of two parts. The first part of the poset will consist of a computable antichain $c_0, c_1, \ldots$, and the second part will consist of two antichains $a_0, a_1, \ldots$ and $b_0, b_1, \ldots$ where $a_0 \leq b_0$ and, for each i, $a_{i+1} \leq b_i$ and $a_{i+1} \leq b_{i+1}$, see Figure 11.1.

We will complete the partial ordering on $\mathcal{D}$ by specifying the relations between the two parts in stages. Clearly, up to renaming the antichains, there is a unique decomposition of the second part of the poset into two antichains. We think of a decomposition of $\mathcal{D}$ into two antichains as coding up a set $C_f$ by specifying $i \in C_f$ if and only if $f$ assigns $c_i$ to the same antichain as the $a$'s. Then, for each $i$, we define $c_i$ to be greater than $a_s$ if $i \in A^{s+1} \setminus A^s$ and incomparable to $a_s$ otherwise, and define $c_i$ to be less than $b_s$ if $i \in B^{s+1} \setminus B^s$ and incomparable to $b_s$ otherwise. It is then easy to check that $\mathcal{D}$ is a computable poset of height two and that, up to renaming the antichains, the correspondence $f \to C_f$ is a one-to-one degree preserving correspondence between decompositions of $\mathcal{P}$ into two antichains and separating sets of $A \cup B$. For the case where


(A)
a 1,i
b 1,i
a 0,i
b 0,i
i not in A or B
(B)
a 1,i
b 1,i
f
a 0,i
b 0,i
i in B
(C)
a 1,i
b 1,i
e
a 0,i
b 0,i
i in A


b0
b1
b2
b3
b4
b5
a0
a1
a2
a3
a4
a5

$k > 2$, one simply adds to the poset described a set of $k-2$ computable infinite antichains, all of whose elements are comparable with every element of $\mathcal{D}$ and so that elements from different antichains are also comparable.

(3) *The problem of expressing a computable poset $\mathcal{P} = (P, \leq P)$ of dimension $d$ as the intersection of $d$ linear orderings.*

We consider the case oftwo dimensional partial orderings. First we partition $\mathbb{N}$ into two infinite computable sets $C = \{c_0 < c_1 < \cdots\}$ and $D = \{d_0 < d_1 < \cdots\}$. For each i, we let $C_i = \{c_{5i}, c_{5i+1}, c_{5i+2}, c_{5i+3}, c_{5i+4}\}$. We shall define a computable partial ordering $<_P$ on $\omega$ in stages. Given any two sets E and F, $E <_P F$ will denote that, for any $e \in E$ and $f \in F$, $e <_P f$. We start by defining $<_P$ so that $C_0 <_P C_1 <_P C_2 <_P \cdots$. This means that if $<_1$ and $<_2$ are two linear orderings such that $<_1 \cap <_2 = <_P$, then the only difference between $<_1$ and $<_2$ on C is how $<_1$ and $<_2$ order the elements within the blocks $C_i$. For each block $C_i$, $<_P$ is defined so that we have the Hasse diagram in Figure 11.1(A).

It is then easy to check that, up to a permutation of the indices of the linear orderings $<_1$ and $<_2$, there are precisely two ways to define $<_1$ and $<_2$ on $C_i$ so that $<_1 \cap <_2$ equals $<_P$ restricted to $A_i$, namely,

**(I)** $c_{5i} <_1 c_{5i+1} <_1 c_{5i+2} <_1 c_{5i+3} <_1 c_{5i+4}$ and
$c_{5i+2} <_2 c_{5i+4} <_2 c_{5i+3} <_2 c_{5i} <_2 c_{5i+1}$, or

**(II)** $c_{5i} <_1 c_{5i+1} <_1 c_{5i+2} <_1 c_{5i+4} <_1 c_{5i+3}$ and
$c_{5i+2} <_2 c_{5i+3} <_2 c_{5i+4} <_2 c_{5i} <_2 c_{5i+1}$.

Note that the difference between (I) and (II) is that in the ordering where the elements $c_{5i}, c_{5i+1}$ precede the elements $c_{5i+2}, c_{5i+3}, c_{5i+4}$, we have $c_{5i+3}$ preceding $c_{5i+4}$ in (I), while in (II) $c_{5i+4}$ precedes $c_{5i+3}$.

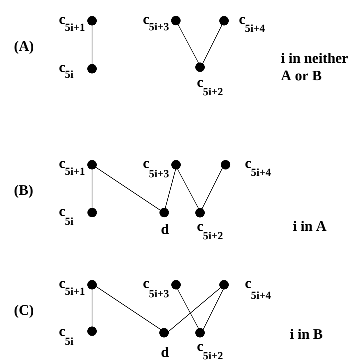
We can thus use a pair of linear orderings $<_1$ and $<_2$ such that $<_1 \cap <_2 = <_P$ is defined within the blocks $C_i$ to code a set $S(<_1, <_2) \subseteq \omega$ by declaring $i \in S$ if and only if $<_1$ and $<_2$ are of type (I) on $C_i$.

The key to our ability to code up a tree of separating sets for a pair of disjoint c. e. sets $A$ and $B$ is the following. If we add an element $d$ to the Hasse diagram as pictured in Figure 11.1(B), then only linear orderings $<_1$ and $<_2$ of type (I) can be extended to $C_i \cup \{d\}$ so that $<_1 \cap <_2 = <_P$ and if we add an element $d$ to the Hasse diagram as pictured in Figure 11.1(C), then only linear orderings $<_1$ and $<_2$ of type (II) can be extended to $C_i \cup \{d\}$ so that $<_1 \cap <_2 = <_P$.

That is, it is easy to check that, up to a permutation of indices there is only one way to define linear orderings $<_1$ and $<_2$ on $C_i \cup \{d\}$ so that $<_1 \cap <_2 = <_P$ if $<_P$ has the Hasse diagram as pictured in Figure 11.1(B), namely

$(I')$: $c_{5i} <_1 d <_1 c_{5i+1} <_1 c_{5i+2} <_1 c_{5i+3} <_1 c_{5i+4}$ and
$c_{5i+2} <_2 c_{5i+4} <_2 d <_2 c_{5i+3} <_2 c_{5i} <_2 c_{5i+1}$.

Similarly, up to a permutation of indices, there is only one way to define linear orderings $<_1$ and $<_2$ on $C_i \cup \{d\}$ so that $<_1 \cap <_2 = <_P$ if $<_P$ has the Hasse diagram as pictured in Figure 5 (C), namely


(A)
c5i+1
c5i
c5i+3
c5i+4
c5i+2
i in neither A or B
(B)
c5i+1
c5i
d
c5i+3
c5i+2
c5i+4
i in A
(C)
c5i+1
c5i
d
c5i+3
c5i+2
c5i+4
i in B

$(II')$: $c_{5i} <_1 d <_1 c_{5i+1} <_1 c_{5i+2} <_1 c_{5i+4} <_1 c_{5i+3}$ and
$c_{5i+2} <_2 c_{5i+3} <_2 d <_2 c_{5i+4} <_2 c_{5i} <_2 c_{5i+1}$.

Now to complete our definition of $<_P$ on $\omega$, we proceed in stages as follows.

*Stage 0* If $i \in A^0$, let $C_{i-1} <_P \{d_0\} <_P C_{i+1}$, and define $<_P$ on $C_i \cup \{d_0\}$ so that we have a Hasse diagram as in Figure 11.1(B). If $i \in B^0$, let $C_{i-1} <_P \{d_0\} <_P C_{i+1}$ and define $<_P$ on $C_i \cup \{d_0\}$ so that we have a Hasse diagram as in Figure 11.1(C). If $A^0 \cup B^0 = \emptyset$, define $\{d_0\} <_P C$. Note this defines $<_P$ on all of $C \cup \{d_0\}$ by transitivity.

*Stage $s > 0$*. Assume we have defined $<_P$ on $C \cup \{d_0, \ldots, d_{s-1}\}$ so that for all $j < s$, $C_{i-1} <_P \{d_j\} <_P C_{i+1}$ if $i \in (A^j \cup B^j) \setminus (A^{j-1} \cup B^{j-1})$ and $\{d_j\} <_P C \cup \{d_0, \ldots, d_{j-1}\}$ otherwise. Then if $i \in A^s \setminus A^{s-1}$, let $C_{i-1} <_P \{d_s\} <_P C_{i+1}$ and define $<_P$ on $C_i \cup \{d_s\}$ so that we have a Hasse diagram as pictured in Figure 11.1(B). If $i \in B^s \setminus B^{s-1}$, let $C_{i-1} <_P \{d_s\} <_P C_{i+1}$ and define $<_P$ on $C_i \cup \{b_s\}$ so that we have a Hasse diagram as pictured in Figure 11.1(C). If $(A^s \cup B^s) \setminus (A^{s-1} \cup B^{s-1}) = \emptyset$, define $\{d_s\} <_P C \cup \{d_0, \ldots, d_{s-1}\}$. Again this defines $<_P$ on all of $C \cup \{d_0, \ldots, d_s\}$ by transitivity.

This completes the proof of Theorem 11.1.3. $\square$

As usual, there are a number of immediate corollaries and we state only a few.

**Theorem 11.1.4.** *(a) There is a computable poset of width $k$ which has no covering by $k$ chains.*

*(b) There is a computable poset $\mathcal{A}$ of height $k$ such that any two distinct coverings of $\mathcal{A}$ by $k$ antichains are Turing incomparable, where distinct means not obtainable from the other by a permutation of the antichains in combination with the shifting of a finite number of elements.*

*(c) If $\mathbf{a}$ is a Turing degree and $\mathbf{0} <_T \mathbf{a} \leq_T \mathbf{0}'$, then there is a computable poset $\mathcal{A} = (A, R)$ of dimension $d$, but not of computable dimension $d$ such that there exists a set $\{(A, L_1), \ldots, (A, L_d)\}$ of degree $\mathbf{a}$ of linear orderings such that $R = L_1 \cap \cdots \cap L_d$.*

## 11.2 Linear orderings

There are three problems discussed in this subsection related to a given computable linear ordering $\mathcal{A} = (A, \leq^A)$.

**(1)** The problem of finding a subordering of $\mathcal{A}$ of type $\omega$ or of type $\omega^*$.

**(2)** The problem of finding an $\omega$-successivity or an $\omega^*$-successivity in $\mathcal{A}$.

**(3)** The problem of find a self-embedding of $\mathcal{A}$.

**(1) Suborderings of type $\omega$ or $\omega^*$**

A standard classical result is that any infinite linear ordering has a subordering $\{f(0), f(1), \ldots\}$ of order type either $\omega$ or $\omega^*$ (the order type of the negative integers). Tennenbaum and independently Denisov showed that there is an infinite computable linear ordering of order type $\omega + \omega^*$ which has no computably enumerable subordering of either type (see Rosenstein [165] or Downey [57]). The suborderings of type $\omega$ (respectively $\omega^*$) are simply the functions $f : \omega \to A$ such that $f(n) \leq^A f(n+1)$ (resp. $f(n+1) \leq^A f(n)$) for all $n$. Thus in each case the set of solutions to the problem of finding such a subordering is a $\Pi^0_1$ class, but is clearly not bounded. For example, if $\mathcal{A}$ is the standard ordering $(\omega, \leq)$, then the class of suborderings of $\mathcal{A}$ of type $\omega$ is just the class of all increasing sequences of natural numbers, which is homeomorphic to $\omega^\omega$ and not even compact. We observe that the class of suborderings of type $\omega$ is always a perfect set, since for any such subordering $f$ and any $n$, there is another subordering of type $\omega$ given by $(f(0), f(1), \ldots, f(n), f(n+2), f(n+4), \ldots)$.

**Theorem 11.2.1.** *For any computable linear ordering $\mathcal{A} = (A, \leq^A)$, the class of suborderings of $\mathcal{A}$ of type $\omega$ (respectively, of type $\omega^*$) is a perfect $\Pi^0_1$ class.*

Thus all we can say is that if a computable linear ordering has a subordering of type $\omega$ (respectively, type $\omega^*$), then it has such a subordering which is computable in some $\Sigma^1_1$ set. It was shown by Manaster that any computable linear ordering has a $\Pi^0_1$ subordering of type $\omega$ or of type $\omega^*$ (see Downey [57]).

**(2) Successivities**

An element $b$ of $\mathcal{A}$ is said to be the *successor* of an element $a$ if $a <^A b$ and there is no $c$ such that $a <^A c <^A b$; in such a case, we write $b = S_A(a)$. We say that a subordering $f$ of type $\omega$ in $\mathcal{A}$ is *an $\omega$-successivity* if $f(n+1)$ is the successor of $f(n)$ in the linear ordering for each $n$, and similarly define an $\omega^* - successivity$. Then the family $P$ of $\omega$-successivities is a $\Pi^0_1$ class and likewise the family of $\omega^*$-successivities.

Observe that the class of $\omega$-successivities of the standard ordering on $\omega$ consists of all sequences $(n, n+1, n+2, \ldots)$ and is thus a countable set in which all elements are isolated. As for the suborderings above, this class is not necessarily compact.

In general, there is at most one $\omega$-successivity $f$ for each starting element $f(0) = a$, so that every member of the class $P$ of $\omega$-successivities is isolated; a class with this property is said to be *scattered*. Clearly $P$ is also countable. Furthermore, we can define a bounded computable tree $T$ with $P = [T]$ by $(a_0, a_1, \ldots, a_n) \in T$ if and only if

$$(\forall i < n)[a_i <_A a_{i+1} \;\&\; (\forall m < a_{i+1})\neg(a_i <_A m <_A a_{i+1})].$$

A similar argument applies for $\omega^*$-successivities.

**Theorem 11.2.2.** *For any computable linear ordering $\mathcal{A} = (A, \leq^A)$, the class of $\omega$-successivities (respectively $\omega^*$-successivities) of $\mathcal{A}$ is a scattered, bounded $\Pi^0_1$ class.*

As an immediate application, we have the following.

**Corollary 11.2.3.** *Every $\omega$-successivity (respectively $\omega^*$-successivity) of a computable linear ordering $\mathcal{A}$ is computable in $\mathbf{0}'$.*

This of course may also be proven directly from the definition of a successivity. It follows from the result of Tennenbaum and Denisov that there is a computable linear ordering of type $\omega + \omega^*$ which has no computable $\omega$-successivity.

**(3) Self-embeddings**

Another classical result is due to Dushnik and Miller [62], who showed that an infinite countable linear ordering always has a non-trivial self-embedding. Hay, Manaster and Rosenstein [78] constructed a computable linear ordering of type $\omega$ with no non-trivial computable self-embedding. A map $f : A \to A$ is a self-embedding of $\mathcal{A}$ if, for all $a$ and $b$, $f(a) \leq^A f(b)$ if and only if $a \leq b$. The family of self-embeddings of a computable linear ordering is again seen to be a $\Pi^0_1$ class. For the standard ordering on $\omega$, it is clear that a self-embedding is the same thing as a subordering of type $\omega$. Thus the class of self-embeddings need not be compact.

Now $\mathcal{A}$ always has a computable self-embedding, namely the identity function. If $\mathcal{A}$ has a non-trivial self-embedding, then we can fix an element $a$ and consider the $\Pi^0_1$ class of self-embeddings $f$ such that $f(a) \neq a$. It follows as usual that $\mathcal{A}$ at least has a non-trivial self-embedding which is computable in some $\Sigma^1_1$ set.

**Theorem 11.2.4.** *For any computable linear ordering $\mathcal{A} = (A, \leq^A)$, the class of self-embeddings of $\mathcal{A}$ is a $\Pi^0_1$ class.*

**Theorem 11.2.5.** *For any computable linear ordering $\mathcal{A} = (A, \leq^A)$, if $\mathcal{A}$ has a non-trivial self-embedding, then $\mathcal{A}$ has a self-embedding computable in a $\Sigma^1_1$ set.*

Downey and Lempp [61] showed that the proof-theoretical stregnth of the Dushnik-Miller theorem is ACA, which implies that every computable linear ordering has a self-embedding which is computable in $\mathbf{0}'$.

## 11.3 Ordered algebraic structures

In this section, we consider two problems:

**(1)** The problem of finding an ordering of an Abelian group.

**(2)** The problem of finding an ordering of a formally real field.

In each case, the set of solutions to a given effective problem can always be represented by a *r.b.* $\Pi^0_1$ class and in case (2), any *r.b.* $\Pi^0_1$ class can be represented by such a set.

In this section, we will assume that a computably presented group, ring, or field is given by computable addition, subtraction, multiplication and division functions on the set $\omega$, as appropriate. A *c. e. ring* is the quotient of a computable ring modulo a c. e. ideal and a *c. e. group* is the quotient of a computable group modulo a c. e. normal subgroup. An ordering will be represented by the cone of positive elements.

A formally real field is a field $F$ such that no sum of (non-zero) squares equals zero. A field $(F, +^F, \cdot^F)$ is said to be *ordered* by the relation $\leq$ provided that $\leq$ is a linear ordering such that for all $a, b, c \in F$,

**(i)** $a \leq b \rightarrow a +^F c \leq b +^F c$.

**(ii)** $(0 \leq a \ \& \ 0 \leq b) \rightarrow 0 \leq a \cdot^F b$.

An ordering for a commutative group $(G, +^G, 0^G)$ is defined similarly except in this case the ordering $\leq$ need only satisfy condition (i).

The set $C = C_{\leq} = \{a \in F : 0 \leq a\}$ clearly satisfies the following for any $a, b \in F$:

**(i)** $a, b \in C \rightarrow a +^F b \in C$.

**(ii)** $a, b \in C \rightarrow a \cdot^F b \in C$.

**(iii)** $(a \in C \ \& \ 0^F -^F a \in C) \iff a = 0^F$.

**(iv)** $a \in C \ \vee \ 0^F -^F a \in C$.

A subset $C$ of $F$ satisfying (i) to (iv) is said to be a *positive cone* of $F$. Thus any linear ordering of $F$ defines a positive cone and conversely any positive cone $C$ of $F$ defines a linear ordering by

$$a \leq b \iff b -^F a \in C.$$

Thus we will identify the set of linear orderings of a field $F$ with the set of positive cones of $F$.

For a commutative group $(G, +^G, 0^G)$, a cone $C$ need only satisfy (i), (iii) and (iv).

The classical result of Artin-Schreier [4] is that any formally real field can be ordered. Craven showed in [50] that any closed subset $C$ of the Cantor space can be represented as the set of orderings of some formally real field $F$. Metakides and Nerode [137] made this proof effective by showing that if $C$ is a $\Pi^0_1$ class, then $F$ may be taken to be a computable field. Downey and Kurtz observed that the field $F$ may have additional orderings which are compatible with the group structure although not compatible with the field structure. The classical result for groups is due to Levi [116], who showed that an Abelian group can be ordered if and only if it is torsion-free. Downey and Kurtz constructed in [60] a computable group isomorphic to $\oplus_\omega Z$ which has no computable ordering.

**Theorem 11.3.1.** *For each specific c. e. instance of the problems (1) and (2) listed above, the set of solutions can be represented as a c. b. $\Pi^0_1$ class.*

*Proof.* For computable structures, this is immediate from the discussion above. For a c. e. structure, say, $F = R/I$, observe that a positive cone $C$ on $R/I$ corresponds to a subset $C'$ of $R$ satisfying clauses (i), (ii) and (iv) along with the following modified version of clause (iii).

**(iii)** $(a \in C' \ \& \ 0^F -^F a \in C') \iff a \in I.$

We leave it to the reader to translate these four clauses into a definition of a computable tree $T$ such that $[T]$ represents the set of positive cones on $F$. The proof for ordered groups is similar. □

We can as usual derive a number of immediate corollaries from the results of Part One. For example,

**Theorem 11.3.2. (a)** *Any c. e. presented group which has an ordering has an ordering of c. e. degree.*

**(b)** *If the set of orderings of the c. e. presented group $G$ is countably infinite and nonempty, then $G$ has a computable ordering.*

**(c)** *If the c. e. presented field $F$ has only finitely many orderings, then every ordering of $F$ is computable.*

Next we turn to the other direction of our correspondence, that is, representing an arbitrary $\Pi^0_1$ class by the set of solutions to certain of these problems. The problem of orderings of formally real fields was solved by Metakides and Nerode in [138].

**Theorem 11.3.3.** *Any c. b. $\Pi^0_1$ class $P$ can be represented by the set of orderings of a formally real field.*

*Proof.* Let the computable tree $T \subseteq \{0,1\}^{<\omega}$ be given so that $P = [T]$. The construction begins with the underlying ring $R = \mathbf{Q}[x_i : i \in \omega]$ (the ring of polynomials with rational coefficients in infinitely many variables). We define a computable maximal ideal of $R$ such that the set of orderings of the field $R/I$ represents $[T]$. We sketch a proof is which is somewhat different from that in [138].

The first step of our construction is to adjoin to $\mathbf{Q}$ the radicals $\sqrt{p_i}$, where $p_i$ is the $i$-th prime. That is, we put $x_i^2 - p_i$ into $I$ for each $i$. Thus we have initially a continuum of possible orderings on $Q[\sqrt{p_i} : i < \omega]$, where to each $\Pi \in \{0,1\}^\omega$ there corresponds the ordering $R(\Pi)$ determined by taking $x_i > 0$ if $\Pi(i) = 0$ and $x_i < 0$ if $\Pi(i) = 1$. Now for any $\sigma \notin T$, we use an auxiliary variable $y_\sigma$ to eliminate the ordering corresponding to $\sigma$ in the following manner. We uniformly and effectively define, for $\sigma$ of length $n$, a polynomial $f_\sigma(x_0, \ldots, x_{n-1})$ such that for $(e_0, \ldots, e_{n-1}) \in \{0,1\}^n$, $f_\sigma((-1)^{e_0}\sqrt{2}, (-1)^{e_1}\sqrt{3}, \cdots, (-1)^{e_{n-1}}\sqrt{p_{n-1}}) < 0$ if and only if $(e_0, \ldots, e_{n-1}) = \sigma$. Then we add to $I$ the polynomial $y_\sigma^2 = f_\sigma(x_0, \ldots, x_{n-1})$, thus adjoining to our field a square root for $f_\sigma(x_0, \ldots, x_{n-1})$. It follows that if $\sigma \prec \Pi$, then the ordering $R(\Pi)$ is not compatible with the field, since it forces a negative element to have a square root. The function $f_\sigma$ is defined to be $f_\sigma(x_0, \ldots, x_{n-1}) = c_\sigma - (-1)^{\sigma(0)}x_0 - \cdots - (-1)^{\sigma(n-1)}x_{n-1}$, where $c_\sigma$ is the least integer $c$ such that $\sqrt{2} + \sqrt{3} \cdots + \sqrt{p_{n-1}} > c$.

(For example, let $\sigma = (0, 1)$. Then we want $f_\sigma(\sqrt{2}, -\sqrt{3}) < 0$, $f_\sigma(\sqrt{2}, \sqrt{3}) > 0$, $f_\sigma(-\sqrt{2}, -\sqrt{3}) > 0$, and $f_\sigma(-\sqrt{2}, \sqrt{3}) > 0$. We compute that $3 < \sqrt{2} + \sqrt{3} < 4$ and define $f_\sigma(x_0, x_1) = 3 - x_0 + x_1$.)

Finally, to prevent any additional orderings from arising due to the new roots in the field, we add a sequence of roots $y_{i,j}$ to the field such that $y_{i,0} = y_i$ and $y_{i,j+1}^2 = y_{i,j}$. Thus each $y_i$ and each $y_{i,j}$ is forced to be positive. $\square$

This representation theorem has, as usual, a number of immediate corollaries of which we state only a few.

**Theorem 11.3.4.** *(a) There is a computable formally real field which has no computable ordering.*

*(b) There is a computable formally real field which has continuum many orderings and such that any two distinct orderings are Turing incomparable.*

*(c) There is a computable formally real field $F$ such that if $\mathbf{a}$ is the degree of any ordering of $F$ and $\mathbf{b}$ is a r.e. degree with $\mathbf{a} \leq_T \mathbf{b}$, then $\mathbf{b} =_T \mathbf{0}'$.*

*(d) There is a computable formally real field $F$ which has a unique non-computable ordering $\leq_0$, such that this ordering $\leq_0$ has degree $\mathbf{0}'$, and such that for any other ordering $\leq$ of $F$, there is some finite subset $A$ of $F$ such that for any ordering $\leq'$ of $R$, if $\leq$ agrees with $\leq$ on $A$, then $\leq = \leq'$.*

D. R. Solomon [182] showed that the analogue of the Metakides-Nerode theorem fails for Abelian groups, that is, every abelian group has either two or has infinitely many orderings and therefore not every $\Pi_1^0$ class may be represented as the set of orderings of a computable abelian group.

# Chapter 12

# Infinite Games

The set of winning strategies for an effective, closed $\{0,1\}$-game of perfect information was shown in [32] to strongly represent any c. b. $\Pi^0_1$ class. We will consider more general closed games here.

For any subset $C$ of $\mathbb{N}^{\mathbb{N}}$, the infinite game $G(C)$ of perfect information is defined as follows. Two players, I and II, alternately play an infinite sequence $z = (x(0), y(0), x(1), y(1), \ldots)$ and player II wins this play if $z \in C$. A *strategy* for Player II is a (partial) function $\Theta$ from $\omega^{<\omega}$ into $\omega$. For any play $x = (x(0), x(1), \ldots)$ of the game by Player I, the play $\Theta(x)$ of the game when $\Theta$ is applied to $x$ is given by $(x(0), y(0), x(1), \ldots)$, where, for each n, $y(n) = \Theta((x(0), y(0), \ldots, y(n{-}1), x(n))$. The strategy $\Theta$ is said to be a *winning strategy* for Player II in the game $G(C)$ if, for any play $x$ of the game by Player I, $\Theta(x) \in C$. The notion of a strategy and a winning strategy for Player I is similarly defined. The game $G(C)$ is said to be *determined* if one of the two players has a winning strategy. Gale and Stewart showed in [70] that the game $G(C)$ is determined if $C$ is either closed or open. For a closed set $C$, we have $C = [T]$ for some tree $T$, and we will sometimes refer to $G(C)$ as $G(T)$. We say that $G(T)$ is a *computably presented Gale-Stewart game* if $T$ is a recursive tree and that $G(T)$ is *bounded (respectively, computably bounded)* if the set $[T]$ is bounded (resp. c. b. ).

As pointed out in [32], strategies need to be coded to avoid always having a perfect set of winning strategies.

Let $\tau_0, \tau_1, \ldots$ effectively enumerate the nonempty elements of $\omega^{<\omega}$ in increasing order where we order the sequences by the sum of the sequence plus the length and then lexicographically. Thus $\tau_0 = (0), \tau_1 = (00), \tau_2 = (1), \ldots$. For each $\tau \in \omega^{<\omega}$, let $n(\tau)$ be the unique $n$ such that $\tau = \tau_n$. Then an arbitrary sequence $z = (z(0), z(1), \ldots) \in \omega^\omega$ codes a strategy $\Theta_z$ for Player II in the following manner. For any play $x = (x(0), x(1), \ldots)$ of Player I, the strategy $\Theta_z$ produces the following response $y = (y(0), y(1), \ldots)$ by Player II. First, $y(0) = z(n((x(0)))$ and for any $k$, $y(k+1) = z(n)$, where $\tau_n = (x(0), \ldots, x(k))$, that is, $\Theta_z((x(0), y(0), \ldots, y(k-1), x(k)) = z(n)$. Thus $z(0) = \Theta_z((0))$, $z(1) = \Theta_z(0, \Theta_z(0), 0)$, $z(2) = \Theta_z((1))$, and so on. It is clear that the result $\Theta_z(x)$ of applying this strategy to a play $x = (x(0), x(1), \ldots)$ of the game by Player I can be computed from $x$ and $z$ by a computable functional. For a finite sequence $z\lceil n = (z(0), \ldots, z(n-1))$, $\theta_{z\lceil n}$ is a *partial strategy* which, applied to any partial play $x\lceil m+1 = (x(0), \ldots, x(m))$ of Player I with $n(x\lceil m) < n$, gives a partial response $\theta_{z\lceil n}((x(0), y(0), \ldots, y(m-1), x(m))) = y(m)$ where for all $r \leq m$, $y(r) = z(k_r)$ if $n((x(0), \ldots, x(r)) = k_r$.

Now, for any tree $T \subseteq \omega^{<\omega}$, let $WS(T)$ be the set of codes

$$z = (z(0), z(1), \ldots) \in \omega^\omega$$

for winning strategies of Player II in the game $G(T)$.

**Theorem 12.0.5.** *For any computable tree $T$:*

*(a) $WS(T)$ is a $\Pi^0_1$ class.*

*(b) If $T$ is finitely branching, then $WS(T)$ is bounded.*

*(c) If $T$ is highly computable, then $WS(T)$ is computably bounded.*

*Proof.* We will define a computable tree $Q$ such that $WS(T) = [Q]$, as follows. First $\emptyset$ is in $Q$ and then for any $\sigma = (z(0), \ldots, z(n-1))$, $\sigma \in Q$ if and only if, for all sequences $\nu = (x(0), \ldots, x(r-1))$ where $n(\nu) < n$, the result of applying the partial strategy $\theta_\sigma$ coded by $\sigma$ to the partial play $\nu$ is in $T$. It follows from the discussion above that there is a computable function $g$ such that, for each $n$, the value $z(n)$ of a coded strategy gives the play $y(g(n))$ of player II at step $g(n)$. If $T$ is finitely branching, then there are only finitely many possible choices for $y(g(n))$ which allow player II to win the game, so that only finitely many values are possible for $z(n)$. This makes $WS(T)$ bounded. If $T$ is highly computabl, then we can actually compute a list of these possible values from $g(n)$. Thus $WS(T)$ will be computably bounded. $\square$

As usual, we can derive a number of immediate corollaries. We state the following and leave the rest to the reader.

**Theorem 12.0.6.** *Let $T$ be a computable tree such that player II has a winning strategy for the Gale-Stewart game $G(T)$.*

*(a) There is a winning strategy which is computable in some $\Sigma^1_1$ set and, if there are only finitely many winning strategies, then each winning strategy is hyperarithmetic.*

*(b) If $T$ is finitely branching, then there is a winning strategy which is computable in $0''$.*

*(c) If $T$ is highly computable, then there is a winning strategy of c. e. degree and, if there are only countably many winning strategies, then there is a computable winning strategy.*

*(d) If $T$ is highly computable and there is no computable winning strategy, then there is a continuum of pairwise Turing incomparable winning strategies.*

Next we consider the set of winning strategies for Player I (who is trying to get the play into the open set). Let $WS'(T)$ be the set of codes for winning strategies of Player I. Note that for a Gale-Stewart game $G(C)$, the set of winning strategies of Player I is in general not a closed set or an open set.

**Theorem 12.0.7.** *For any computable tree $T$:*

*(a) $WS'(T)$ is a $\Pi^1_1$ class.*

*(b) If $T$ is finitely branching, then $WS(T)$ is an open set.*

*(c) If $T$ is highly computable, then $WS(T)$ is a $\Sigma^0_1$ class.*

*Proof.* We describe the class of actual strategies $\Theta$ and leave it to the reader to translate this into the coded strategies as in the proof of Theorem 12.0.5. In general, $\Theta$ is a winning strategy for Player I if and only if, for all plays $y$ of Player II, the result $\Theta(y)$ of the game when Player I uses the strategy $\Theta$ is not in the set $[T]$, that is,

$$(\forall y)(\exists n)[(x(0), y(0), x(1), y(1), \ldots, x(n), y(n)) \notin T]$$

where $x(i+1) = \Theta((x(0), y(0), \ldots, x(i), y(i)))$ for all $i$.

If $T$ is finitely branching, let $f(n)$ give an upper bound for the possible values of $\sigma(n)$ for any $\sigma \in T$. Then we can use König's Infinity Lemma as usual to express this in the form

$$(*)\ (\exists n)(\forall (y(0), y(1), \ldots, y(n)))[(x(0), y(0), x(1), y(1), \ldots, x(n), y(n)) \notin T],$$

where each $y(i) \leq f(2i)$, so that the $(\forall)$ quantifier is bounded, which shows that $WS'(T)$ is an open set.

Finally, if $T$ is highly computable, then we may take the function $f$ to be computable, so that the characterization (*) above makes $WS'(T)$ a $\Sigma^0_1$ class. $\square$

**Theorem 12.0.8.** *Let $T$ be a computable tree such that Player I has a winning strategy for the Gale-Stewart game $G(T)$.*

**(a)** *There is a $\Delta^1_2$ winning strategy and, if there are only finitely many winning strategies, then each winning strategy is $\Delta^1_2$.*

**(b)** *If $T$ is finitely branching, then there is a computable winning strategy.*

*Proof.* (a) This follows from the theorem that $\Delta^1_2$ is a basis for $\Pi^1_1$, which is a corollary of the Novikov-Kondo-Addison Uniformization Theorem (see Hinman [80], pp. 196-198) for details).

(b) Since $WS'(T)$ is open and nonempty, there must be an interval of coded winning strategies, which of course will contain a computable strategy. $\square$

Now we consider the reverse direction of the correspondences given in Theorems 12.0.5 and 12.0.7.

**Theorem 12.0.9.** *For any computable tree $Q$, there is a computable tree $T$ and an effective one-to-one degree preserving correspondence between the $\Pi^0_1$ class $[Q]$ of infinite paths through $Q$ and the class $WS(T)$ of winning strategies for the effectively closed game $G(T)$. If $Q$ is finitely branching (respectively highly computable), then $T$ may be taken to be finitely branching (resp. highly computable).*

*Proof.* Let the computable tree $Q$ be given. Our basic idea is that each path $\Pi = (\pi(0), \pi(1), \ldots) \in [Q]$ should correspond to a strategy $\Theta_\Pi$ which acts as follows. Given any partial play, $((x(0), \ldots, x(m))$ of Player I, $\Theta_\Pi$ will respond with

$$\Theta_\Pi((x(0), y(0)), \ldots, y(m-1), x(m))) = y(m)$$

where $y(m) = 0$ if $x(i) > 0$ for any $i \leq m$ and $y(m) = \pi(m)$ if $x(i) = 0$ for all $i \leq m$. Thus whenever Player I plays a value $x(i) > 0$, then ever after $\Theta_\Pi$ will respond with a 0 and if Player I plays all 0's, then $\Theta_\Pi$ will respond by reproducing the path $\Pi$. It is easy to see that when we code the strategy $\Theta_\Pi$ via a sequence $z = (z(0), z(1), \ldots)$ that $z$ will have the same Turing degree as $\Pi$. Thus the correspondence $\Pi \rightarrow \Theta_\Pi$ will be an effective 1:1 degree preserving correspondence. Thus all we need to do is recursively define a computable tree $T \subseteq \omega^{<\omega}$ so that $WS(T) = \{z : z \text{ is a code of } \Theta_\Pi \text{ for some } \Pi \in Q\}$. We begin with sequences $(a, b)$ of length 2 by putting $(a, b) \in T$ if and only if, either $a > 0$ and $b = 0$ or $a = 0$ and $(b) \in Q$. (This ensures that if Player I starts with an $x > 0$, then any winning strategy $\Theta$ for Player II must respond with a 0, whereas if Player I starts with a 0, then Player II must respond by starting a sequence in $Q$. Similar remarks will apply to the subsequent nodes we put in $T$.) Then, for each $n$ and each $\tau = (x(0), y(0), \ldots, x(n), y(n)) \in T$, do the following:

**(1)** If $x(k) > 0$ for some $k \leq n$, then put $\tau^\frown a^\frown 0 \in T$ and leave $\tau^\frown a^\frown b$ out of $T$ for all $a$ and for all $b > 0$.

**(2)** If $x(k) = 0$ for all $k \leq n$, then put $\tau^\frown a^\frown b \in T$ if and only if, either $a > 0$ and $b = 0$ or $a = 0$ and $(y(0), \ldots, y(n), b) \in Q$.

It easily follows from the definition of $T$ that for any $\Pi = (\pi(0), \pi(1), \ldots) \in [Q]$, $\Theta_\Pi$ is a winning strategy for Player II for the game $G(T)$. Now suppose that $\Theta$ is a winning strategy for Player II for $G(T)$. Then we can define a $\Pi = (\pi(0), \pi(1), \ldots) \in [Q]$ such that $\Theta = \Theta_\Pi$ by recursion as follows. For each n, let $\pi(n) = \Theta((0, \pi(0), 0, \pi(1), \ldots, 0, \pi(n-1), 0))$. It is easy to see from our definition of $T$ that $\Pi \in Q$ and that $\Theta = \Theta_\Pi$. Thus the correspondence $\Pi \rightarrow \Theta_\Pi$ is our desired effective 1:1 degree preserving correspondence between $[Q]$ and $WS(T)$.

Suppose now that $Q$ is finitely branching (respectively, highly computable). Let $f(\pi)$ be an upper bound on $\{s : \pi^\frown s \in Q\}$; if $Q$ is highly computablee, then $f$ is computable. Now given a partial code $\sigma = (z(0), \ldots, z(n-1)) \in WS(T)$ for a strategy for the game $G(T)$, we will indicate how to compute an upper bound $g(\sigma)$ for $\{t : \sigma^\frown t \in WS(T)\}$. First compute the $n$-th finite sequence

$\tau_n = (\tau(0), \ldots, \tau(k-1))$ in the enumeration described above, and use $\sigma$ to compute the partial play $\pi = (\tau(0), y(0), \ldots, \tau(k-2), y(k-2), \tau(k-1))$–this can be done since for any $i < k$, $\tau\lceil i$ appears before $\tau$ in the enumeration. Now there are two cases in the computation of $g(\sigma)$. If $\tau(k-1) > 0$, then $g(\sigma) = 0$ and if $\tau(k-1) = 0$, then $g(\sigma) = f(\pi)$. Thus $WS(T)$ is finitely branching and if $Q$ is highly computable, then $g$ is recursive so that $WS(T)$ is highly computable. □

As usual, there are a number of immediate corollaries and we state only a few. Note that all of the examples below are games in which player II (who is trying to force the play into the closed set) has the winning strategy.

**Corollary 12.0.10.** *(a) There is a computably presented Gale-Stewart game such that Player II has a winning strategy but has no hyperarithmetic winning strategy.*

*(b) There is a computably presented, bounded Gale-Stewart game $G(C)$ such that Player II has a winning strategy and for any winning strategy $\Theta$ with $\mathbf{0}' <_T \Theta \leq_T \mathbf{0}''$, there is a $\Sigma^0_2$ set $A$ such that $\mathbf{0}' <_T A \leq_T \Theta$.*

*(c) For any c. e. degree $\mathbf{c}$, there is a computably presented, computably bounded Gale-Stewart game $G(C)$ such that Player II has a winning strategy and the set of c. e. degrees which contain winning strategies for $G(C)$ equals the set of c. e. degrees $\geq_T \mathbf{c}$.*

Next we consider the reverse direction for games in which Player I has a winning strategy. Here the bounded games all have computable winning strategies and nothing more can be said. For the unbounded games, the reverse direction demonstrates the connection between $\Pi^1_1$ classes and the game quantifier of Moschovakis. Recall that the $\Pi^0_1$ class with index $e$ is the set $[T_e]$ of infinite paths through the $e$-th primitive recursive tree $T_e$. A theorem of Moschovakis states that the set of indexes $e$ such that Player I has a winning strategy for the game $G(T_e)$ is a universal $\Pi^1_1$ set. See [142] for a discussion of the game quantifier and this theorem.

Note that every winning strategy for Player I is a limit point of the set of winning strategies for Player I, since once the play of the game has gotten into the open set, Player I may play anything at all from that point on. Thus we cannot hope to represent even every $\Pi^0_1$ class with a one-to-one correspondence.

**Theorem 12.0.11.** *For any $\Pi^1_1$ class $Q \subseteq \omega^\omega$, there is a recursively presented Gale-Stewart game $G(C)$ and a recursive function $F$ such that $y \in V \iff F(y) \in WS'(C)$.*

*Proof.* Suppose that $y \in Q \iff (\forall x)(\exists n) R(x\lceil n, y\lceil n)$. Define the closed set $C$ to be $\{(x, y) : (\forall n)\neg R(x\lceil n, y\lceil n)\}$. For each $y \in \omega^\omega$, let $F(y)$ code the strategy which simply plays $y$ in response to any play $x$ of Player I. Then it is clear that $F(y)$ codes a winning strategy if and only if $y \in Q$. □

**Theorem 12.0.12.** *(a) There is a computably presented Gale-Stewart game $G(C)$ such that the set $WS'(T)$ of winning strategies for Player I is not $\Sigma^1_1$.*

*(b) There is a computably presented Gale-Stewart game $G(C)$ for which Player I has a winning strategy but has no hyperarithmetic winning strategy.*

*Proof.* (a) This is immediate from Theorem 12.0.11.

(b) Let $Q = \{z\}$ be a $\Pi^1_1$ singleton such that $z$ is not hyperarithmetic and let the game $G(C)$ and the recursive function $F$ be given by Theorem 12.0.11. Then it is clear that Player I has a unique winning strategy which consists of playing $z(n)$ at his $n$-th turn, and that this strategy has the same degree as $z$. □

# Chapter 13

# The Rado Selection Principle

In this section, we summarize the results of Jockusch, Lewis and Remmel from [86]. A Rado Family consists of collection of finite subsets $\{A_i : i \in I\}$ of $A = \cup_{i \in I} A_i$ and a collection of finite partial functions $\{\phi_F :\in A^F \ : F \text{ is a finite subset of } I\}$ such that for each finite subset $F$ of $I$, $\phi_F(i) \in A_i$ for all $i \in F$. The Rado selection problem is to find a choice function $f : I \to A$ such that for any finite subset $F$ of $I$, there is a finite extension $E \supseteq F$ such that $f(i) = \phi_E(i)$ for all $i \in F$. We call such a choice function a *Rado selector*. Rado proved in [157] that any such family has a Rado selector. A finite set $F = \{x_1 < \ldots < x_n\}$ of natural numbers may be coded by $k = 2^{x_1} + 2^{x_2} + \cdots + 2^{x_n}$. In this case, we write $F = D_k$. We let 0 code the empty set. Then a family $\{A_i : i < \omega\}$ of finite sets may be coded by a function $f$ such that $A_i = D_{f(i)}$ for each $i$. Similarly a family of finite partial choice functions $\phi_F$ may be coded by a single function $g$ such that $g(i) = j$ if and only if $D_j = \{2^{x+1}3^{y+1} : x \in D_i \ \& \ \phi_{D_i}(x) = y\}$. A Rado family together with the coding described above is an *effective Rado family* $A = I = \omega$ and if the coding functions $f$ and $g$ are both computable.

Given an effective Rado family $\mathcal{F}$ as above, let $Ch(\mathcal{F})$ be the set of functions $h : \omega \to \omega$ such that

**(i)** $h(i) \in A_i$ for each $i$ and

**(ii)** for each finite $F \subseteq \omega$, there is a finite extension $E$ such that $\phi_E(i) = h(i)$ for all $i \in F$.

The following is Theorem 3 of [86].

**Theorem 13.0.13.** *For any effective Rado family $\mathcal{F}$, there is a bounded strong $\Pi^0_2$ class $P$ and an effective, degree-preserving correspondence between $P$ and $Ch(\mathcal{F})$.*

*Proof.* We can define a tree $T$ which is computable in $\mathbf{0}'$ such that $[T] = Ch(\mathcal{F})$ as follows. A finite path $(y_0, y_1, \ldots, y_n)$ is in $T$ if and only if

(i) $y_i \in A_i$ for all $i \leq n$ and

(ii) there exists a finite set $M$ such that $\{0, \ldots, n\} \subseteq M$ and $\phi_M(i) = y_i$ for all $i \leq n$. □

Applying Theorems 2.2.15, and 4.2.2, we obtain the following.

**Corollary 13.0.14.** *Let $\mathcal{F}$ be an effective Rado family. Then*

*(a) $\mathcal{F}$ has a Rado selector of $\Sigma_2^0$ degree.*

*(b) If $\mathcal{F}$ has only countably many Rado selectors, then $\mathcal{F}$ has a Rado selector which is computable in $\mathbf{0}'$.*

The following is Theorem 2 of [86].

**Theorem 13.0.15.** *For any nonempty bounded strong $\Pi_2^0$ class $P$, there exists an effective Rado family $\mathcal{F}$ and an effective, degree-preserving correspondence between $P$ and $Ch(\mathcal{F})$.*

We can now prove the following.

**Corollary 13.0.16.** *(i) There is an effective Rado family such that, for any degree $\mathbf{a}$ of a Rado selector for $\mathcal{F}$ and any $\Sigma_2^0$ degree $\mathbf{b} \geq_T \mathbf{a}$, $\mathbf{b} = \mathbf{0}''$.*

*(ii) There is an effective Rado family such that, for any two degrees $\mathbf{a}$, $\mathbf{b}$ of Rado selectors for $\mathcal{F}$, $\mathbf{a} \not\leq \mathbf{b} \vee \mathbf{0}'$.*

*(iii) There is an effective Rado family such that, for any degree $\mathbf{a} \leq_T \mathbf{0}''$ of a Rado selector for $\mathcal{F}$, there is a $\Sigma_2^0$ degree $\mathbf{b}$ with $\mathbf{0}' \leq_T \mathbf{b} \leq_T \mathbf{a}$.*

# Chapter 14

# Analysis

Computable functions on real numbers are just effectively continuous functions and $\Pi^0_1$ classes of reals are just effectively closed sets. Computable real functions may be represented by computable functions on natural numbers, by enumerating a countable basis of rational intervals. Effectively closed, compact sets of reals may be represented by $\Pi^0_1$ classes in $\{0,1\}^{\mathbb{N}}$. Weihrauch [198] has provided a comprehensive foundation for computability theory on various spaces, including the space of compact sets and the space of continuous real functions.

The basic example of a $\Pi^0_1$ class of reals is the sete of zeros of a computable function. Nerode and Huang [144] showed that any $\Pi^0_1$ class in $\{0,1\}^{\mathbb{N}}$ may be represented as the set of zeroes of a computable real function and Ko [101] showed that this can be done by polynomial time computable functions. This leads easily to the set of points where extreme values occur and to the set of fixed points of a computable function. Effective real dynamical systems have been studied by Cenzer [**?**], [102] and more recently by Cenzer, Dashti, King, Toska and Wyman [20, 21] and by S. Simpson.

Index sets for effectively closed sets of reals were studied by Cenzer and Remmel in [37, 38, 39].

Results from Chapter 5 are used to obtain the complexity of index sets related to the cardinality, computable cardinality, measure and category of effectively closed sets of reals. Index sets for computable real functions are defined and lead to complexity results for index sets corresponding to the zeroes, extrema, and fixed points of such functions.

Brattka and Weihrauch [10, 198] identify three different types of "effectively closed" sets in Euclidean space $\Re^n$. These are determined from an enumeration $I_m$ of the basic open sets (or intervals) and considering whether the set of $m$ such that $I_m$ (or $\overline{I_n}$ meets $K$ (or not) is a computably enumerable set. Of course, there are four possible notions here, and these can be refined further by asking whether the c.e. sets are in fact computable. These notions are developed in [38] and applied to the graphs of computable functions. In particular, we examine the question of whether a function with an effectively closed graph is "necessarily" continuous.

Polynomial time and NP versions of effectively closed sets are studied and a version of the "P=NP" problem is given. Here the choice of the basic open sets is crucial.

The fundamental problems for which the solution sets may be represented as $\Pi^0_1$ classes are the following.

**(1) Zeroes of continuous functions**

The classical problem here is to find a zero for a continuous function. The intermediate value theorem can be used to show the existence of a zero for a continuous function which is negative at one point and positive at another point. The effective version of this theorem also holds, that is, any computable function on the reals which is negative at one point and positive at another point has a computable zero, which can be computed by repeatedly splitting the interval between the two initial points. (See Pour-El–Richards [156] for a proof.) However, Lacombe [112, 113] showed that there are computable functions which have zeroes but have no computable zeros. We will give the improvement of this result due to Nerode and Huang [144] by showing that every $\Pi^0_1$ class is the set of zeroes of some computable function.

**(2) The Extreme Value Theorem**

The classical result here is that any function which is continuous on a compact set takes on a maximum and a minimum on that set. The problem here is to find a point where the maximum or minimum is attained. Lacombe showed that the extreme values of a computable function on $[0, 1]$ are themselves computable and also constructed a computable function $F$ on $[0, 1]$ which does not attain its maximum at any computable point. We will present the result of Nerode and Huang [144] that any $\Pi^0_1$ class may be represented as the set of points where some effectively continuous function attains its maximum.

**(3) Fixed points of continuous functions**

The problem here is to find a fixed point for a given continuous function. A simple application of the intermediate value theorem shows that any continuous function $F$ on [0,1] has a fixed point. It is well known that if $F$ is effectively continuous, then $F$ will have a computable fixed point. The Brouwer Fixed Point Theorem says that a continuous function on $[0, 1] \times [0, 1]$ will also have a fixed point, but Orevkov [152] showed that there need not be a computable fixed point. J. Miller [139] defined the notion of a *fixable* set as a $\Pi^0_1$ class $Q \subseteq [0, 1] \times [0, 1]$ for which there exists a computably continuous function $F$ such that $Q = \{z : F(z) = z\}$. He gave a beautiful result which characterizes the fixable sets. Results for other spaces are different. On the real line, the continuous function $F(x) = x + 1$ has no fixed point. On $\omega^\omega$, the function $F((x(0), x(1), \ldots) = (1 + x(0), 1 + x(1), \ldots)$ has no fixed point. On the Cantor space the function $F(x(0), x(1), \ldots) = (1-x(0), 1-x(1), \ldots)$ has no fixed point.

**(4) Dynamical systems**

We will give a few results on effective real dynamical systems from Cenzer [**?**] and from Ko [102]. We shall view a dynamical system as determined by a continuous function $F$ on a space $X$. The associated problem is to determine the behavior of the sequence $x, F(x), F(F(x)), \ldots$ for a given $x$. In particular, we want to find those points $x$ for which this sequence is bounded or converges to some finite number and those $x$ for which the sequence is unbounded or diverges to infinity where $X$ is either the real line or the Baire space. If $F$ is a polynomial, then it is always possible to compute a bound $c$ such that $\{F^{(n)}(x) : n < \omega\}$ is bounded if and only if $|F^{(n)}(x)| < c$ for all $n$. In fact, we can take $c$ large enough so that $F(x) > x + 1$ for all $x > c$, so that $lim_{n\to\infty} F^{(n)}(x) = \infty$ for all $x > c$. In this situation, we say that $\infty$ is an *attracting point* for $F$. Then $\{x : |F^{(n)}(x)| \leq c \text{ for all } n\}$ is called the *Julia set* of $F$. (See Blum, Shub and Smale [9].) It is then easy to see that the Julia set of any continuous function must be a compact set and we will show that for a computably continuous function, the Julia set is a $\Pi^0_1$ class. The first problem for dynamical systems is to find a member of the Julia set.

A point $x$ is said to be a *periodic point* of a continuous function $F$ if $F^{(n)}(x) = x$ for some finite $n$. The *basin of attraction B(x)* of $x$ is defined to be $\{u : lim_n F^{(n)}(u) = x\}$. The periodic point $x$ is said to be *attracting* if there is some open neighborhood $U$ about $x$ such that $U \subseteq B(x)$. The basin of attraction of infinity may also be defined as $\{u : lim_n F^{(n)}(u) = \infty\}$. Thus the basin of attraction is an open set. We will show that for a computably continuous function, the complement of a basin of attraction is a $\Pi^0_1$ class. If 1 is an attracting periodic point of a function $F$ on $\{0,1\}^\omega$ or $[0,1]$, then we will refer to the complement of $B(1)$ as the Julia set of $F$. The problem here is to find a point not in the basin of attraction.

Before turning to the problems mentioned above, we give a brief introduction to computable analysis, including the problem of characterizing the computable image of the interval and the related concept of a real as a Dedekind cut of rationals, which was studied by Soare in [179, 180].

A basic principle of computable analysis is that a computable function on the real numbers is an effectively continuous function and a $\Pi^0_1$ class is an effectively closed set. We will consider the real line $\Re$, as well as three subspaces: the space of irrationals, which is homeomorphic to the Baire space $\omega^\omega$ and two compact subspaces, the interval [0,1] and the Cantor space, which is computably homeomorphic to $\{0,1\}^\omega$. Since $\Re$ is computably homeomorphic to the open interval (0,1) via the order-preserving map $\frac{e^x}{1+e^x}$, we will frequently identify $\Re$ with $(0,1)$ and treat it as a subset of $[0,1]$.

Let $\mathcal{D}$ be the set of dyadic rationals in $[0,1]$. Then $[0,1]$ has a basis of open intervals $(a,b)$,$[0,c)$ or $(d,1]$ where $a,b,c,d \in \mathcal{D}$. Thus an open subset of [0,1] is a countable union

$$U = \cup_n(a_n, b_n) \bigcup \cup_n[0, c_n) \bigcup \cup_n(d_n, 1]$$

of dyadic intervals. The open set $U$ is said to be *effectively open*, or $\Sigma^0_1$, if the sequences $a_n$, $b_n$, $c_n$ and $d_n$ are computable. Then a closed set $C$ is said to be *effectively closed*, or $\Pi^0_1$, if it is the complement of an effectively open set.

Any $x \in \{0,1\}^{\mathbb{N}}$ represents a real $r_x = \sum_n x(n)/2^n \in [0,1]$. In addition, for any $\sigma \in \{0,1\}^*$, $\sigma^\frown 0^\omega$ represents the dyadic rational $q_\sigma = \sum_{i<n} \sigma(i)/2^i$. Some difficulty arises from the fact that $q_\sigma$ has another representation, $\sigma\lceil(n-1)^\frown 0^\frown 1^\omega$ (assuming that $\sigma$ ends in a 1). Each dyadic rational is of course computable, so that we may unambiguously say that $r$ is a *computable real* if $r = r_x$ for some computable sequence $x \in \{0,1\}^\omega$. Then a subset $P$ of $\{0,1\}^\omega$ represents a subset of $[0,1]$ if and only if, for all $x, y$ such that $r_x = r_y$, we have $x \in P$ if and only if $y \in P$. For any $\sigma \in \{0,1\}^{<\omega}$ of length $n$, the members of $I(\sigma)$ represent the members of the real closed interval $[q_\sigma, q_\sigma + 2^{-n}]$, which we denote by $U(\sigma)$. More generally, if $r < s$ are computable reals, then the interval $[r,s]$ is a $\Pi^0_1$ class, since, if $r = r_x$ and $s = r_y$, then

$$r_z \in [r,s] \iff (\forall n)[q_{x\lceil n} - 2^{-n} \leq q_{z\lceil n} \leq q_{y\lceil n} + 2^{-n}].$$

**Lemma 14.0.17.** *(a) The following are equivalent for any subset $K$ of $[0,1]$.*

- *(1) $K$ is a $\Pi^0_1$ class*
- *(2) $K$ is closed and $\{\langle p,r\rangle \in \mathcal{D}^2 : K \cap [p,r] = \emptyset\}$ is a c. e. set.*
- *(3) $K$ is represented by a $\Pi^0_1$ class $P \subset \{0,1\}^\omega$.*

*(b) $K$ may be represented by a computable binary tree with no dead ends if and only if $\{\langle p,r\rangle \in \mathcal{D}^2 : K \cap [p,r] = \emptyset\}$ is computable.*

*Proof.* (a) We show that both (1) and (3) are equivalent to (2). Suppose first that $K$ is a $\Pi^0_1$ class and let

$$[0,1] \setminus K = \cup_n (a_n, b_n) \bigcup \cup_n [0, c_n) \bigcup \cup_n (d_n, 1].$$

Then

$$K \cap [p,r] = \emptyset \iff (\exists n)[p,r] \subseteq \cup_{m<n}(a_m, b_m) \bigcup \cup_{m<n}[0, c_n) \bigcup \cup_{m<n}(d_n, 1].$$

Suppose next that $A = \{\langle p,r\rangle : K \cap [p,r] = \emptyset\}$ is an c. e. set. Then $K$ is a $\Pi^0_1$ class since

$$[0,1] \setminus K = \bigcup \{(p,r) : \langle p,r\rangle \in A\}.$$

Furthermore, $K$ is represented by $[T]$ where the $\Pi^0_1$ tree $T$ is defined as follows. Given $\sigma$ of length $n$, let

$$\sigma \in T \iff [q_\sigma, q_\sigma + 2^{-n}] \not\subseteq \cup\{(p,r) : \langle p,r\rangle \in A^n\}.$$

Here we replace $q + 2^{-n}$ with 1 if $q = 1$.)

Finally suppose that $K = \{r_x : x \in P\}$ for some $\Pi^0_1$ class $P = [T] \subseteq \{0,1\}^\omega$. Then for any $\sigma$,

$$K \cap [q_\sigma, q_\sigma + 2^{-n}] = \emptyset \iff \sigma \notin Ext(T)$$

Since any dyadic interval $[p, r]$ may be decomposed into a finite union of intervals of the form $[q_\sigma, q_\sigma + 2^{-n}]$, it follows that $\{\langle p, r\rangle : K \cap [p, r] = \emptyset\}$ is an c. e. set.

(b) This follows from the observation that, if $K$ is represented by $[T]$, then

$$\sigma \in Ext(T) \iff K \cap [q_\sigma, q_\sigma + 2^{-|\sigma|}] \neq \emptyset.$$

□

An arbitrary $\Pi^0_1$ class $P \subseteq \{0,1\}^\omega$ can be represented by a $\Pi^0_1$ subclass of [0,1] by the following lemma.

**Lemma 14.0.18.** *For any $\Pi^0_1$ class $P \subseteq \{0,1\}^\omega$, there is a $\Pi^0_1$ subclass $Q \subseteq \{0,1\}^\omega$ which represents a subset of $[0,1] \setminus \mathcal{D}$ which is computably homeomorphic to $P$.*

*Proof.* Let the computable homeomorphism $\Phi$ be defined by

$$\phi(x(0), x(1), \ldots) = (1, 0, x(0), 1, 0, x(1), \ldots)$$

and let $Q = \phi[P]$. $Q$ represents a subset of [0,1] since every element of $Q$ has both infinitely many "1"s and infinitely many "0"s. □

We can characterize those intervals which are $\Pi^0_1$ classes using the notion of the *Dedekind cut* $L(r) = \{q \in \mathcal{D} : q \leq r\}$ of a real number $r$. Soare showed in [179, 180] that if $x \in \{0,1\}^\omega$ is the characteristic function of a $\Pi^0_1$ set (respectively a $\Sigma^0_1$ set), then $L(r_x)$ is a $\Pi^0_1$ set (resp. a $\Sigma^0_1$ set) and that these implications are not reversible.

The set $\omega^{<\omega}$ and the space $\omega^\omega$ may be linearly ordered by the lexicographic ordering $\leq_L$, where $x <_L y$ if, for some $n$, $x(n) < y(n)$ and $x(i) = y(i)$ for all $i < n$. This ordering is computable on $\omega^{<\omega}$ and thus is $\Pi^0_1$ on $\omega^\omega$, since

$$x \leq_L y \iff (\forall n) x\lceil n \leq y\lceil n.$$

We now define the interval $[x, y] = \{z : x \leq_L z \leq_L y\}$ and also $[x, \infty] = \{z : x \leq_L z\}$. Then we let $L(x) = \{\sigma \in \omega^{<\omega} : \sigma^\frown 0^\omega \leq x\}$. These notions may also be restricted to $\{0,1\}^\omega$ and $\{0,1\}^{<\omega}$. Observe that for non-dyadic rationals $r_x$ and $r_y$, $r_x < r_y$ if and only if $x <_L y$.

**Lemma 14.0.19.** *(a) For any $x < y$ in either $[0,1]$, $\{0,1\}^\omega$, or $\omega^\omega$, the interval $[x, y]$ is a $\Pi^0_1$ class if and only if $L(x)$ is a $\Sigma^0_1$ set and $L(y)$ is a $\Pi^0_1$ set.*

*(b) In either $[0,1]$, $\{0,1\}^\omega$, or $\omega^\omega$: $L(x)$ is a computable set (respectively computable in $A$) if and only if $x$ is computable (resp. in $A$)*

*(c) For any $x \in \{0,1\}^\omega$, if $x$ is the characteristic function of a $\Sigma^{0,A}_1$ (respectively $\Pi^{0,A}_1$) set, then $L(x)$ is a $\Sigma^{0,A}_1$ (resp. $\Pi^{0,A}_1$) set.*

*Proof.* (a) First consider the case, where $x < y$ and $x, y \in \omega^\omega$. We claim that $[x, y]$ is a $\Pi^0_1$ class if and only if both $[x, \infty]$ and $[0, y]$ are $\Pi^0_1$ classes. The if direction follows from the fact that $[x, y] = [x, \infty] \cap [0, y]$. For the other direction, choose $\sigma \in \omega^\omega$ and $n$ such that $x\lceil n \leq_L \sigma <_L y\lceil n$ and observe that $[x, \infty] = [x, y] \cup [\sigma^\frown 1^\omega, \infty]$ and $[0, y] = [0, \sigma^\frown 0^\omega] \cup [x, y]$.

Thus we need only show that $[x, \infty]$ is a $\Pi^0_1$ class iff $L(x)$ is a $\Sigma^0_1$ set and that $[0, y]$ is a $\Pi^0_1$ class iff $L(x)$ is a $\Pi^0_1$ set. Suppose that $[x, \infty] = [T]$ for some computable tree $T$. Then

$$\sigma \in L(x) \iff \sigma^\frown 0^\omega \notin [T] \iff (\exists n)\sigma^\frown 0^n \notin T$$

and hence $L(x)$ is $\Sigma^0_1$ set. Vice versa, suppose that $L(x)$ is a $\Sigma^0_1$ set. Then we have

$$z \in [x, \infty] \iff (\forall m)(z\lceil m \notin L(x))$$

so that $[x, \infty]$ is a $\Pi^0_1$ class.

Similarly, if $[0, y] = [T]$ for some computable tree, then

$$\sigma \in L(y) \iff (\forall n)(\sigma^\frown 0^n \in T)$$

so that $L(y)$ is a $\Pi^0_1$ set. Vice versa, if $L(y)$ is $\Pi^0_1$ set, then

$$z \in [0, y] \iff (\forall n)(z\lceil n \in L(y))$$

so that $[0, y]$ is a $\Pi^0_1$ class.

For $x, y \in \{0, 1\}^\omega$, the argument is similar, except that $[x, \infty]$ is replaced by $[x, 1^\omega]$.

For $r_x, r_y \in [0, 1]$, the problem reduces to the previous case of $\{0, 1\}^\omega$, as long as we take $x$ to end in $0^\omega$ whenever $r_x \in \mathcal{D}$ and $y$ to end in $1^\omega$ whenever $r_y \in \mathcal{D}$, so that $q_\sigma \in [r_x, r_y] \iff \sigma \in [x, y]$.

(b) We give the argument for $\omega^\omega$. $L(x)$ is computable in $x$, since $\sigma \in L(x) \iff \sigma \leq_L x\lceil|\sigma|$. Also, $x$ is computable in $L(x)$, since for each $n$, $x(n+1)$ is the least $a$ such that $x\lceil n^\frown a \in L(x)$ & $x\lceil n^\frown a + 1 \notin L(x)$.

(c) Now suppose that $x$ is the characteristic function of a $\Pi^0_1$ set, i.e. $x$ is the characteristic function of $\omega \setminus A$ where $A$ is an r.e. set. Then let $A^s$ for $s \geq 0$ be some effective enumeration of $A$. Thus $x$ is the decreasing limit of a sequence $(x_0, x_1, \ldots)$ where $x_s$ is the characteristic function of $A^s$. Then $\sigma \in L(x) \iff (\forall n)(\sigma \leq_L x_n\lceil|\sigma|)$. Similarly, if $x$ is the characteristic function of a $\Sigma^0_1$ set $A$ then $x$ is the increasing limit of the sequence $(x_n)$. Hence $\sigma \in L(x) \iff (\exists n)(\sigma \leq_L x_n\lceil|\sigma|)$. □

It follows from part (b) that $L(x)$ is $\Delta^0_2$ if and only if $x$ is $\Delta^0_2$, and that if $x$ is $\Pi^0_2$ (respectively, $\Sigma^0_2$), then $L(x)$ is $\Pi^0_2$ (resp. $\Sigma^0_2$.)

**Theorem 14.0.20.** *(a) Let $x \in \omega^\omega$. If $x$ is the maximum element of a c. b. $\Pi^0_1$ class, $L(x)$ is a $\Pi^0_1$ set. If $L(x)$ is a $\Pi^0_1$ set and, in addition, $x$ is not hyperimmune, i.e. there is a computable function $f$ such that $x(e) \leq f(e)$ for all $e$, then $x$ is the maximum element of some r.b. $\Pi^0_1$ class. If $x$ is the* *minimum element of some r.b. $\Pi^0_1$ class, then $L(x)$ is a $\Sigma^0_1$ set. If $L(x)$ is $\Sigma^0_1$ set and, in addition, $x$ is not hyperimmune, then $x$ is the minimum element of some r.b. $\Pi^0_1$ class.*

(b) *For any $x$ in $[0,1]$, $x$ is the maximum element of some $\Pi^0_1$ class if and only if $L(x)$ is $\Pi^0_1$ and $x$ is the minimum element of some $\Pi^0_1$ class if and only if $L(x)$ is $\Sigma^0_1$.*

(c) *For any $x \in \omega^\omega$ or $[0,1]$, $x$ is the maximum (respectively, minimum) element of a $\Pi^0_1$ class represented by a tree with no dead ends if and only if $x$ is computable.*

(d) *For any $x \in \omega^\omega$, if $x$ is the maximum element of a bounded $\Pi^0_1$ class, then $L(x)$ is a $\Pi^0_2$ set and if $x$ is the minimum element of a bounded $\Pi^0_1$ class, then $L(x)$ is a $\Sigma^0_2$ set.*

(e) *For any $x \in \omega^\omega$, if $x$ is the maximum element of a $\Pi^0_1$ class, then $L(x)$ is a $\Sigma^1_1$ set and if $x$ is the minimum element of a $\Pi^0_1$ class, then $L(x)$ is a $\Pi^1_1$ set.*

*Proof.* We just give proofs for the maximum element versions.

(a) Suppose that $L(x)$ is a $\Pi^0_1$ set and there is a computable function $f$ such that $x(e) \leq f(e)$ for all $e$. Then $x$ is the maximum element of the $\Pi^0_1$ interval $[0,x]$ by Lemma 14.0.19. Hence $x$ is the maximal element of the *r.b.* $\Pi^0_1$ class $[0,x] \cap [T]$ where $T$ is the computable tree such that $\sigma \in T \iff (\forall i \leq |\sigma|)(\sigma(i) \leq f(i))$.

Now let $x$ be the maximum element of a *r.b.* $\Pi^0_1$ class $P = [T]$. Then $\sigma \in L(x)$ if and only if

$$(\exists y)[y \in P \;\&\; \sigma \leq_L y] \iff (\exists \tau \in \omega^{|\sigma|})[\tau \in Ext(T) \;\&\; \sigma \leq_L \tau].$$

Since $T$ is *r.b.*, the search for $\tau$ is bounded and, since $Ext(T)$ is a $\Pi^0_1$ set, $L(x)$ is a $\Pi^0_1$ set.

If $T$ has no dead ends, then $Ext(T)$ is computable, so that $L(x)$ is computable. This completes the proof of part (a) as well as part (c).

Part (b) now follows from Lemma 14.0.17.

(c) Any computable $x$ is the maximum element of the *r.b.* class $\{x\}$. The maximum element of a *r.b.* $\Pi^0_1$ class $P = [T]$ is computed by letting $x(n)$ be the largest $i$ such that $(x\lceil n-1)^\frown i \in T$.

Parts (d) and (e) follow from the characterization of $L(x)$ given above, since $Ext(T)$ is always $\Sigma^1_1$ and is $\Pi^0_2$ if $P$ is bounded. □

### 14.0.1 Computable continuous functions

Next we turn to the definition of computably continuous functions. For functions on $\omega^\omega$ or $\{0,1\}^\omega$, a computable function $y = F(x)$ is given by an oracle Turing machine which uses input $x$ as an oracle to compute the values $y(n)$ and is continuous since each value $y(n)$ depends on only finitely many values of $x$.

**Lemma 14.0.21.** *A function $F : \omega^\omega \to \omega^\omega$ (respectively, $F : \{0,1\}^\omega \to \{0,1\}^\omega$) is computably continuous if and only if there is computable function $f : \omega^{<\omega} \to \omega^{<\omega}$ (resp. $f : \{0,1\}^{<\omega} \to \{0,1\}^{<\omega}$) such that*

*(1) for all $\sigma \prec \tau$, $f(\sigma) \preceq f(\tau)$,*

*(2) for all $x \in \omega^\omega$, $\lim_{n\to\infty} |f(x\lceil n)| = \infty$, and*

*(3) for all $x \in \omega^\omega$, $\lim_{n\to\infty} f(x\lceil n) = F(x)$.*

*Proof.* Given such a representation $f$ for $F$, clearly we can compute $y(n)$ for $y = F(x)$ from $x$ by computing $f(x\lceil k)$ for sufficiently large $k$.

Given a computable function $F$, define the representation $f$ as follows. On input $\sigma$ of length $n$, compute the values of $\tau(i)$ where $\tau = f(\sigma)$ for each $i < n$ by applying the algorithm for $F$ for $n$ steps, using oracle $\sigma$. The length of $\tau$ will be the least $k < n$ such that $\tau(k)$ does not converge in $n$ steps. $\square$

In general, a function $F$ on the Baire space is continuous if and only if it has a representation $f$ as above. Thus $F$ is continuous if and only if it is computable in some parameter $x \in \omega^\omega$.

The definition of computably continuous real functions is more difficult.

**Definition 14.0.22.** *A function $F : [0,1] \to [0,1]$ is computable (or computably continuous if there is a uniformly computable sequence of functions $f_n : \mathcal{D} \to \mathcal{D}$ such that, for any $x \in \{0,1\}^\omega$, $F(r_x) = \lim_i f_i(q_{x\lceil i})$ and a computable function $\nu : \omega \to \omega$ such that, for all natural numbers $m, n, k$ and all dyadic rationals $q, r$, if $|q - r| < 2^{-\nu(k)}$ and $m, n > \nu(k)$, then $|f_m(q) - f_n(r)| < 2^{-k}$.*

This definition is easily seen to be equivalent to other standard definitions, such as those given by Lacombe [112]. See Pour-El–Richards [156] for some history.

Note for any computable real function, $F(x)$ is computable real for any computable real $x$.

Functions of several variables are treated similarly, thus a uniformly computable sequence of functions $\{f_n\}_{n\in\omega}$ and a computable function $\nu$ represent a continuous function $F : [0,1]^2 \to [0,1]$ if $\lim f_i(q_{x\lceil i}, q_{y\lceil i}) = F(x,y)$ for any reals $x, y$ and if $|f_m(q_1,q_2) - f_n(r_1,r_2)| < 2^{-k}$ whenever $m, n > \nu(k)$ and both $|q_1 - r_1|, |q_2 - r_2| < 2^{-\nu(k)}$. For example, the standard distance function $|x - y|$ may be represented by taking $f_n(q,r) = |q - r|$ for all $n$ and $\nu(k) = k + 1$.

We say a function $F : \{0,1\}^\omega \to \{0,1\}^\omega$ *represents* a real function $G$ provided $y = F(x)$ whenever $r_y = G(r_x)$.

**Lemma 14.0.23.** *If $F$ is a continuous (respectively computably continuous) map on $\{0,1\}^\omega$ such that $F(x) = F(y)$ whenever $r_x = r_y$, then $F$ represents a continuous (respectively computably continuous) map on $[0,1]$.*

*Proof.* Given the representation function $f$ for $F$, let $f_i(q_\sigma) = q_{f(\sigma)}$ for all $i$ and let $\nu(k)$ be the least $n$ such that $|f(\sigma)| > k$ for all $\sigma \in \{0,1\}^n$. $\square$

We remark that not every computably continuous real function may be represented by a computable function on $\{0,1\}^\omega$; the distance function $|x-r|$ for any fixed rational $r \in (0,1)$ is a counterexample. For example, suppose that $r = \frac{1}{6}$ and $G(x) = |x - \frac{1}{6}|$. Now suppose that $F : \{0,1\}^\omega \to \{0,1\}^\omega$ represents $G$ and that $f : \{0,1\}^{<\omega} \to \{0,1\}^{<\omega}$ represents $F$. Now $G(\frac{2}{3}) = \frac{1}{2}$ which has two representations $x_1 = 1^\frown 0^\omega$ and $x_0 = 0^\frown 1^\omega$. Let $x_2 = (10)^\omega$ so that $x_3$ represents $\frac{2}{3}$. Then either $F(x_2) = x_1$ or $F(x_2) = x_0$. Suppose first that $F(x_2) = x_0$. Then for some $n$, $0 \prec f((10)^n)$. But then $(10)^{n\frown}1^\omega$ is a number greater than $\frac{2}{3}$ so that $1 \prec f((10)^n 1^k)$ for some $k$ which is a contradiction. Similarly if $F(x_2) = x_1$, then for some $n$, $1 \prec f((01)^n)$. But then $(10)^{n\frown}0^\omega$ is a number less than $\frac{2}{3}$ so that $0 \prec f((10)^n 0^k)$ for some $k$ which is again a contradiction.

A computable metric on the Baire space is defined by $\delta(x,y) = 1/2^n = 0^n 1^\omega$, where $n$ is the least such that $x(n) \neq y(n)$, and $\delta(x,y) = 0 = 0^\omega$ if $x = y$.

The graph of a function $F : X \to X$ is defined as usual to be $gr(F) = \{(x, F(x)) : x \in X\}$. For $X = \omega^\omega$, we can view the graph as a subset of $X$ by associating the pair $(x,y)$ with the element $z = x \otimes y$, where $z(2n) = x(n)$ and $z(2n+1) = y(n)$. For any class $P$ and any $x \in X$, let $\pi_x(P) = \{y : x \otimes y \in P\}$. For a function $F$ from $[0,1]$ to $[0,1]$, the graph may be represented by a subset of $\{0,1\}^\omega$, namely $\{x \otimes y : f(r_x) = r_y\}$.

A classical result says that a function on the interval is continuous if and only if the graph is closed. We give the effective version here.

**Theorem 14.0.24.** *(a) The graph of a computably continuous function on $\omega^\omega$ is a $\Pi^0_1$ class.*

*(b) Let $X$ be either $\{0,1\}^\omega$ or $[0,1]$. Then a function $F : X \to X$ is computably continuous if and only if the graph of $F$ is a $\Pi^0_1$ class. Furthermore, the graph of any computably continuous function may be represented by a tree with no dead ends.*

*Proof.* (a) Suppose first that $F : \omega^\omega \to \omega^\omega$ is computably continuous and is represented by $f : \omega^{<\omega} \to \omega^{<\omega}$. Define the computable tree $T$ with $[T] = gr(F)$ by putting $\sigma \otimes \tau \in T$ if and only if $\tau$ is consistent with $f(\sigma)$.

(b) Given a computable $F : \{0,1\}^\omega \to \{0,1\}^\omega$, define the computable tree $T$ with $gr(F) = [T]$ as in (a). Then $Ext(T)$ is $\Sigma^0_1$, and therefore computable, by the following easily verified claim.

$$\text{CLAIM: } \sigma \otimes \tau \in Ext(T) \iff (\exists \sigma' \succ \sigma)\tau \prec f(\sigma').$$

Given a computable tree $T$ so that $gr(F) = [T]$, define the computable representing function $f$ by letting $f(\sigma)$ be the common part of $\{\tau : \sigma \otimes \tau \in T\}$.

Next suppose that $F$ is a computably continuous function on $[0,1]$ and let the computable sequence $f_i$ of dyadic rational functions and the computable modulus function $\nu$ be given as in Definition 14.0.22. We can assume that $\nu(k) > k$ for all $k$. In this case, we can define our desired computable tree $T$ with $gr(F) = [T]$ to be the set of pairs $\sigma \otimes \tau$ of length $2n-1$ or $2n$ such that $|f_n(q_\sigma) - q_\tau| \leq 2^{1-k}$ for all $k$ such that $|\sigma| \geq \nu(k)$. Again it is easy to see that $EXT(T)$ is computable.

Suppose now that $gr(F)$ is a $\Pi^0_1$ class and, by Lemma 14.0.17, let $T \subseteq \{0,1\}^{<\omega}$ be a computable tree so that $[T]$ represents $gr(F)$. $T$ may not be the graph of a function, since each dyadic real has two representations. However any two representations of length $n$ differ by $2^{-n}$. Thus, for any $i$ and any $\sigma$ of length $n$, we let $f_i(q_\sigma) = q_\tau$ for the lexicographically least $\tau$ of length $|\sigma|$ such that $\sigma \otimes \tau \in T$, and let $\nu(k)$ be the least $n$ such that, for all $\sigma$ of length $n$ and any $\tau_1, \tau_2$ with $\sigma \otimes \tau_1$ and $\sigma \otimes \tau_2$ both in $T$, $\delta_q(\tau_1, \tau_2) < 2^{-k}$. □

We next examine the complexity of the image of a $\Pi^0_1$ class under a computablly continuous function. The classical results are that the image of any compact set under a continuous function is compact and that the image of a closed set is an analytic set.

**Theorem 14.0.25.** *Let $F$ be a computably continuous function on a $\Pi^0_1$ subclass $P = [T]$ of $\omega^\omega$ or $[0,1]$ and let $F[P] = \{F(x) : x \in P\}$. Then*

*(a) $F[P]$ is a $\Sigma^1_1$ class,*

*(b) if $P$ is bounded, then $F[P]$ is a strong $\Pi^0_2$ class, and*

*(c) if $P$ is computably bounded, then $F[P]$ is a computably bounded $\Pi^0_1$ class and, furthermore, if there is a computable tree $T$ with no dead ends such that $P = [T]$, then there is a computable tree $S$ with no dead ends such that $F[P] = [S]$.*

*Proof.* (a) This part follows immediately from the fact that $y \in F[P] \iff (\exists x)(x \in P \;\&\; \langle x, y\rangle \in gr(F))$.

(b) Suppose that $T$ is a finitely branching, computable tree and let $S$ be a computable tree such that $gr(F) = [S]$. Then it follows from König's Lemma that $F[P] = [R]$, for the finitely branching $\Sigma^0_1$ tree $R$ defined by

$$\tau \in R \iff (\exists \sigma)[\sigma \in T \textit{ and } \sigma \otimes \tau \in S].$$

(c) Now suppose that $T$ is computably bounded and let $F$ be represented by the computable function $f : \omega^{<\omega} \to \omega^{<\omega}$. Then it is easy to see the definition above in (b) becomes computable.

To find a bound for the possible value of $\tau(n)$ for $\tau \in R$, compute the least $m$ such that $|f(\sigma)| > n$ for all $\sigma \in T$ of length $m$. Then we compute the maximum value $h(r)$ of $f(\sigma(n))$ for all $\sigma \in T$ of length $n$. Thus $R$ is highly computable.

ly continuous map $F(x) = r + (s - r)x$.

Suppose that $K$ is the image of the computably continuous map $F$. It follows from the Intermediate Value Theorem that $K = [r, s]$ where the reals $r$ and $s$ are the maximum and minimum elements of $P$. It follows from Theorem 14.0.25 that $K$ may be represented by a tree with no dead ends and then from Theorem 14.0.20 that $r$ and $s$ are computable. □

**Corollary 14.0.26.** *Let $F$ be a computably continuous function on $\omega^\omega$, $\{0,1\}^\omega$, or $[0,1]$. Then the maximum and minimum values of $F$ on $P$ are computable reals (if they exist).*

**Theorem 14.0.27.** *Each of the following sets is a $\Pi^0_1$ class for any computably continuous function $F : X \to X$, where the space $X$ may be $\{0,1\}^\omega$, $[0,1]$, $\omega^\omega$ or $\Re$. In case (3), the class always has a compuable member when $X = [0,1]$. In case (4), the class is always bounded when $X = \Re$.*

*(a) The set of points $x$ where $F(x) = x_0$ for any fixed computable $x_0$.*

*(b) The set of points where $F$ attains its maximum (minimum).*

*(c) The set of fixed points of $F$.*

*(d) The Julia set of $F$ where $X = \omega^\omega$ or $\Re$.*

*(e) The complement of the basin of attraction of a computable periodic point.*

*Proof.* (a) This is immediate from Theorem 14.0.24.

(b) It follows from Corollary 14.0.26 that the maximum and minimum are computable if they exist. The result now follows from part (b).

(c) This is easily reduced to part (b). For $\Re$, $x$ is a fixed point of $F$ if and only if $F$ is a zero of $G(x) = F(x) - x$. For $[0,1]$, take $G(x) = |F(x) - x|$. For $\{0,1\}^\omega$ or $\omega^\omega$, define $z = G(x)$ by $z(n) = |F(x)(n) - x(n)|$.

A computable fixed point $r$ may be found for a computably continuous function on $X = [0,1]$ by the standard procedure. If $F$ has a dyadic fixed point, then there is nothing to do. If not, then repeatedly split the interval in two and choose the subinterval with $F(x) < x$ on one end and $F(x) > x$ on the other. Then $r$ is the unique element in the intersection of these intervals.

(d) This is immediate from the characterization of the Julia set as $\{x : (\forall n)|F^n(x)| \leq c\}$ for a fixed computable point $c$. Note that in $\omega^\omega$, $\{x : x \leq_L x_0\}$ is not a bounded $\Pi^0_1$ class in our sense of being the paths through a finite branching tree.

(e) Given an attracting point $c$ for $F$, there is some computable interval $(a,b) \subseteq B(x)$ containing $c$. Then the complement of the basin of attraction may be characterized as

$$\{x : (\forall n)(F^n(x) \leq a \vee F^n(x) \geq b)\}$$

□

As usual, we give a few immediate corollaries from the results of Part One.

**Theorem 14.0.28.** *Let $F : X \to X$ be a computably continuous map, where $X$ is either $\{0,1\}^\omega$, $[0,1]$ or $\Re$.*

*(a) If $F$ attains a maximum $M$, then there are two points $x_1$ and $x_2$ with $F(x_1) = F(x_2) = M$ such that any function computable in both $x_1$ and $x_2$ is computable.*

*(b) If $F$ has only countably many zeroes, then $F$ has a computable zero.*

(c) *If $F$ has only finitely many fixed points, then every fixed point of $F$ is computable.*

(d) *If the Julia set of $F : \Re \to \Re$ has no computable member, then it contains a continuum of pairwise Turing incomparable elements.*

(e) *If the basin of attraction $B(x_0)$ of a computable fixed point $x_0$ of $F$ is not all of $X$, then there is a point $x$ of c. e. degree which is not in $B(x_0)$.*

*Proof.* We just note that in each case, a function defined on $\Re$ may be restricted to a finite interval and thus be treated as a map on the interval. For example, if $F$ has a zero, take a computable interval $[a, b]$ on $F$ has a zero and let $[c, d]$ be the image of $[a, b]$ under $F$. Then $F$ may be composed with maps between $[0, 1]$ and the two intervals to obtain a map $G : [0, 1] \to [0, 1]$ so that the set of zeroes of $G$ is homeomorphic to a subset of the set of zeroes of $F$. □

**Theorem 14.0.29.** *Let $F : \omega^\omega \to \omega^\omega$ be a computably continuous map.*

(a) *If $F$ attains a maximum $M$, then $F(x) = M$ for a point $x$ which is computable in some $\Sigma^1_1$ set.*

(b) *If $F$ has only countably many zeroes, then $F$ has a hyperarithmetic zero.*

(c) *If $F$ has only finitely many fixed points, then every zero of $F$ is hyperarithmetic.*

Next we give the collection of converses to Theorem 14.0.28. The first three parts are due to Nerode and Huang [144] and may also be found in Ko [101].

Next we give the collection of converses to Theorem 14.0.28. The first three parts are due to Nerode and Huang [144] and may also be found in Ko [101].

**Theorem 14.0.30.** *Let $P$ be a $\Pi^0_1$ subclass of the space $X$, either $\{0, 1\}^\omega$, $[0, 1]$, $\omega^\omega$ or $\Re$.*

(1) *There is a computably continuous function $F$ such that $P$ is the set of zeroes of $F$.*

(2) *There is a computably continuous function $F$ with maximum value $M$ such that $P = \{x : F(x) = M\}$.*

(3) (a) *If $X$ is either $\{0, 1\}^\omega$, $\omega^\omega$ or $\Re$, then there is a computably continuous function $F$ such that $P$ is the set of fixed points of $F$.*

   (b) *If $X$ is $[0, 1]$ and $P$ has a computable member, then there is a computably continuous function $F$ such that $P$ is the set of fixed points of $F$.*

(4) *If $P$ is bounded and has both a computable maximum and a computable minimum element, then there is a computably continuous function such that*

*(a) P is the complement of the basin of attraction of a computable periodic point, where $X = [0,1]$ or $\Re$.*

*(b) P is the Julia set of F where $X = \Re$*

*Proof.* (1) First suppose $P \subseteq \omega^\omega$ and let $T$ be a computable tree such that $P = [T]$. Define the computable function $F$ by

$$F(x) = \begin{cases} 0^\omega, & \text{for } x \in P \\ 0^n\frown 1\frown 0^\omega, & \text{if } n \text{ is the least with } x\lceil n \notin T. \end{cases}$$

If $P$ represents a subset of $[0,1]$, then the function $F$ is modified when $x$ represents a dyadic, so that $F(\sigma\frown 1\frown 0^\omega) = F(\sigma\frown 0\frown 1^\omega)$ for all $\sigma$. Thus when $r_x = r_y$ is dyadic, we let

$$F(x) = F(y) = \begin{cases} 0^\omega, & \text{for } x \in P \\ 0^n\frown 1\frown 0^\omega, & \text{if } n \text{ is the least with } x\lceil n \notin T \text{ and } y\lceil n \notin T. \end{cases}$$

For a subset $P$ of $\Re$, let $Q$ be the image of $P$ under the isomorphism $G$ with $(0,1)$ together with the point 0, if $P$ has no lower bound and the point 1, if $P$ has no upper bound. Then let $H$ be the computably continuous map with set $Q$ of zeroes. It follows that $P$ is the set of zeroes of $H \circ G$.

(2) Let $F$ be the function defined in the proof of (1) and observe that 0 is the minimum value of $F$ in each case. For the maximum argument on $[0,1]$ or $\Re$, just take $G(x) = 1 - F(x)$. For the maximum argument on $\omega^\omega$, note that the range of $F$ is a subset of $\{0,1\}^\omega$ and take $G(x)(n) = 1 - F(x)(n)$.

(3) (a) Let $F$ be given by (1) so that $P$ is the set of zeroes of $F$. Now let $G(x) = F(x) + x$ for the real line, and, for $\omega^\omega$ or $\{0,1\}^\omega$, let $G(x)(n) = x(n)$ if $F(x)(n) = 0$ and $G(x)(n) = 1 - x(n)$, if $F(x)(n) \neq 0$.

(b) Let $x_0$ be a computable member of the $\Pi^0_1$ class $P$ and let $F$ be the function given by (1) so that $x \in P$ if and only if $F(x) = 0$. Define $G(x)$ to be $x + (x_0 - x)F(x)$.

(4) (a) Let $P$ be a $\Pi^0_1$ proper subclass of $[0,1]$. Then there is some computable element $x_0 \notin P$. Let $F$ be the computable function given by part (1) such that $F(x) = 0$ for $x \in P$ and $F(x) > 0$ for $x \notin P$. Let $P_1 = P \cap [0, x_0]$ and $P_2 = P \cap [x_0, 1]$. Let $M_1$ be the maximal element of $P_1$ and let $M_2$ be the minimal element of $P_2$, so that both $M_1$ and $M_2$ are computable. Now define the function $G$ by cases.

$G(x) = M_1 + F(x)(x_0 - M_1)$, for $x \leq M_1$;
$G(x) = x + (x - M_1)(x_0 - x)$, for $M_1 \leq x \leq x_0$;
$G(x) = x - (M_2 - x)(x - x_0)$, for $x_0 \leq x \leq M_2$;
$G(x) = M_2 - F(x)(M_2 - x_0)$, for $M_2 \leq x$.

Then $x_0$, $M_1$ and $M_2$ are all fixed points of $G$. We claim that $P$ is the complement of the basin of attraction of $x_0$. The following inequalities are immediate from the above definition.

$M_1 \leq G(x) \leq x_0$, for $x < M_1$;
$x < G(x) < x_0$, for $M_1 < x < x_0$;
$x_0 < G(x) < x$, for $x_0 < x < M_2$;

$x_0 \leq G(x) < M_2$, for $M_2 < x$.

First we show that the basin of attraction of $x_0$ for $G$ includes $[M_1, M_2]$. Given $M_1 < x < x_0$, we see that $x < G(x) < x_0$. It follows that $G^n(x)$ is an increasing sequence with limit $L$ such that $G(L) = L$ and $M_1 < L \leq x_0$. Thus we must have $L = x_0$. A similar argument works for $x_0 < x < M_1$.

Next suppose that $x \notin P$ and either $x < M_1$ or $x > M_2$. Then either $G(x) \in [M_1, M_2]$, so that $x$ is in the basin of attraction of $G$.

Now suppose that $x \in P$, so that either $x \in P_0$ or $x \in P_1$. For $x \in P_0$, we have $F(x) = 0$ and $x \leq M_1$, so that $G(x) = M_1$ and thus $G^n(x) = M_1$ for all $n > 0$. Thus $x$ is not in the basin of attraction of $G$. Similarly for $x \in P_1$, $G^n(x) = M_2$ for all $n > 0$, so that $x$ is not in the basin of attraction of $G$.

For $X = \Re$, just identify $X$ with a subclass of $(0, 1)$ as in (1) above.

(b) Let $P$ be a bounded $\Pi^0_1$ class of reals with a computable minimal element $m$ and a computable maximal element $M$ and let the computably continuous function $F$ be given by (1) so that $F(x) = 0$ for all $x \in K$ and $F(x) > 0$ for all $x \notin K$. Now define the function $F$ in the following cases.

$G(x) = m + M - x$, for $x \leq m$.

$G(x) = M + F(x)$, for $m \leq x \leq M$.

$G(x) = 2x - M$ for $x \geq M$. □

Since any countable $\Pi^0_1$ subset of $[0, 1]$ and any $\Pi^0_1$ subset which may be represented by a tree with no dead ends has a computable member, we have the following immediate corollary.

**Corollary 14.0.31. (a)** *If the nonempty $\Pi^0_1$ subclass $K$ of [0,1] may be represented by a tree with no dead ends, then $K$ is the set of fixed points of some computably continuous function from [0,1] into [0,1].*

**(b)** *Any countable, nonempty $\Pi^0_1$ subclass $K$ is the set of fixed points of some computable function from [0,1] into [0,1].*

As usual, we have a number of immediate corollaries, of which we state only a few.

**Theorem 14.0.32.** *Let $X$ be $\{0, 1\}^\omega$, $\omega^\omega$, $\Re$, or $[0, 1]$.*

*(a) For any r.e. degree $\mathbf{c}$, there is a computably continuous function $F$ on $X$ such that the set of c. e. degrees which contain zeroes of $F$ equals the set of c. e. degrees $\geq_T \mathbf{c}$.*

*(b) There is a computably continuous function $F$ on $X$ which has a fixed point and such that any two distinct fixed points are Turing incomparable if $X$ is $\{0, 1\}^\omega$, $\omega^\omega$, $\Re$. There is a computably continuous function $F$ on $[0, 1]$ which has a unique computable fixed point and uncountable many non-computable fixed points and such that any two distinct non-computable fixed points are Turing incomparable.*

(c) *There is a computably continuous function $F$ which has a maximum $M$ on $X$, such that there is a unique non-comptable point $x_0$ where $M$ is attained and $x_0$ is also the unique accumulation point of the set where $M$ is attained.*

(d) *There is a computably continuous function on $\Re$ with attracting point at infinity such that every computable point is attracted to infinity but not every point is attracted to infinity.*

(e) *There is a computably continuous function on $[0,1]$ with a attracting point at infinity such that every computable point is attracted to infinity but not every point is attracted to infinity.*

*Proof.* Note that in part (b) when $X = [0,1]$, we may add a single computable point to the $\Pi^0_1$ class so that it can represent the set of fixed points. □

**Theorem 14.0.33.** *(a) There is a computably continuous function on $\omega^\omega$ which has a zero but has no hyperarithmetic zero.*

(b) *There is a computably continuous function on $\omega^\omega$ which attains a maximum $M$ such that $F(x) \neq M$ for any hyperarithmetic point $x$.*

(c) *There is a computably continuous function on $\omega^\omega$ which has a fixed point but has no hyperarithmetic fixed point.*

Ko [102] improved part (4) of Theorem 10.15 by showing that if the $\Pi^0_1$ class $P$ has either a p-time maximum element or a p-time minimum element, then there is a p-time computable function $f$ with Julia set $P$. Furthermore, Ko shows in [102] that there is such a set $P$ which has a non-computable Hausdorff dimension, which implies that there is a p-time computable function $f$ such that the Julia set of $f$ has non-computable Hausdorff dimension.

## 14.1 Symbolic Dynamics

In this section, we examine computable dynamical systems and symbolic dynamics associated with computable functions on the Cantor space $\{0,1\}^{\mathbb{N}}$ and the unit interval $[0,1]$.

Computable real dynamical systems have been studied by Cenzer [17], where the the Julia set of a computably continuous real function is shown to be a $\Pi^0_1$ class and Ko [102], who examined fractal dimensions and Julia sets. Computable complex dynamical systems have recently been investigated by Braverman and Cook [11] and Braverman and Yampolsky [12], who showed that there is a complex number $c$ such that the Julia set corresponding to the function $f(z) = z^2 + c$ is not decidable.

In particular, the computability of a closed set $K$ in a computable metric space $(X,d)$ may be defined in terms of the distance function $d_K$, where $d_K(x)$ is the infimum of $\{d(x,y) : y \in K\}$. $K$ is a $\Pi^0_1$ class if and only if $d_K$ is upper semi-computable and $K$ is a *decidable* (or *computable*) $\Pi^0_1$ class if $d_K$ is computable.

xxx

put this in earlier

zzz

For any finite $k$, the *shift* function on $\{0, 1, \dots, k\}$ is defined by $\sigma(x) = y$, where $y(n) = x(n+1)$. A closed set $Q \subseteq \{0, 1, \dots, k\}$ is said to be a *subshift* if it is closed under the shift function. We will refer to a $\Pi^0_1$ class which is also a subshift as a *subsimilar* $\Pi^0_1$ class.

Fix a finite alphabet $\Sigma$, let $F : \Sigma^{\mathbb{N}} \to \Sigma^{\mathbb{N}}$ be a computable function and let a partition $\{U_0, U_1, \dots, U_k\}$ of $\Sigma^{\mathbb{N}}$ into clopen sets be given. The *itinerary* of a point $x \in \Sigma^{\mathbb{N}}$ is the sequence $It(x) \in \{0, 1, \dots, k\}^{\mathbb{N}}$ where

$$It(x)(n) = i \iff F^n(x) \in U_i.$$

Now let $IT[F] = \{It(x) : x \in \Sigma^{\mathbb{N}}\}$. We show that $IT[F]$ is a decidable subsimilar $\Pi^0_1$ class and that, for any decidable subsimilar $\Pi^0_1$ class $Q \subseteq \Sigma^{\mathbb{N}}$, there exists a computable $F$ such that $Q = IT[F]$.

### 14.1.1 Undecidable subshifts

In this section, we construct a subsimilar $\Pi^0_1$ class with no computable element. We will give the construction in $\{0, 1\}^{\mathbb{N}}$, but it can be generalized to $\Sigma^{\mathbb{N}}$ for any finite $\Sigma$. Now every decidable $\Pi^0_1$ class has a computable element (in fact, the leftmost path is computable). Hence we have an undecidable subsimilar $\Pi^0_1$ class.

Let us say that a string $v$ is a *factor* of a string $w$ if there exist $w_1$ and $w_2$ such that $w = w_1{}^\frown v^\frown w_2$. For any set $S$ of strings, we may define a closed set $P_S$, where $x \in P_S$ if and only if, for all $n$ and all $w \in S$, $w$ is not a factor of $x\lceil n$. If the set $P_S$ is nonempty, then $S$ is said to be *avoidable*. For this section, we restrict ourselves to $\Sigma = \{0, 1\}$

**Lemma 14.1.1.** *Given any sequence $x_0, x_1, \dots$ of elements of $\{0, 1\}^{\mathbb{N}}$, there is a nonempty subshift containing no $x_i$.*

*Proof.* Define the sequence $l_0, l_1, \dots$ by $l_0 = 3$ and, for $n > 0$,

$$l_n = 3(2^{\frac{n(n+3)}{2}}).$$

This will imply that $l_{n+1} = 2^{n+2} l_n$. Now let $w_n = x_n\lceil 2l_n$ for each $n$ and define subshift $P$ to consist of all $x$ which do not contain any $w_n$ as a factor. Clearly $x_i \notin P$ for all $i$. It remains to show that $P$ is nonempty, that is, $\{w_n : n \in \mathbb{N}\}$ is avoidable.

It is important to notice that given any word $w$ of length $2k$, it has at most $k+1$ distinct factors of length $k$. Since there are $2^k$ words of length $k$, for $k$ large enough so that $2^k > k+1$, there are words of length $k$ that do not appear as a factor of $w$. With this in mind, we construct recursively two sequences of words $< A_n >_{n \in \mathbb{N}}$ and $< B_n >_{n \in \mathbb{N}}$ such that, for all $n$:

1. $|A_n| = |B_n| = l_n$.
2. $A_0$ and $B_0$ are not factors of $w_0$; this is possible since $|w_0| = 6$ so $w_0$ has at most 4 distinct factors of length 3.
3. $A_{n+1}$ and $B_{n+1}$ are taken from $\{A_n, B_n\}^*$, have $A_n$ as a prefix, and have length $m = 2^{n+2} = l_{n+1}/l_n$. This is possible since there are $2^{m-1}$ such words, but there are at most $l_{n+1} + 1$ factors of length $l_{n+1}$ in $w_{n+1}$ and $2^{m-1} \geqq l_{n+1} + 1 + 2$.

Now let $x = lim_n A_n$. This exists since each $A_n \prec A_{n+1}$. We claim that $x \in P$. Suppose by way of contradiction that some $w_n$ is a factor of $x$. We can view $x$ as an infinite concatenation of blocks length $l_n$, where each block is either $A_n$ or $B_n$. Since $w_n$ has length $2\ l_n$, it must completely contain one of the blocks, which would imply that either $A_n$ or $B_n$ is a factor of $w_n$. This contradiction shows that $x \in P$. □

We need to improve this lemma in two ways. First, we may have only a subset of words $w_k$ of length $l_{n_k}$. Second, we need an effective version.

**Theorem 14.1.2.** *There is a recursive sequence of natural numbers $l_0, l_1, \ldots$ such that if for any subsequence $< l_{n_k} >_{k\in\mathbb{N}}$ and any set $S = \{v_k : k \in \mathbb{N}\}$ of words such that $|v_k| = l_{n_k}$, $S$ is avoidable. Furthermore, if $\phi$ is a partial computable function such that $\phi(n_k) = v_k$, then there is a nonempty subsimilar $\Pi^0_1$ class $P$ such that no element of $P$ contains any factor $v_k$.*

*Proof.* For the first part, simply let $w_{n_k} = v_k$ and choose arbitrary words $w_i$ of length $l_i$ for $i \notin \{n_k : k \in \mathbb{N}\}$ and apply the lemma.

For the second part, we have

$$x \in P \iff (\forall n)(\forall k)[v_k \text{ is not a factor of } x\lceil n]$$

In more detail, notice that $v_k$ is not a factor of $x\lceil n$ if and only if, for all $v$, if $\phi_s(k) = v$, then $v$ is not a factor of $x\lceil n$. □

**Theorem 14.1.3.** *There is a nonempty subsimilar $\Pi^0_1$ class $P$ with no computable element.*

*Proof.* Let the sequence $< l_n >$ be given as in Lemma 14.1.1. Let Let $\phi_0, \phi_1, ..., \phi_e, \ldots$ be an enumeration of partial computable functions. Now define the partial recursive function $\phi$ by

$$\phi(k) = \begin{cases} \phi_k\lceil l_k, & \text{if } \phi_k(i)\downarrow \text{ for all } i < 2l_k; \\ \text{undefined}, & \text{otherwise.} \end{cases}$$

By Theorem 14.1.2, there is a nonempty subsimilar $\Pi^0_1$ class $P$ such that no element of $P$ has any word $\phi(k)$ as a factor. Now let $y$ be any computable element of $\{0,1\}^{\mathbb{N}}$. Then $y = \phi_k$ for some $k$ such that $\phi_k$ is a total function. Thus $\phi(k) = \phi_k\lceil k$ is defined and is not a factor of any $x \in P$ and hence certainly $\phi_k \notin P$. □

### 14.1.2 Symbolic Dynamics of Computable Functions

Fix a finite alphabet $\Sigma = \{0, 1, \dots, k\}$, let $F : \Sigma^{\mathbb{N}} \to \Sigma^{\mathbb{N}}$ be a computable function and let a partition $\{U_0, U_1, \dots, U_k\}$ of $\Sigma^{\mathbb{N}}$ into clopen sets be given. The *itinerary* of a point $x \in SN$ is the sequence $It(x) \in \{0, 1, \dots, k\}^{\mathbb{N}}$ where

$$It(x)(n) = i \iff F^n(x) \in U_i.$$

Now let $IT[F] = \{It(x) : x \in \Sigma^{\mathbb{N}}\}$. We observe that $IT[F]$ is a subshift. That is, suppose $y = It(x) \in IT[F]$. Then $\sigma(y) = It(F(x))$, so that $\sigma(y) \in IT[F]$ as well. The function $It$ is continuous and hence $IT[F]$ is a closed set, as seen by the proof of the following lemma.

**Lemma 14.1.4.** *The function from $\Sigma^{\mathbb{N}} \to \{0, 1, \dots, k\}^{\mathbb{N}}$ mapping $x$ to $I(x)$ is computable.*

*Proof.* Given clopen sets $U_0, \dots, U_k$, there exists a finite $j$ and a finite subset $W$ of $\{0, 1\}^j$ such that each $U_i$ is a finite union of intervals $I[w]$ for some set of $w \in W$. Thus one can determine from $y \lceil j$ the unique $i$ for which $y \in U_i$. Given $x \in \Sigma^{\mathbb{N}}$, let $y = I(x)$. To compute $y(n)$, it suffices to find the first $j$ values of $F^n(x)$, which can be computed uniformly from $x$ and $n$. □

**Theorem 14.1.5.** *Fix a computable function $F : \Sigma^{\mathbb{N}}$ to $\Sigma^{\mathbb{N}}$, let a partition $\{U_0, U_1, \dots, U_k\}$ of $\Sigma^{\mathbb{N}}$ into clopen sets be given and let $I(x)$ denote the itinerary of $x$ under $F$. Then*

*(a) For any computable $x \in \Sigma^{\mathbb{N}}$, $I(x)$ is computable.*

*(b) The set $IT[F]$ of itineraries is a decidable, subsimilar $\Pi^0_1$ class.*

*Proof.* Part (a) follows from the fact that computable functions map computable points to computable points and (b) follows from the fact that the image of a decidable $\Pi^0_1$ class under a computable function is a decidable $\Pi^0_1$ class. (See Section **??** for details.) □

Next we prove the converse. Note that $F^0(x) = x$ for all $x \in \Sigma^{\mathbb{N}}$ and therefore $IT[F]$ meets every $U_i$. Note that if $Q$ is a subshift and $Q$ does not meet $I[i]$, then $Q \subseteq \{0, 1, \dots, i-1, i+1, \dots, k\}$.

**Theorem 14.1.6.** *Let $\Sigma = \{0, 1, \dots, k\}$ be a finite alphabet and let $Q \subseteq \Sigma^{\mathbb{N}}$ be a decidable, subsimilar $\Pi^0_1$ class which meets $I[i]$ for all $i$. Then there exists a partition $\{U_0, \dots, U_k\}$ and a computable $F : \Sigma^{\mathbb{N}} \to \Sigma^{\mathbb{N}}$ such that $Q = IT[F]$.*

*Proof.* We will use the partition given by $U_i = I[i]$. Since $Q$ is decidable, we can define a function $G : \Sigma^{\mathbb{N}} \to Q$ such that $G(x) = x$ for all $x \in Q$. Let $Q = [T]$ where $T$ is a computable tree without dead ends. The approximating function $g$ for $G$ is defined as follows. For any $w \in \{0, 1, \dots, k\}^n$, find the longest initial segment $v$ such that $v \in T$ and let $g(v)$ be the lexicographically least (or leftmost) extension of $v$ which is in $T \cap \{0, 1, \dots, k\}^n$; this exists since $T$ has no dead ends. Now let $F(x) = \sigma(G(x))$. We claim that $\Sigma_F = Q$.

For any $x \in Q$, we have $F(x) = \sigma(x)$ and $\sigma(x) \in Q$, since $Q$ is a subshift. Hence $F^n(x) = \sigma^n(x)$, so that $F^{n+1}(x)(0) = x(n+1)$ and belongs to the set $U_{x(n)}$. Thus the itinerary $I(x) = x$. This shows that $Q \subseteq IT[F]$.

Next consider any $x \in \Sigma^{\mathbb{N}}$. We will show by induction that $F^n(x) = \sigma^n(G(x))$. For $n = 1$, this is the definition. Then

$$F^{n+1}(x) = \sigma(G(F^n(x))) = \sigma(G(\sigma^n(G(x)))),$$

by induction. But $G(x) \in Q$, so that $\sigma^n(G(x)) \in Q$ by subsimilarity and therefore $G(\sigma^n(G(x))) = \sigma^n(G(x))$ and finally $F^{n+1}(x) = \sigma^{n+1}(G(x))$, as desired. It follows that for $n > 0$, $It(x)(n) = G(x)(n)$. But for $n = 0$, the assumption that $Q$ meets $I[x(0)]$ implies that $G(x)(0) = x(0)$ and hence $It(x)(0) = x(0) = G(x)(0)$ as well. Therefore $It(x) \in Q$ as desired. $\square$

# Chapter 15

# Feasible versions of combinatorial problems

The main goal in this chapter is to apply the results of Chapter 7 to the mathematical problems discussed above in Chapters 8 to 14

We observe that any feasible structure is computable, therefore the set of solutions to a feasible problem is also the set of solutions to a computable problem. Thus results such as Theorems 8.2.1, 10.1.1 and 11.1.1 have feasible versions. The reverse direction is more interesting.

We consider computable representation theorems such as Theorems 8.2.5, 10.1.3 and 11.1.3, and corollary results such as Theorems 8.2.6, 10.1.5 and 11.1.4.

These representation theorems showed that the set of solutions to a computable problem of various sorts can represent either every c. b. $\Pi^0_1$ class or at least every $\Pi^0_1$ class of separating sets. In this section we obtain better results, in most cases, by improving "recursive" to "polynomial-time". Now an infinite computable problem may be assumed to have universe $\omega$, since any two infinite computable sets are recursively isomorphic. (Here the universe of a graph-coloring problem, for example, is the set the vertices.) However, it is not true that any two polynomial-time sets are polynomial-time isomorphic. (For example, it is clear that there is no p-time map from $Tal(\omega)$ onto $Bin(\omega)$.) Thus a polynomial-time structure with some p-time set for its universe may not be computably isomorphic to any p-time structure with universe $\omega$. For example, a p-time Abelian groups with all elements of finite order is constructed by Cenzer and Remmel in [31] which is not even isomorphic to any p-time group with standard universe either ($Tal(\omega)$ or $Bin(\omega)$). For many of the problems considered above, we will show that any computable problem can be reduced first to a p-time problem and then to a p-time problem with standard universe.

We illustrate the general strategy with the graph coloring problem. Recall from Section 10.10.2 that, for $k \geq 3$, the set of $k$-colorings of a recursive graph can be represented by a c. b. $\Pi^0_1$ class and conversely can represent an arbitrary c. b. $\Pi^0_1$ class. Let $G = (V, E)$ be a computable graph. Then the set of

$k$-colorings of $G$ can be represented as the $\Pi_1^0$ class $[T]$ of infinite paths through a computable tree $T$. Now Theorem 7.1.4 constructs for us a p-time tree $P$ such that $[T] = [P]$. Then the converse representation creates from $P$ a graph whose $k$-colorings are in an effective degree-preserving finite-to-one correspondence with the infinite paths through $P$. Furthermore, inspection of the proof from [161] shows that this graph will actually be polynomial time, since $P$ is polynomial time. This shows that the $k$-colorings of any computable graph can always be placed in an effective degree-preserving correspondence with the $k$-colorings of some p-time graph, and, therefore, that the $k$-colorings of a p-time graph can strongly represent any c. b. $\Pi_1^0$ class.

However, there is no natural correspondence between the recursive graph and the p-time graph constructed in this manner. We can do better using Theorem 7.2.1.

**Theorem 15.0.7.** *For each computable instance $P$ of any of the following problems, there is a p-time instance $Q$ of the problem which is computably isomorphic to $P$. Furthermore, except in cases (13) and (14), if $P$ has a computable solution, then we can take $Q$ to have a p-time solution.*

*(1) Finding a $k$-coloring for a $k$-colorable highly computable graph, for any $k \geq 3$.*

*(2) Finding a marriage in a highly computable society.*

*(3) Finding a surjective marriage in a symmetrically highly computable society.*

*(4) Finding a surjective marriage in a symmetrically highly computable society where each person knows at most two other people.*

*(5) Finding a $k$-partition of a highly computable graph such that no set in the partition is adjacent to $m$ other sets, for $m > 2$.*

*(6) Finding a one-way (or two-way) Hamiltonian (or Euler) path starting from a fixed vertex for a highly computable graph.*

*(7) Covering a computablee poset of width $k$ by $k$ chains, for any $k \geq 2$.*

*(8) Covering a computable poset of height $k$ by $k$ antichains, for any $k \geq 2$.*

*(9) Expressing a computable partial ordering on a set as the intersection of d linear orderings on the set.*

*(10) Finding a subordering of type $\omega$ (or of type $\omega^*$) of a computable ordering.*

*(11) Finding an $\omega$-successivity (or an $\omega^*$-successivity) in a computable linear ordering.*

*(12) Finding a non-trivial self-embedding of a computable linear ordering.*

*(13) Finding a winning strategy for an effectively closed binary game.*

*(14) Finding a prime ideal of a recursive Boolean algebra.*

*Proof.* For problems (1) through (12), this follows immediately from Theorem 7.2.1, since each of these problems can be viewed as a relational structure and the given solution can be viewed as a function mapping to a fixed range. In the dimension of posets problem, we can interpret the solution as a finite set of relations. For problem (13), Theorem 4.4 of [32] shows that any computable game may viewed as a p-time game in that the set of infinite paths which are winning for Player I will be the set of infinite paths through a p-time tree. For problem (14), Theorem 2.6 of [30] shows that any computable Boolean algebra is computably isomorphic to a p-time Boolean algebra. □

We note that a computable game with a computable winning strategy is not necessarily isomorphic to a p-time game with a p-time winning strategy, since by Theorem 4.5 of [32], there is a computable game with unique winning strategy, which is computable but not p-time.

**Corollary 15.0.8.** *For each recursive instance $P$ of any of the problems listed in Theorem 15.0.7, there is a p-time instance $Q$ of the problem such that the $\Pi^0_1$ class of solutions to $P$ is computably homeomorphic to the $\Pi^0_1$ class of solutions to $Q$.*

*Proof.* In each case, it is easy to see that the computable isomorphism between $P$ and $Q$ gives rise to a computable homeomorphism between the $\Pi^0_1$ classes of solutions.

For example, we consider the coloring problem. Recall that the $\Pi^0_1$ class of $k$-colorings on a computable graph $G = (V, E)$ (where $V$ may be assumed to equal $\omega$) is the set $[T]$ of infinite paths through the computable $k$-ary tree $T$, where a finite sequence $(\sigma(0), \ldots, \sigma(n-1)) \in \{1, 2, \ldots, k\}^n$ is in $T$ if and only if $\sigma(i) \neq \sigma(j)$ for all $(i, j) \in E$. Now suppose that $f$ is a computable isomorphism mapping $G$ to the computable graph $G' = (V', E')$, so that $V' = \{f(0), f(1), \ldots\}$ and $(f(i), f(j)) \in E'$ if and only if $(i, j) \in E$. Then we can define the tree $k+1$-ary $T'$ by having $(\tau(0), \ldots, \tau(n-1)) \in \{0, 1, \ldots, k\}$ in $T'$ if and only if

**(1)** $\tau(v) = 0 \iff v \notin V'$;

**(2)** $\tau(u) \neq \tau(v)$ whenever $(u, v) \in E'$.

Then $[T']$ represents in a reasonable way the set of legal $k$-colorings on $G'$ and we have a natural homeomorphism from $[T]$ to $[T']$ defined by $H(x)(f(i)) = x(i)$ and $H(x)(v) = 0$ if $v \notin V'$. □

We can now represent $\Pi^0_1$ classes as the set of solutions to p-time problems of the types listed above. We list only some of the results.

**Corollary 15.0.9.** *For each of the problems (1) through (9), and (13) listed in Theorem 15.0.7,* *(a) The problem of finding a computable solution to a p-time problem can strongly represent the c. b.* $\Pi^0_1$ *class of separating sets for any pair of disjoint infinite c. e. sets.*

*(b) There is a p-time instance of the problem with no computable solution.*

*(c) If* $\mathbf{a}$ *is a Turing degree and* $\mathbf{0} <_T \mathbf{a} <_T \mathbf{0}'$*, then there is a p-time instance* $P$ *of the problem such that* $P$ *has a solution of degree* $\mathbf{a}$ *but has no computable solution.*

*For problems (1), (3), (6) and (13), we have also:*

*(d) The problem of finding a computable solution to a p-time problem can strongly represent an arbitrary c. b.* $\Pi^0_1$ *class.*

*(e) There exists a p-time instance* $P$ *of the problem such that*

*(i)* $P$ *has a unique non-computable solution* $y$ *which is also the unique limit solution and has degree* $\mathbf{0}'$ *and such that any other solution is computable;*

*(ii) if* $R$ *is any computable sub-problem of* $P$ *and* $z$ *is any computable solution of* $R$*, then either (i) there are only finitely many solutions of* $P$ *which extend* $z$*, or (ii) all but finitely many solutions of* $P$ *extend* $z$*.*

*(iii) if* $x$ *is any computable solution of* $P$*, then there is some finite sub-problem* $F$ *of* $P$ *such that any solution of* $P$ *which agrees with* $x$ *on* $F$ *must equal* $x$*.*

We have seen that, by changing the names of the vertices, we can transform a computable graph into a p-time graph. However, we would prefer for a countably infinite graph to have the set $V$ of vertices equal to some standard universe such that the tally or binary representation of the set of natural numbers. This would, for instance, allow us to define the homeomorphism of Corollary 15.0.8 without worrying about the set of non-vertices. The p-time graph constructed by Theorem 7.1.4 will have a rather sparse set of vertices and this appears to be an essential part of the theorem. We will next indicate how to fill out the p-time structure given by Theorem 15.0.7 to a structure with universe $Bin(\omega)$ such that there is a degree-preserving correspondence, which is one-to-one (up to a finite permutation), between the $\Pi^0_1$ classes of solutions of the associated problems. For example, in the coloring problem, we add vertices whose colors will be determined, up to a permutation, by the coloring of the vertices of $Q$.

**Theorem 15.0.10.** *For each computable instance* $P$ *of the combinatorial problems listed below, there is a p-time instance* $Q$ *with universe* $Bin(\omega)$ *and a degree-preserving correspondence between the solutions of* $P$ *and the solutions of* $Q$*.*

*(1) Finding a* $k$*-coloring for a* $k$*-colorable highly computable graph, for any* $k \geq 3$*.*

*(2) Finding a marriage in a highly computable society.*

*(3) Finding a surjective marriage in a symmetrically highly computable society.*

*(4) Finding a surjective marriage in a symmetrically highly computable society where each person knows at most two other people.*

*(5) Finding a $k$-partition of a highly computable graph such that no set in the partition is adjacent to $m$ other sets.*

*(6) Finding a (one-way or two-way) Hamiltonian or Euler path for a highly computable graph.*

*(7) Covering a computable poset of width $k$ by $k$ chains, for any $k \geq 2$.*

*(8) Covering a computable poset of height $k$ by $k$ antichains, for any $k \geq 2$.*

*(9) Expressing a computable partial ordering on a set as the intersection of $d$ linear orderings on the set.*

*(10) Finding an $\omega$-successivity (or an $\omega^*$-successivity) in a computable linear ordering.*

*Proof.* In each case, we may assume by Corollary 15.0.8 that we start with a p-time instance of the problem which is a relational structure $\mathcal{B}$ with some universe $B \subseteq Bin(\omega)$. Now it follows from Lemma 2.3 of [31] that $\mathcal{B}$ is computably isomorphic to a p-time structure $\mathcal{A}$ with universe $A \subseteq Tal(\omega)$. Then Lemma 2.6 of [31] says that the disjoint union $A \oplus Bin(\omega)$ is p-time isomorphic to $Bin(\omega)$, where $X \oplus Y = \{\langle 0, x\rangle : x \in X\} \cup \{\langle 1, y\rangle : y \in Y\}$. Then we will create a p-time structure $\mathcal{C}$ with universe $A \oplus Bin(\omega)$ which has a copy of $\mathcal{A}$ together with a copy of $Bin(\omega)$, where the relations will be defined on $Bin(\omega)$ and between $A$ and $Bin(\omega)$ so as to determine the degree-preserving correspondence between the solutions of $\mathcal{A}$ and those of the extension $\mathcal{C}$. Since the universe $C$ of $\mathcal{C}$ is p-time isomorphic to $Bin(\omega)$, it follows from Lemma 2.2 of [31] that $\mathcal{C}$ is computably isomorphic to a p-time structure with universe $Bin(\omega)$. Then we will let $Q$ be the problem associated with this structure. It follows that there will be a degree preserving correspondence between the set of solutions of $Q$ and the set of solutions of the original problem $P$. In each case, we will assume that our original structure is p-time and has for its universe a p-time subset $A$ of $Tal(\omega)$ and that there is a p-time list of $Bin(\omega) \setminus A$. These assumptions are justified by the above discussion. In each case, the correspondence will be one-to-one unless otherwise indicated.

(1) Finding a $k$-coloring for a $k$-colorable highly computable graph, for any $k \geq 3$.

This is Theorem 2.1 of [33]. Here the correspondence is one-to-one, up to a finite permutation of the colors on the new vertices.

(2) Finding a marriage in a highly computable society.

Let $S = (B, G, K)$ be a p-time society. Then we will directly extend $S$ to a highly recursive p-time society $S' = (B', G', K')$ where $B' = G' = Bin(\omega)$. Let $Bin(\omega) \setminus B = \{b_0, b_1, \ldots\}$ and let $Bin(\omega) \setminus G = \{g_0, g_1, \ldots\}$ be p-time lists of the new boys and girls in the society $S'$. Then $K'$ is defined by putting $(b_i, g_i) \in K'$ for all $i$. It is clear that any marriage $f$ on $S$ has a unique extension $f'$ to $S'$ defined by letting $f'(b_i) = g_i$ for all $i$. It follows that $f$ and $f'$ have the same degree.

(3) Finding a surjective marriage in a symmetrically highly computable society.

The extension is the same as in (2). It is clear that $f'$ will be onto if and only if $f$ is onto.

(4) Finding a surjective marriage in a symmetrically highly computable society where each person knows at most two other people.

The extension is again the same as in (2). It is clear that if each person in $S$ knows at most two other people, then each person in the extension $S'$ also knows at most two other people.

(5) Finding a $k$-partition of a highly computable graph such that no set in the partition is adjacent to $m$ other sets, with $m > 2$.

Let the p-time graph $G = (V, E)$ be given. We define a p-time graph $G_1 = (V_1, E_1)$ to be a regular $m-1$-ary tree of complete $k$-graphs. That is, define the regular $(m-1)$-ary tree $T_{m-1}$ to consist of a root node $\emptyset$ together with the set $\{bin(0), bin(1), \ldots, bin(m-1)\} \times \{bin(1), bin(2), \ldots, bin(m-2)\}^*$, where $\emptyset$ has $m-1$ successors $(bin(i), \emptyset)$ for $i < m$ and $(bin(i), \sigma)$ has $m-2$ successors $(bin(i), \sigma^\frown bin(j))$ for $j < m-1$. Then we let

$$V_1 = \{bin(1), bin(2), \ldots, bin(k)\} \times T_{m-1},$$

and we let $((bin(i), \sigma), (bin(j), \tau)) \in E_1$, where $\sigma = (\sigma(0), \ldots, \sigma(s-1))$ and $\tau = (\tau(0), \ldots, \tau(t-1))$, provided that $\sigma = \tau$ or either $\tau$ is a successor of $\sigma$ or $\sigma$ is a successor of $\tau$. It is clear that if the graph is computably partitioned into the complete $k$-graphs corresponding to the nodes of $T_{m-1}$, then each set in the partition is adjacent to at most $m-1$ other sets. We see also that each node of $T_{m-1}$ has $m-1$ neighbors, so that any two distinct nodes have at least $2m-4$ other neighbors. Now let $\{A_i : i < \omega\}$ be a $k$-partition of $G_1$. Suppose that some $A_i$ contains vertices $u$ and $v$ corresponding to different nodes of the $T_{m-1}$. Then $u$ and $v$ taken together have at least $2(k-1) + (2m-4)k = (2m-2)k - 2$ other adjacent vertices in $G_1$. Since $k-2$ of these could belong to $A_i$, we see that $A_i$ has at least $(2m-3)k$ adjacent vertices. Since each set in the partition has at most $k$ vertices, it follows that $A_i$ is adjacent to at least $2m-3$ sets in the partition. Thus since $m > 2$, we have $2m-3 > m-1$ so that the set $A_i$ is adjacent to too many sets.

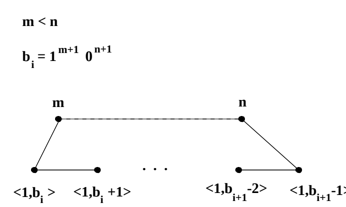
Now let $G' = G \oplus G_1$. It is clear that for any $k$-partition of $G$, there is a partition of $G'$ of the same degree which is given by adding the recursive partition of $G_1$ into the nodes of the tree as defined above. We claim that these are the only possible partitions. That is, suppose $\{B_j : j < \omega\}$ is a $k$-partition of $G'$. It suffices to show that for any $u \in V_1$ and any $j$, if $u \in B_j$, then the entire node to which $u$ belongs must be included in $B_j$. Suppose that this is false. It follows from the argument above that $B_j$ may not contain an element of a different node of $T$. Thus the set $B_j$ has all $k(m-1)$ vertices from the adjacent nodes as neighbors as well as at least one vertex from the node of $u$. But this clearly implies that at least $m$ sets of the partition must be adjacent to $B_j$.

(6) Finding a one-way (or two-way) Hamiltonian Euler path starting from a fixed vertex for a highly recursive graph.

Let the p-time graph $G = (V, E)$ be given with $V = \{v_0 < v_1 < \cdots\}$ a subset of $Tal(\omega)$. Let each edge $(tal(m), tal(n))$ of $V$ with $tal(m) < tal(n)$ be coded as $0^{n+1}1^{m+1}$ in $Bin(\omega)$. Let $b_0, b_1, \ldots$ enumerate the codes of edges in increasing order and let $b_i = 0^{n_i+1}1^{m_i+1}$ for each $i$. Now let $V' = V \oplus Bin(\omega)$ and let $E'$ be defined by joining $\langle 1, b_i\rangle$ to $\langle 0, tal(m_i)\rangle$, joining $\langle 0, tal(n_i)\rangle$ to $\langle 1, b_{i+1}-1\rangle$, joining $\langle 1, b\rangle$ to $\langle 1, b+1\rangle$ whenever $b+1 \neq b_i$ for any $i$, and joining $\langle 1, b_0-1\rangle$ to $\langle 0, tal(m_0)\rangle$. Note that other than the initial sequence of edges connecting $\langle 1, 0\rangle$ to $\langle 1, b_0-1\rangle$ and then to $\langle 0, tal(m_0)\rangle$, this has the effect of replacing an edge $(m, n) \in E$ where $m < n$ and $b_i = 1^{m+1}0^{n+1}$ by a sequence of edges $(\langle 0, m\rangle, \langle 1, b_i\rangle), (\langle 1, b_i\rangle, \langle 1, b_i+1\rangle), \ldots, (\langle 1, b_{i+1}-2\rangle, \langle 1, b_{i+1}-1\rangle), (\langle 1, b_{i+1}-1\rangle, \langle 0, n\rangle)$. See Figure 15.

Thus to test whether $\langle 1, b\rangle$ and $\langle 1, c\rangle$ are joined by an edge, where $b < c$, we simply check that $c = b+1$ and that, if $c = 0^{n+1}1^{m+1}$ with $m < n$, then $(tal(m), tal(n)) \notin E$. To determine whether $\langle 0, v\rangle$ and $\langle 1, c\rangle$ are joined by an edge, we first check that $v = tal(m) \in V$ and that either (i) $c = 0^{s+1}1^{r+1}$ or (ii) $c = 0^{s+1}1^{r+1} - 1$ for some edge $(tal(r), tal(s))$ in $G$ with $r < s$. Finally, in case (i), we check that $r = m$ and, in case (ii), we either check that $m = m_0$ and that $c+1 = b_0$ or else we compute the largest code $0^{q+1}1^{p+1}$ of an edge of $G$ less than $c+1$ and check that $m = q$. If everything checks, then there is an edge and otherwise there is not. Thus $G'$ is a p-time graph.

Now suppose that $f$ is a one-way Euler path on $G$ starting from $f(0) = v_0 = tal(m_0)$. Then we can define a corresponding Euler path on $G'$ starting from $\langle 1, bin(0)\rangle$ by beginning with the sequence
$\langle 1, bin(0)\rangle, \langle 1, bin(1)\rangle, \ldots, \langle 1, b_0-1\rangle, \langle 0, v_0\rangle$ and then replacing in turn each edge $(f(i), f(i+1))$ which joins $tal(m)$ to $tal(n)$ with $m < n$, either by the sequence $\langle 0, f(i)\rangle, \langle 1, bin(b)\rangle, \langle 1, bin(b+1)\rangle, \ldots, \langle 1, bin(c-1)\rangle, \langle 0, f(i+1)\rangle$, if $f(i) = tal(m) < f(i+1)$, or by the sequence $\langle 0, f(i)\rangle, \langle 1, bin(c-1)\rangle, \langle 1, bin(c-2)\rangle, \ldots, \langle 1, bin(b+1)\rangle, \langle 1, bin(b)\rangle, \langle 0, f(i+1)\rangle$ if $f(i) = tal(n) > f(i+1)$, where $b = 0^{n+1}1^{m+1}$ and $c$ is the least code greater than $b$ for an edge of $G$. It is clear


m < n
$b_i = 1^{m+1}\ 0^{n+1}$
m
n
. . .
<1,$b_i$ >
<1,$b_i$ +1>
<1,$b_{i+1}$-2>
<1,$b_{i+1}$-1>

that this one-way Euler path is computable in $f$.

Conversely, let $g$ be a one-way Euler path in $G'$ starting from $\langle 1, bin(0)\rangle$. It is clear that the path must proceed through
$\langle 1, bin(1)\rangle, \langle 1, bin(2)\rangle, \ldots, \langle 1, b_0 - 1\rangle$ and then to $\langle 0, v_0\rangle$. Now let $f : \omega \to V$ be defined by letting $\langle 0, f(i)\rangle$ be the $i$-th vertex of the form $\langle 0, x\rangle$ in the path $g$. Then $f$ is a one-way Euler path for $G$ and it follows from the construction that $g$ is the corresponding path as defined above, since there is only one way in $G'$ to go from $\langle 0, f(i)\rangle$ to $\langle 0, f(i+1)\rangle$.

For two-way Euler paths, modify the construction by eliminating the finite initial sequence $\langle 1, bin(0)\rangle, \langle 1, bin(1), \ldots, \langle 1, bin(m_0) - 1\rangle$ of vertices of $G'$ along with the edges through those vertices. Then the remaining vertex set is still p-time isomorphic to $Bin(\omega)$ and the argument goes through as above.

The Hamiltonian paths require a different construction. We assume without loss of generality that the vertices of $G$ include 0 and that all are multiples of 4 (in binary) and let $4m_0, 4m_1, \ldots$ enumerate the vertices of $G$ in increasing order. Define the graph $G'$ to have vertex set $V' = Bin(\omega)$ with edges defined as follows. For each $i$, there will be two sequences of edges joining the set of binary numbers from $4m_i + 1$ up to $4m_{i+1}$, as follows:

**(i)** $4m_{i+1}, 4m_{i+1} - 4, 4m_{i+1} - 8, \ldots, 4m_i + 4, 4m_i + 2, 4m_i + 6, \ldots, 4m_{i+1} - 2,$

**(ii)** $4m_{i+1}, 4m_{i+1} - 3, 4m_{i+1} - 7, \ldots, 4m_i + 1, 4m_i + 3, \ldots, 4m_{i+1} - 1.$

These are the vertices associated with $4m_{i+1}$. In addition, for each edge joining $4m_i$ and $4m_j$ in $G$ with $m_i, m_j \neq 0$, there are edges joining $4m_i - 1$ with $4m_j - 2$ and joining $4m_i - 2$ with $4m_j - 1$. For an edge in $G$ joining $4m_i$ with 0, there is an edge joining $4m_i - 1$ with 0. The procedure for determining whether there is an edge joining $a$ and $b$ is the following. First look for the largest $m$ and $n$ such that $4m \in V$ and $4m < a$ and $4n \in V$ and $4n < b$. In the special case that $a = 0$, $a$ and $b$ are joined if and only if $b + 1$ is joined to 0 as vertices in $G$. Otherwise there are several cases. First suppose that $m = n$; then $a$ are $b$ are joined if and only if, either they differ by exactly 4 or $\{a, b\} = \{4m + 1, 4m + 3\}$ or $\{a, b\} = \{4m + 2, 4m + 4\}$. Next suppose that $m \neq n$. Then $a$ and $b$ are joined if and only if either $a + 1$ and $b + 2$ are joined as vertices in $G$ or $a + 2$ and $b + 1$ are joined as vertices in $G$. Thus $G'$ is a p-time graph.

Now let $f$ be a one-way Hamiltonian path on $G$ starting from $v_0 = 0$ and suppose that $f(i) = 4m_{r_i}$. Then there is a corresponding Hamiltonian path $g$ in $G'$ obtained by replacing the edge from $v_0$ to $4m_{r_1}$ with the sequence of edges joining $v_0$ to $4m_{r_1} - 1$ and then on to $4m_{r_1} - 3$ and $4m_{r_1}$ as described above, and for $i > 0$, replacing each edge $(f(i), f(i+1))$ with the sequence of edges first joining $4m_{r_i}$ to $4m_{r_i} - 4$ and then on to $4m_{r_i} - 2$ as described above, then joining $4m_{r_i} - 2$ to $4m_{r_{i+1}} - 1$, and closing with the sequence joining $4m_{i+1} - 1$ to $4m_{i+1} - 3$ and then $4m_{i+1}$. Thus for each $i > 0$, the even vertices associated with $f(i)$ are joined to the odd vertices associated with $f(i+1)$.

Conversely, let $g$ be a one-way Hamiltonian path in $G'$ starting from $v_0 = 0$ and define $f(i) = 4m_{r_i}$ so that $0, 4m_{r_1}, \ldots$ lists the members of $G$ in order of appearance in the path $g$. It follows from the construction that $f$ is a one-way

Hamiltonian path for $G'$ starting from $v_0$ and that $g$ is the corresponding path as defined above.

For the two-way Hamiltonian paths, the construction is modified by adding an edge joining $v_0$ with $4m_i - 2$ for each edge joining $v_0$ with $4m_i$ in $G$. Then for any two way Hamiltonian path $f$ in $G$, there will be two corresponding two-way paths in $G'$, one in which the even vertices associated with $f(i)$ are joined to the odd vertices associated with $f(i+1)$ and one in which the odd vertices associated with $f(i)$ are joined with the even vertices of $f(i+1)$. Thus the correspondence here is two-to-one.

(7) The problem of covering a computable poset of width $k$ by $k$ chains, for any $k \geq 2$.

Let $\mathcal{P} = (P, \leq_P)$ be a p-time poset where $P \subseteq Tal(\omega)$. Then define a p-time poset $\mathcal{R} = (R, \leq_R)$ where $R = P \oplus (\{bin(1), \dots, bin(k)\} \times Bin(\omega))$ and $\langle 0, p\rangle \leq_R \langle 0, q\rangle$ iff $p \leq_P q$, $\langle 0, p\rangle \leq_R \langle 1, n\rangle$ for all $p$ and $n$, and
$\langle 1, m\rangle \leq_R \langle 1, n\rangle$ iff $m = \langle bin(i), bin(r)\rangle$ and $n = \langle bin(i), bin(s)\rangle$ where $r \leq s$. Then it is clear that for any covering $f$ of $\mathcal{P}$ by $k$ chains induces a covering $f'$ of covering of $\mathcal{R}$ by $k$ chains where

**(i)** $f'(\langle 0, p\rangle) = f(p)$ for all $p \in P$ and

**(ii)** $f'(\langle 1, \langle bin(i), n\rangle\rangle) = f'(\langle 1, \langle bin(i), m\rangle\rangle)$ for all $i$,$m$ and $n$.

Thus the covering is determined by the value of $f'$ on the finitely many new points $\langle 1, \langle bin(1), 0\rangle\rangle, \dots, \langle 1, \langle bin(1)0\rangle\rangle$. This shows that $f'$ has the same degree as $f$ and that $f'$ is unique up to a permutation of the names of chains. Then $\mathcal{R}$ is p-time isomorphic to p-time linear ordering $\mathcal{S}$ whose universe is $Bin(\omega)$.

(8) The problem of covering a computable poset of height $k$ by $k$ antichains, for any $k \geq 2$.

This is the dual of problem (7). The partial order is now defined by making $\langle 1, \langle bin(i), n\rangle\rangle \leq \langle 1, \langle bin(j), m\rangle\rangle \iff (i < j \;\&\; m = n)$.

(9) The dimension of posets problem.

Let $\mathcal{P} = (P, \leq_P)$ be a poset and let $Bin(\omega) \setminus P = \{v_i : i < \omega\}$. The partial order $\leq'$ is defined on $Bin(\omega)$ by making $p \leq' v_i$ for all $p \in P$ and all $i$ and making $v_i \leq' v_j$ if and only if $v_i \leq v_j$ (where $<$ is the usual ordering on $Bin(\omega)$.

(10) Finding an $\omega$-successivity (or an $\omega^*$-successivity) in a computable linear ordering.

Given a p-time linear ordering $L_1 = (A, <_1)$ on a p-time set $A = \{a_0 < a_1 < \cdots\}$, we may assume $a_0 = 0$ and that each $a_i = bin(4m_i)$. Now define the p-time ordering $L_2 = (Bin(\omega), <_2)$ by replacing each point $a = bin(4m_i)$ with a block $B(a)$:

$$\begin{aligned}
&bin(4m_i + 1) < bin(4m_i + 5) < \cdots < bin(4m_{i+1} - 3)\\
&< bin(4m_{i+1} - 1) < bin(4m_{i+1} - 5) < \cdots < bin(4m_i + 3)\\
&< bin(4m_i) < bin(4m_i + 4) < bin(4m_i + 8) < \cdots < bin(4m_{i+1} - 4)\\
&< bin(4m_{i+1} - 2) < bin(4m_{i+1} - 6) < \cdots < bin(4m_i + 2)
\end{aligned}$$

That is, we use the elements between $4m_i$ and $4m_{i+1}$ which are equivalent to 1 mod 4 to form a chain between $4m_i+1$ and $4m_{i+1}-1$, then we use the elements between $4m_i$ and $4m_{i+1}$ which are equivalent to 3 mod 4 in reverse order to form a chain between $4m_{i+1}-1$ and $4m_i$, etc.

Now suppose that $f$ is an $\omega$-successivity in $L_1$. Then we can recursively obtain an $\omega$-successivity $g$ in $L_2$ by replacing each point $f(i)$ with the block $B(f(i))$. Conversely, given an $\omega$-successivity $g$ in $L_2$, the $\omega$-successivity $f$ of $L_1$ may be defined by making $f(i)$ the $i$-th binary number in the successivity $g$ which is divisible by 4 and it then follows that $g$ is the successivity obtained from $f$ as above. The argument for $\omega^*$-successivities is similar. □

We remark that, for the three-coloring problem, it is possible to improve this result by having the 3-colorings of the the original computable graph be restrictions of the 3-colorings of the p-time graph to the original recursive vertex set.

Theorem 15.0.10 can now be applied to obtain improved versions of Corollary 15.0.9. We list only a few here.

**Corollary 15.0.11.** *(a) There exists a p-time graph $G$ with universe $Bin(\omega)$ which has a unique non-computable Hamiltonian path $\pi$, where $\pi$ has degree $\mathbf{0}'$ and such that any other Hamiltonian path is the unique extension to $G$ of a Hamiltonian path on some finite subgraph $F$ of $G$.*

*(b) There is a p-time partial ordering with universe $Bin(\omega)$ of width $k$ which has no computable covering by $k$ chains.*

*(c) For any $x \leq_T \mathbf{0}'$, there is a p-time linear ordering $\mathcal{A}$ with universe $Bin(\omega)$ such that there is $\omega$-successivity (respectively $\omega^*$-successivity) of $\mathcal{A}$ of degree $x$ and every $\omega$-successivity (respectively $\omega^*$-successivity) of $\mathcal{A}$ is either computable or has the same Turing degree as $x$.*

Theorem 15.0.10 and Corollary 15.0.11 demonstrate that the problem of finding solutions to feasible problems is just as difficult as the problem of finding solutions to recursive problems. Therefore more conditions will have to be put on a problem than just feasibility if our goal is to guarantee the existence of a feasible solution, or even the existence of a recursive solution. There are many possible approaches to this goal, some of which were explored in [33] for the graph-coloring problem.

Finally, we consider the problem of finding a prime ideal of a recursive Boolean algebra, or more generally, of a recursive ring.

**Theorem 15.0.12.** *For any recursive Boolean algebra $\mathcal{B}$, there is a p-time commutative ring $\mathcal{R}$ with unity, having universe $Bin(\omega)$, and a one-to-one degree-preserving map between the class of prime ideals of $\mathcal{R}$ and the class of prime ideals of $\mathcal{B}$.*

*Proof.* By Theorem 15.0.7, we may assume that $\mathcal{B}$ is a p-time Boolean algebra, and thus a Boolean ring. Now define the ring $\mathcal{R} = \mathcal{B} \oplus \mathbf{Q}$. $\mathbf{Q}$ is chosen here because it has no (proper) prime ideals. The ring $\mathbf{Q}$ of rationals may be represented as a p-time ring with universe $Bin(\omega)$ and it follows from Lemmas 2.2 and 2.6 of [31] that $\mathcal{R}$ is p-time isomorphic to a ring with universe $Bin(\omega)$. For any prime ideal $I$ of $\mathcal{B}$, it is easy to check that $I \oplus \mathbf{Q}$ is a prime ideal of $\mathcal{R}$ and that these are the only prime ideals of $\mathcal{R}$. □

**Corollary 15.0.13.** *(i) For any degree $\mathbf{a} <_T \mathbf{0}'$, there exists a recursive commutative ring $\mathcal{R}$ with a prime ideal $I$ of degree $\mathbf{a}$ such that $I$ is the unique non-recursive prime ideal of $\mathcal{B}$ and such that any other prime ideal of $\mathcal{B}$ is finitely generated.*

*(ii) There is a computable commutative ring with unity, $\mathcal{R}$, which has a unique non-computable prime ideal $I$, such that any other prime ideal of $\mathcal{R}$ is finitely generated, and such that for any c. e. ideal $J$ of $\mathcal{R}$, either there are only finitely many prime ideals of $\mathcal{R}$ extending $J$ or else all but finitely many of the prime ideals of $\mathcal{R}$ extend $J$.*

# Part C

# Advanced Topics and Current Research Areas

# Chapter 16

# The Lattice of $\Pi^0_1$ classes

The study of the lattice $\mathcal{E}$ of computably enumerable sets under inclusion has been one of the central tasks of computability theory since the 1960s. The inclusion lattice $\mathcal{E}_\Pi$ of $\Pi^0_1$ classes has an interesting algebraic structure, in some ways analogous to the dual of the lattice $\mathcal{E}$ of c.e. sets. Recent work has focused on comparing and contrasting the two lattices. Important issues include the definability and complexity of various properties, automorphisms of certain substructures of the lattice.

Here is an example. Given two $\Pi^0_1$classes $P \subset Q$, the interval $[P, Q] = \{R : P \subseteq R \subseteq Q\}$ of $P$ and in particular $[\emptyset, Q]$ is an *initial segment* of $\mathcal{E}_\Pi$. A $\Pi^0_1$class $P$ is said to be *thin* if $[\emptyset, Q]$ is a Boolean algebra. $P$ is *perfect* if every element of $P$ is a limit point. Cholak et al. [45] have shown that the family of all perfect thin classes is in certain ways analogous to the hyper-hypersimple c.e. sets. That is, any two perfect thin classes are automorphic in $\mathcal{E}_\Pi$, the family of perfect thin classes is definable in $\mathcal{E}_\Pi$ and the degrees of perfect thin classes are exactly the c.e. array noncomputable degrees. (Here the degree of $P = [T]$ is the degree of the set of nodes of $T$ which have an extension in $P$.)

An infinite $\Pi^0_1$ class $P$ is *minimal* if every $\Pi^0_1$ subclass of $P$ is either finite or cofinite in $P$. This is of course dual to the notion of a maximal c.e. set. For any lattice $\mathcal{L}$, let $\mathcal{L}^*$ be the quotient lattice of $\mathcal{L}$ modulo finite difference. Then $P$ is minimal if and only if $[0, P]^*$ is the trivial Boolean algebra. Cenzer, Downey, Jockusch and Shore [22] first constructed a minimal thin class. Cenzer and Nies [28] characterized the order types of the finite intervals of $\mathcal{E}^*_\Pi$ as finite distributive lattices with the dual reduction property. Furthermore, for each such lattice $L$, the theory of $L$ is decidable. In particular, this means that there are intervals (in fact, initial segments) of order type $n$ for any finite ordinal $n$. This contrasts with the classic result that finite intervals of $\mathcal{E}^*$ are all Boolean algebras. However, it is shown in [28] that for any *decidable* $\Pi^0_1$ class $P$, if $[0, P]^*$ is finite, then it must be a Boolean algebra. Finally, if $P$ is decidable and $[0, P]$ is not a Boolean algebra, then the theory of $[0, P]$ interprets the theory of arithmetic and is therefore undecidable.

End intervals $[P, TN]$ were studied in [29] where it was shown that there are exactly two possible isomorphism types of *end* intervals (where $P$ is either clopen or not).

In his thesis (and continued in joint work with Cenzer), Riazati [164, 41] studied minimal extensions of $\Pi^0_1$ classes (an analogue of maximal subsets). He proved an analogue of the Owings Splitting Theorem and used it to prove that decidable minimal extensions are not possible.

Lawton [114] introduced the notion of *minor* superclasses of $\Pi^0_1$classes, as an analogue of major subsets of c.e. sets and gave a characterization of the $\Pi^0_1$ classes which have strong minor superclasses.

It is easy to see that the family of finite classes is invariant under automorphism. Cenzer and Nies [29] showed that the property of being finite is definable in the lattice and in general, the family of countable $\Pi^0_1$ classes of rank $\alpha$ is definable if and only if $\alpha < \omega$.

## 16.1 The dual lattice of c. e. ideals of $\mathcal{Q}$

An *ideal* $I$ of the Boolean algebra $B$ is a subset which is closed under $\vee$ and under $\preceq$, that is, if $a, b \in I$ and $c \preceq b$, then $a \vee b \in I$ and $c \in I$.

For any $\Pi^0_1$ class $P \subset \{0,1\}^{\mathbb{N}}$, let

$$I(P) = \{U \in \mathcal{Q} : U \cap P = \emptyset\}.$$

Then $I(P)$ is a c. e. ideal of $\mathcal{Q}$.

We will make use of $\Sigma^0_k$-Boolean algebras, which may be similarly defined by requiring that $\preceq$ be $\Sigma^0_k$ and that $\wedge, \vee$ be computable in $\mathbf{0}^{(k-1)}$. For a $\Sigma^0_k$ Boolean algebra $\mathcal{B}$, let

$$\mathcal{I}(\mathcal{B}) \ := \ \text{the lattice of } \Sigma^0_k\text{–ideals of } \mathcal{B}.$$

The lattice operations of $\mathcal{I}(\mathcal{B})$ are given by

**(a)** $I \wedge J = I \cap J$ and

**(b)** $I \vee J = \{a \in \mathcal{Q} : (\exists b \in I)(\exists c \in J)(a \leq b \cup c)$.

We claim that the map taking a $\Pi^0_1$ class $P$ to $I(P)$ defines an effective isomorphism from $\mathcal{E}_\Pi$ onto $\mathcal{I}(\mathcal{Q})$, where the meet and join are reversed. Details are left to the exercises.

**Proposition 16.1.1.** *$\mathcal{E}_\Pi$ and $\mathcal{I}(\mathcal{Q})$ are effectively (reverse) isomorphic.* $\square$

Recall that an ideal $I$ in a Boolean algebra $B$ is said to be *principal* if there is some $b$ such that $I = \{a : a \leq b\}$. The ideal $I$ corresponding to a $\Pi^0_1$ class $P$ as above is principal if and only if $P$ is clopen. Thus we will refer to a non-clopen $\Pi^0_1$ class $P$ as *nonprincipal*. For any $\Pi^0_1$ class $P$, let $S(P)$ be the lattice of $\Pi^0_1$ classes $Q$ such that $P \subset Q$.

**Exercises**

16.1.1. Explain concretely how the meet and join operations of $\mathcal{B}$ are computed.

16.1.2. Show that $I(P)$ is a c.e. ideal of $\mathcal{Q}$ for any $\Pi_1^0$ class $P \subset \{0,1\}^{\mathbb{N}}$.

16.1.3. Show that $I(P \cup Q) = I(P) \cap I(Q)$ and $I(P \cap Q) = I(P) \vee I(Q)$. HINT: The second part will use the separation property of $\Pi_1^0$ classes from Corollary 2.2.24.)

16.1.4. Show that for any ideal $I \in \mathcal{I}(\mathcal{Q})$, $P = \{0,1\}^{\mathbb{N}} \setminus \cup I$ is a $\Pi_1^0$ class and $I(P) = I$. (Hint: use compactness.)

## 16.2 Countable thin classes

**Theorem 16.2.1.** *(C-D-J-S) For any computable ordinal $\alpha$, there is a thin $\Pi_1^0$ class $P_\alpha$ with Cantor-Bendixson rank $\alpha$. Furthermore, we may take $P_\alpha$ as the set of paths through a computable tree with no dead ends.*

*Proof.* We first sketch the proof for $\alpha = 1$. We construct a sequence $\tau_e \in \{0,1\}^{<\omega}$ such that $\tau_e^\frown 1 \prec \tau_{e+1}$ for all $e$, a set $A = \cup_e \tau_e$ and a $\Pi_1^0$ class $P = [T]$ such that

(1) $D(P) = \{A\}$, and

(2) for any $e$, if $A \in [T_e]$ then $P \cap I(\tau_e) \subset [T_e]$.

These conditions imply that $A$ is non-computable, since if $A$ were computable, then $\{A\} = [T_e]$ for some $e$, so that by (2), $P \cap I(\tau_e) = \{A\}$, contradicting (1).

These conditions imply that $A$ is non-computable, since if $A$ were computable, then $\{A\} = [T_e]$ for some $e$, so that by (2), $P \cap I(\tau_e) = \{A\}$, contradicting (1).

These conditions also imply that $P$ is minimal (and therefore thin by Theorem 5.2). To see this, suppose that $[T_e] \subset P$. If $A \notin [T_e]$, then $[T_e]$ has no limit point and is therefore finite. If $A \in [T_e]$, then $P \setminus [T_e] \subset P \setminus I(\tau_e)$ by (2) has no limit point and is finite.

The construction is in stages, so that at stage $s$ we have a tree $T^s$ and strings $\tau_e^s$. At stage $s+1$, we simply look for $e \leq s$ such that some $\tau \succ \tau_e^s$ is in $T^s \setminus T_e$ and let $\tau_e^{s+1} = \tau$ for the least such $e$ and $\tau$. For $i < e$, let $\tau_i^{s+1} = \tau_i^s$ and for $\tau_{e+i}^{s+1} = 0\tau^\frown 1^i$. We leave the details to the reader.

The general construction for a computable ordinal $\alpha$ is accomplished using a computable system of notations for $\alpha$ and a uniformly computable family of trees of rank up to $\alpha$ as in the proof of Theorem 4.4.8. The details are omitted. $\square$

Next we consider the possible degrees of members of thin $\Pi_1^0$ classes.

**Theorem 16.2.2.** *(C-D-J-S) There is a $\Pi_1^0$ set $A$ of degree $\mathbf{0}'$ and a minimal, thin $\Pi_1^0$ class $P$ such that $D(P) = \{A\}$.*

*Proof.* Let $B = \mathbf{0}'$ be the union of uniformly computable sets $B^s$. Let $T_0, T_1, \dots$ be an effective enumeration of the primitive recursive trees on $2^{<\omega}$. We will define a $\Pi^0_1$ retraceable set $A = \{a_0 < a_1 < \cdots\}$ and a corresponding $\Pi^0_1$ class $P = I(A)$ of initial subsets of $A$, by Theorem 2.6.2, such that

(1) For any $e$, $e \in B \iff e \in B^{a_e}$.

(2) For any $\Pi^0_1$ class $P_e = [T_e]$, if $A \in P_e$, then $A_n \in P_e$ for all $n \geq e$.

By property (1), $\mathbf{0}'$ is recursive in $A$, so that, since $A$ is $\Pi^0_1$, $A$ has degree $\mathbf{0}'$. It then follows from (2) as in the proof of Theorem 16.2.1 that $P$ is minimal and thin.

The sequence $a_0 < a_1 < \cdots$ is defined by $\Pi^0_1$ recursion in the style of Theorem 2.6.5 by making $a_n$ the least $a$ which satisfies the following:

(i) For all $m < n$, $a_m < a$.

(ii) $n \in B \to n \in B^a$.

(iii) For all $m < n$, either $< a_0, \dots, a_{n-1}, a > \notin T_m$ or $(\forall x)(< a_0, \dots, a_{n-1}, x > \in T_m)$.

(iv) For all $x < a$, either

(a) $x \leq x_{n-1}$ or

(b) $n \in B^a \setminus B^x$ or

(c) for some $m < n$, $< x_0, x_1, \dots, x_{n-1}, x > \in T_m \& < x_0, x_1, \dots, x_i, a > \notin T_m$.

The details are left to the reader. □

The following result is Theorem 2.13 of [22] (p. 102).

**Theorem 16.2.3.** *(C-D-J-S) Let $T$ be a recursive tree and $P$ a $\Pi^0_1$ class such that $P = [T]$. Then for any set $A \in P$,*

*(a) If $P \subset \mathcal{P}(A)$, then $A \leq_T Ext(T)$.*

*(b) If $A$ is a $\Pi^0_1$ set and $P$ is thin, then $A \leq_T Ext(T)$*

*(c) If $T$ has no dead ends and $A$ is either r. e. or co-r. e., then $A$ is recursive.*

*Proof.* (a) To test whether $n \in A$, simply see if there is a $\sigma \in Ext(T)$ of length $n + 1$ such that $\sigma(n) = 1$.

(b) Note that $\mathcal{P}(A)$ is a $\Pi^0_1$ class, so that $Q = P \cap \mathcal{P}(A)$ is a $\Pi^0_1$ subclass of $P$ and is nonempty since $A \in Q$. Since $P$ is thin, we must have $Q = P \cap U$ for some clopen $U = I(\sigma_0) \cup \cdots \cup I(\sigma_k)$. If we now define

$T_Q = \{\sigma \in T : \sigma \mathrm{iscompatiblewith} \sigma_i, \mathrm{forsome} i \leq k\}$,

then it is clear that $Q = [T_Q]$ and that

$Ext(T_q) = \{\sigma \in Ext(T) : \sigma \mathrm{iscompatiblewith} \sigma_i, \mathrm{forsome} i \leq k\}$,

so that $Ext(T_Q)$ is recursive in $Ext(T)$. Now $Q \subset \mathcal{P}(A)$, so that by (a) we have $A \leq_T Ext(T_Q) \leq_T Ext(T)$.

(c) It is immediate from (b) that if $A$ is $\Pi^0_1$ set, then $A$ is recursive. If $A$ is an r.e. set, then $\omega \setminus A$ is $\Pi^0_1$ and belongs to the thin $\Pi^0_1$ class $\{\omega \setminus X : X \in P\}$. □

**Theorem 16.2.4.** *(C-D-J-S) Let $T$ be a recursive tree such that $P = [T]$ is a thin $\Pi^0_1$ class and let $A \in P$. Then*

*(a) $A' \leq_T A \oplus \mathbf{0}''$ (so that it is not possible that $A \geq_T \mathbf{0}''$.)*

*(b) If $T$ has no dead ends, then $A' \leq_T A \oplus \mathbf{0}'$ (so that it is not possible that $A \geq_T \mathbf{0}'$.)*

*Proof.* (a) Let $P = [T]$ be thin and suppose $A \in P$. For each $e$, let $Q_e = \{C : \phi_e^C(e) \uparrow\}$. Then $Q_e$ is a $\Pi^0_1$ class, so there is a clopen set $U(e)$ such that $P \cap Q_e = P \cap U(e)$. Thus if $\phi_e^A(e) \uparrow$, then there is some $\sigma = A\lceil n$ such that $\sigma$ forces $\phi_e^A(e) \uparrow$, that is, such that, for any $B \in P$, if $\sigma \prec B$ then $\phi_e^B(e) \uparrow$. Now define the $\Pi^0_2$ relation $R(e, \sigma)$ which says that $\sigma$ forces $\phi_e^B(e) \uparrow$, by

$R(e,\sigma) \iff (\forall \tau \succ \sigma)[(\tau \in T \& \phi_e^\tau(e) \downarrow) \rightarrow \tau \notin Ext(T)]$.

Then we can compute from $A$ together with $\mathbf{0}''$, whether $e \in A'$ by searching for the least $n$ such that, for $\sigma = x\lceil n$, either $\phi_e^\sigma(e) \downarrow$, in which case $e \in A'$, or $R(e, \sigma)$, in which case $e \notin A'$.

(b) 0bserve that if $Ext(T)$ is recursive, then the relation $R$ defined above will be recursive in $\mathbf{0}'$. □

It follows from (b) and Theorem 4.2 (b) above that if $A$ has rank one in a thin $\Pi^0_1$ class $P = [T]$, where $T$ has no dead ends, then $A$ has *low* degree $\mathbf{a}$, that is, $\mathbf{a}' = \mathbf{0}'$.

Part (a) of this theorem is best possible in the sense that, as shown in Theorem 2.18 of [22], there is a minimal thin $\Pi^0_1$ class $P$ and a set $A$ such that $D(P) = \{A\}$ and $A \oplus \mathbf{0}' \equiv_T \mathbf{0}''$.

We conclude this section by stating without proof several further results from [22].

**Theorem 16.2.5.** *(C-D-J-S) Between any two distinct r. e. degrees $\mathbf{b} < \mathbf{c}$, there is a degree $\mathbf{a}$, a set $A$ of degree $\mathbf{a}$ and a minimal, thin $\Pi^0_1$ class $P$ with $D(P) = \{A\}$.*

There is a family of c. e. degrees which contain members of thin $\Pi^0_1$ classes. In particular, it follows from Theorem 4.9 of Downey-Jockusch-Stob [59] that all array non-computable ( *a.n.c.*) degrees and hence all non $-$ low$_2$ degrees contain members of thin $\Pi^0_1$ classes.

Theorem 5.8 tells us that no set of degree $\mathbf{0}''$ can even belong to a thin $\Pi^0_1$ class. Two further results give lower degrees which also contain no members of thin classes.

**Theorem 16.2.6.** *(C-D-J-S) (a) There is an r.e. degree $\mathbf{a}$ such that no set $B$ of degree $\mathbf{a}$ belongs to any thin $\Pi^0_1$ class.*

*(b) There is a minimal degree $\mathbf{a} < \mathbf{0}'$ such that no set $A$ of degree $\mathbf{a}$ is a member of any thin $\Pi^0_1$ class.*

In contrast, we have the following improvement of Theorem 4.4.14.

**Theorem 16.2.7.** *(C-D-J-S) There is a non-recursive set $A \leq_T \mathbf{0}''$ such that every non-recursive set $B \leq_T A$ is a rank 1 member of a minimal, thin $\Pi^0_1$ class.*

Finally, there is another connection with maximal c. e. sets.

**Theorem 16.2.8.** *(C-D-J-S) There is a maximal set $A$ which is not a member of any thin $\Pi^0_1$ class.*

## 16.3 Initial Segments of the Lattice

In this section, we show that in the lattice $\mathcal{E}_\Pi$ of $\Pi^0_1$ classes there are initial segments $[\emptyset, P] = \mathcal{L}(P)$ which are not Boolean algebras, but which have a decidable theory. In fact, we will construct for any finite distributive lattice $L$ which satisfies the dual of the usual reduction property a $\Pi^0_1$ class $P$ such that $L$ is isomorphic to the lattice $\mathcal{L}(P)^*$, which is $\mathcal{L}(P)$, modulo finite differences. For the 2-element lattice, we obtain a *minimal* class, first constructed by Cenzer, Downey, Jockusch and Shore in 1993. For the simplest new $\Pi^0_1$ class $P$ constructed, $P$ has a single, non-computable limit point and $\mathcal{L}(P)^*$ has three elements, corresponding to $\emptyset$, $P$ and a minimal class $P_0 \subset P$. The element corresponding to $P_0$ has no complement in the lattice. On the other hand, the theory of $\mathcal{L}(P)$ is shown to be decidable. We show that if $P$ is decidable and has only finitely many limit points, then $\mathcal{L}(P)^*$ is always a Boolean algebra. We show that if $P$ is a decidable $\Pi^0_1$ class and $\mathcal{L}(P)$ is not a Boolean algebra, then the theory of $\mathcal{L}(P)$ interprets the theory of arithmetic and is therefore undecidable.

It was proved in Nies [149] that the theory of each interval of the lattice $\mathcal{E}$ which is not a Boolean algebra interprets true arithmetic (and is therefore certainly undecidable). However, we will show that in $\mathcal{L}$ there are initial segments $[\emptyset, P] = \mathcal{L}(P)$ which are not Boolean algebras, but which have a decidable theory.

We will construct for any finite distributive lattice $L$ which satisfies the dual of the usual reduction property a $\Pi^0_1$ class $P$ such that $L$ is isomorphic to the lattice $\mathcal{L}(P)^*$, which is $\mathcal{L}(P)$, modulo finite differences. We will show that $\mathcal{L}(P)$ is isomorphic to a sublattice of $\mathcal{P}(\mathbb{N})$ which is closed under finite differences and then apply a theorem of Lachlan [111] to conclude that the theory of $\mathcal{L}(P)$ is many-one reducible to the theory of the finite lattice $L$ and is therefore decidable.

The construction of the $\Pi^0_1$ class corresponding to a given lattice builds on the construction of a *minimal* $\Pi^0_1$ class in [22]. The simplest minimal $\Pi^0_1$ class $P$ has a single limit point together with countably many isolated points. $P$ has the property that every $\Pi^0_1$ subclass $Q$ of $P$ is either finite or is cofinite in $P$ – furthermore, $Q$ is the intersection of $P$ with a clopen set. Thus the lattice $\mathcal{L}(P)$ of $\Pi^0_1$ subclasses of $P$ is isomorphic to the class of finite/cofinite subsets of $\omega$ and is a Boolean algebra. Such a class plays a role in the lattice $\mathcal{L}$ corresponding to the dual of the role played by a maximal c.e. set in the lattice $\mathcal{E}$.

For the simplest new $\Pi^0_1$ class $P$ constructed, $P$ includes a minimal subclass $P_0$, has a single, non-computable limit point and $P$ has three types of subclasses: (i) finite classes, (ii) cofinite classes, and (iii) classes which are cofinite in $P_0$ and finite in $P - P_0$. The third type of subclass has no complement in the lattice, which is why the lattice is not a Boolean algebra.

This lattice is isomorphic to the lattice $L$ of subsets of $\omega$ containing all finite and cofinite sets together with all sets $S$ containing cofinitely many even numbers and finitely many odd numbers. We observe that $L$ is isomorphic to the dual lattice of complementary sets. The theory of $L$ is seen to be decidable by Lachlan's result, as explained above. It is not hard to see that this lattice may not be realized as the class of c.e. subsets of any c.e. set. Indeed, let $A \subset B$ be c.e. sets and let $\mathcal{L}$ be the interval $[A, B]$ of c.e. sets $C$ such that $A \subset C \subset B$, modulo finite difference. If some set $C$ is not complemented in $\mathcal{L}$, then it follows from repeated applications of the Owings Splitting Theorem ([181], p. 183) that $\mathcal{L}$ is infinite.

The original construction of a minimal thin class in Theorem 2.2 of [22], p. 88, provides a decidable $\Pi^0_1$ class $P$ such that $\mathcal{L}(P)^*$ is the trivial Boolean algebra $\{0, 1\}$.

We will show that if $P$ is decidable and has only finitely many limit points, then $\mathcal{L}(P)^*$ is always a Boolean algebra. Thus if $P$ is a decidable $\Pi^0_1$ class and $\mathcal{L}(P)^*$ is not a Boolean algebra, then $P$ has infinitely many limit points.

Finally, we will show that if $P$ is a decidable $\Pi^0_1$ class and $\mathcal{L}(P)$ is not a Boolean algebra, then the theory of $\mathcal{L}(P)$ interprets the theory of arithmetic and is therefore undecidable.

As usual, we say that sets $A$ and $B$ are equal modulo finite difference (written $A =^* B$) if the symmetric difference $(A - B) \cup (B - A)$ is finite. For a lattice $L$ of sets, let $L^*$ be the quotient lattice of $L$ modulo the equivalence relation $=^*$. We note here that if $A$ and $B$ are $\Pi^0_1$ classes and $A - B$ is finite, then any element of $A - B$ is computable, so that $A - B$ is also a $\Pi^0_1$ class. However, the lattice $\mathcal{E}_P$ is *not* closed under finite differences, since if $x$ is a computable element of the $\Pi^0_1$ class $P$ and is a limit point of $P$, then $\{x\}$ is also a $\Pi^0_1$ class, but $P - \{x\}$ is not even a closed set and thus is not a $\Pi^0_1$ class.

### 16.3.1 Representation of finite lattices

For any $\Pi^0_1$ class $P$, the family $\mathcal{L}(P)$ of $\Pi^0_1$ subclasses of $P$ is an initial segment of the lattice of $\Pi^0_1$ classes. It is clear that each such initial segment is a sublattice of the full lattice of $\Pi^0_1$ classes with least member $\emptyset = 0$ and greatest element $P = 1$, and is distributive. The quotient lattice $\mathcal{L}(P)$ is likewise a distributive lattice. In this section, we characterize the family of finite lattices $L$ which are isomorphic to $\mathcal{L}(P)^*$ for some $\Pi^0_1$ class $P$ and also the family of finite lattices $L$ which are isomorphic to $\mathcal{L}(P)^*$ for some decidable $\Pi^0_1$ class $P$.

We will show that $\mathcal{L}(P)$ satisfies the following *Dual Reduction Property.*

**Definition 16.3.1.** *The lattice $(L, \leq)$ satisfies the dual reduction property if for any $a, b \in L$, there exist $a_1 \geq a$ and $b_1 \geq b$ such that $a_1 \vee b_1 = 1$ and $a_1 \wedge b_1 = a \wedge b$.*

Let $\mathcal{L}(P)^*$ denote the lattice $[\emptyset, P]$ modulo finite difference. This lattice will also be distributive and satisfy the dual reduction property.

**Proposition 16.3.2.** *For any $\Pi^0_1$ class $P$, the lattices $\mathcal{L}(P)$ and $\mathcal{L}^*(P)$ satisfy the dual reduction property.*

*Proof.* Let $P_1$ and $P_2$ be (nonempty) $\Pi^0_1$ subclasses of $P$ and, for $i = 0, 1$, let $T_i$ be a computable tree such that $P_i = [T_i]$ is the set of infinite paths through $T_i$. We define computable trees $S_i \supset T_i$ such that $S_1 \cap S_2 = T_1 \cap T_2$ and $S_1 \cup S_2 = \{0,1\}^{<\omega}$ and let $Q_i = [S_i]$. It will follow that $Q_1 \cap Q_2 = P_1 \cap P_2$ and that $Q_1 \cup Q_2 = \{0,1\}^\omega$; the desired classes are $Q_1 \cap P$ and $Q_2 \cap P$. For the first condition, suppose that $x \in Q_1 \cap Q_2$. Then $x\lceil n \in S_1 \cap S_2$ for each $n$, so that $x\lceil n \in T_1 \cap T_2$ for each $n$, and therefore $x \in P_1 \cap P_2$. For any $x$, we have that for each $n$, either $x\lceil n \in S_1$ or $x\lceil n \in S_2$. Thus without loss of generality $x\lceil n \in S_1$ for infinitely many $n$. Since $S_1$ is a tree, $x\lceil n \in S_1 \to x\lceil m \in S_1$ for $m < n$, so that $x\lceil n \in S_1$ for all $n$ and therefore $x \in Q_1$.

The definition of the trees $S_i$ is by recursion on the length of $\sigma \in \{0,1\}^{<\omega}$. First put the empty string in both $S_1$ and $S_2$ since it is in $T_1 \cap T_2$. Now assume by induction that for strings $\sigma$ of length $\leq n$, we have

(i) $\sigma \in S_1 \cup S_2$ and

(ii) $\sigma \in S_1 \cap S_2 \iff \sigma \in T_1 \cap T_2$.

Now for $\tau = \sigma^\frown 0$ or $\sigma^\frown 1$, there are 4 cases; the final case is most important.

(a) If $\tau \in T_1 \cap T_2$, then we put $\tau \in S_1 \cap S_2$.

(b) If $\tau \in T_1 - T_2$, then we put $\tau \in S_1 - S_2$.

(c) If $\tau \in T_2 - T_1$, then we put $\tau \in S_2 - S_1$.

(d) If $\tau \notin T_1 \cup T_2$, then we consider whether $\sigma \in S_1$ or $S_2$. If $\sigma \in S_2 - S_1$, then we put $\tau \in S_2 - S_1$ and otherwise, we put $\tau \in S_1 - S_2$.

It is easy to check that in each case, if $\tau \in S_i$, then $\sigma \in S_i$, so that each $S_i$ is a tree. The conditions (i) and (ii) follow from the construction by induction on the length of $\sigma$. □

We now prove a converse result.

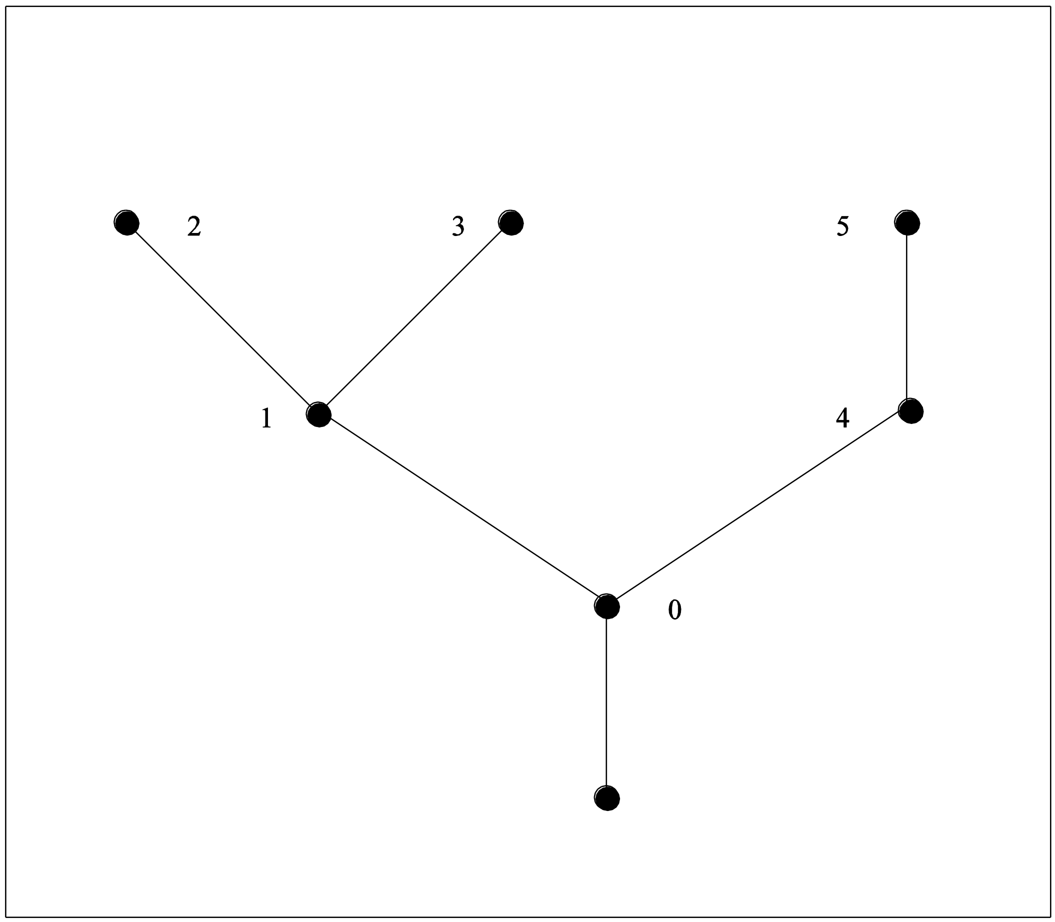
**Theorem 16.3.3.** *For any finite distributive lattice $L$ which satisfies the dual reduction property, there exists a $\Pi^0_1$ class $Q$ such that $\mathcal{L}(Q)^*$ is isomorphic to $L$. Furthermore, the theory of $\mathcal{L}(Q)$ is decidable.*

*Proof.* Notice that for any finite class $Q$, $\mathcal{L}(Q)^*$ will be the one-point lattice. For the simplest non-trivial example, the two-point lattice $L = \{0,1\}$, $Q$ must be a *minimal* $\Pi^0_1$ class, meaning that every $\Pi^0_1$ subclass is either finite or is cofinite in $Q$. Such a class was constructed in [22]. The construction given below is based on the construction of a minimal $\Pi^0_1$ class.

Let $T_e$ be a standard enumeration of the primitive recursive trees, so that $P_e = [T_e]$ enumerates the $\Pi^0_1$ classes as in [36].

We need the following characterization of the finite distributive lattices satisfying the dual reduction property, which follows from Hermann [79].

**Lemma 16.3.4.** *Suppose $L$ is a finite lattice of sets. Then $L$ satisfies the dual reduction property if and only if there exists a tree $S$ with root $\emptyset$ which generates $L$ in the sense that every element of $L$ is a join of a set of nodes.*



2
3
5
1
4
0

Note that $S$ is uniquely determined from $L$ as the set of join-irreducible elements of $L$.

To illustrate this idea, let $S$ consist of the following subsets of $\{0,1,2,3\}$: $\emptyset, \{0\}, \{0,1\}, \{0,1,2\}, \{0,1,3\}, \{0,4\}, \{0,4,5\}$. Here a set $B$ is a successor of a set $A$ if $B = A \cup \{b\}$ for some $b$. $S$ is a lower semi-lattice under the operation of intersection and generates a lattice with the operation of union as follows. The leaves of $S$ are the sets $\{0,1,2\}$, $\{0,1,3\}$ and $\{0,4,5\}$. Thus $S$ generates a lattice $L(S)$ with the addition of 8 sets: $\{0,1,4\}$, $\{0,1,4,5\}$, $\{0,1,2,4\}$, $\{0,1,3,4\}$, $\{0,1,2,3\}$, $\{0,1,2,4,5\}$, $\{0,1,3,4,5\}$ and $\{0,1,2,3,4,5\}$ – the maximum element of $L(S)$. A sketch of the tree $S$ is given above in Figure 16.3.1

Suppose now that the lattice $L$ is generated by a tree $S$ of finite sets with $B$ a successor of $A$ in $S$ if and only if $B = A \cup \{b\}$ for some $b$ as in the above example, so that the new elements are ordered as usual from left to right in the tree.

Each $b \leq m$ may be identified with the unique $B(b) \in S$ such that $B(b) = A \cup \{b\}$ for some $A \in S$. Then we define a partial ordering on $\{0,1,\dots,m\}$ by

$$a \leq_* b \iff B(a) \subset B(b).$$

We may assume that $a \leq_* b$ implies $a \leq b$ (by renumbering if necessary). We may also simplify the problem by assuming, without loss of generality, that there is only one atom $\{0\}$ in $L$. If there are several atoms $\{i\}$ for $i = 1$ to $k$, then we can use the construction for one atom to produce disjoint $\Pi^0_1$ classes $Q_1, \dots, Q_k$ such that $\mathcal{L}(Q_i)^*$ is isomorphic to the lattice $L_i = \{\emptyset\} \cup \{A \in L : i \in A\}$. It is then easy to see that for $Q = \cup_i Q_i$, $\mathcal{L}(Q)^*$ is isomorphic to $L$.

Suppose therefore that the generating tree $S$ has a single atom $\{0\}$ and is a family of subsets of $\{0,1,\dots,m\}$. We will construct the class $Q$ with corresponding subclasses $Q_A$ for each $A \in L$ such that every subclass of $Q$ differs from one of the $Q_A$ by a finite set. The classes are constructed so that $A \subset B \iff Q_A \subset Q_B$. It is immediate that $Q_{\{0\}}$ is a minimal $\Pi^0_1$ class.

Our goal is to define a $\Pi^0_1$ class $Q$ with natural subclasses $Q_A$ for each $A \in L$ so that for each $\Pi^0_1$ class $P_e \subset Q$, there is some $A$ such that the difference between $P_e$ and $Q_A$ is finite.

The class $Q$ will have a single limit element $x$, which will also be the only element of $Q$ containing infinitely many "1"s. If we express $x$ in the form $0^{n_0} * 1 * 0^{n_1} * 1 * \dots$, let $\sigma_0 = 0$ and let $\sigma_k = 0^{n_0} * 1 * \cdots * 0^{n_k}$, then the class $Q_{\{0\}}$ will have additional elements $x_{0,k} = \sigma_k * 0 * 1 * 0^\omega$ for each $k$.

For each $i \leq m$ with $i > 0$, we will have a corresponding *label* $1^{i+1}$ such that for $A \in L$ and $i \in A$, the elements of $Q_A$ will all contain $0 * 1^{i+1} * 0$ as a substring. In fact, we will characterize $Q_A$ as those elements of $Q$ which have no labels of the form $0 * 1^{m+1} * 0$ for any $m \notin A$. Note that this will make $Q_A$ a $\Pi^0_1$ subclass of $Q$. For each $B = A \cup \{i\} \in S$, we will define a sequence of elements $x_{i,k}$ which have labels for all $i \in B$ and no other labels. This will be done so that for each $i$, $x_{i,k}$ is an extension of $\sigma_k$ but not an extension of $\sigma_{k+1}$.

A sketch of the class $Q$ for the simple case of $S = \{\emptyset, \{0\}, \{0,1\}\}$ is given below in Figure Two.

It follows from the above discussion that the map taking $A \in L$ to $Q_A$ is a lattice homomorphism, that is, $A \subset B \iff Q_A \subset Q_B$.

The key to making the subclasses of $Q$, modulo finite difference, isomorphic to $\mathcal{L}$, is the following condition:

($*$): For any $b \leq m$, any $e$ and any $A$ and $B$ in $S$ with $B = A \cup \{b\}$, if $P_e \cap (Q_B - Q_A)$ is infinite, then $Q_B - P_e$ is finite.

Given this condition, we now show that for every $\Pi^0_1$ subclass $P_e$ of $Q$, there exists $C \in L$ such that $P_e = Q_C$ modulo finite difference. Just let $C = \bigcup\{A : Q_A - P_e \text{ is finite}\}$. Clearly $Q_C - P_e$ is finite. Now suppose by way of contradiction that $P_e - Q_C$ is infinite. Then there must be some $B \in S$ with $P_e \cap (Q_B - Q_C)$ infinite. Let $B$ have minimal cardinality among the set of $D$ such that $P_e \cap (Q_D - Q_C)$ is infinite and let $A$ be the predecessor of $B$. Then there is a $b$ such that $B = A \cup \{b\}$ and $P_e \cap (Q_B - Q_A)$ is infinite. It now follows from (*) that $Q_B - P_e$ is finite. But the definition of $C$ now requires that $Q_B \subset Q_C$, contradicting the assumption that $P_e \cap (Q_B - Q_C)$ is infinite. Thus $P_e$ and $Q_C$ have a finite difference, as desired.

Now let us see how to obtain this condition in the construction. Recall that we are defining $x$ as the limit of strings $\sigma_k$ and also defining $x_{b,k}$ for each $k$ and for $b \leq m$ as the limit of, say, $\mu_{b,k}$.

The requirements used in the construction to obtain condition ($*$) are the following, for each $b \leq m$ and each pair of natural numbers $e \leq j$.

**Requirement $\mathbf{R_{b,j,e}}$:**

(i) if $b = 0$ and $x \in P_e$, then $x_{0,j} \in P_e$;

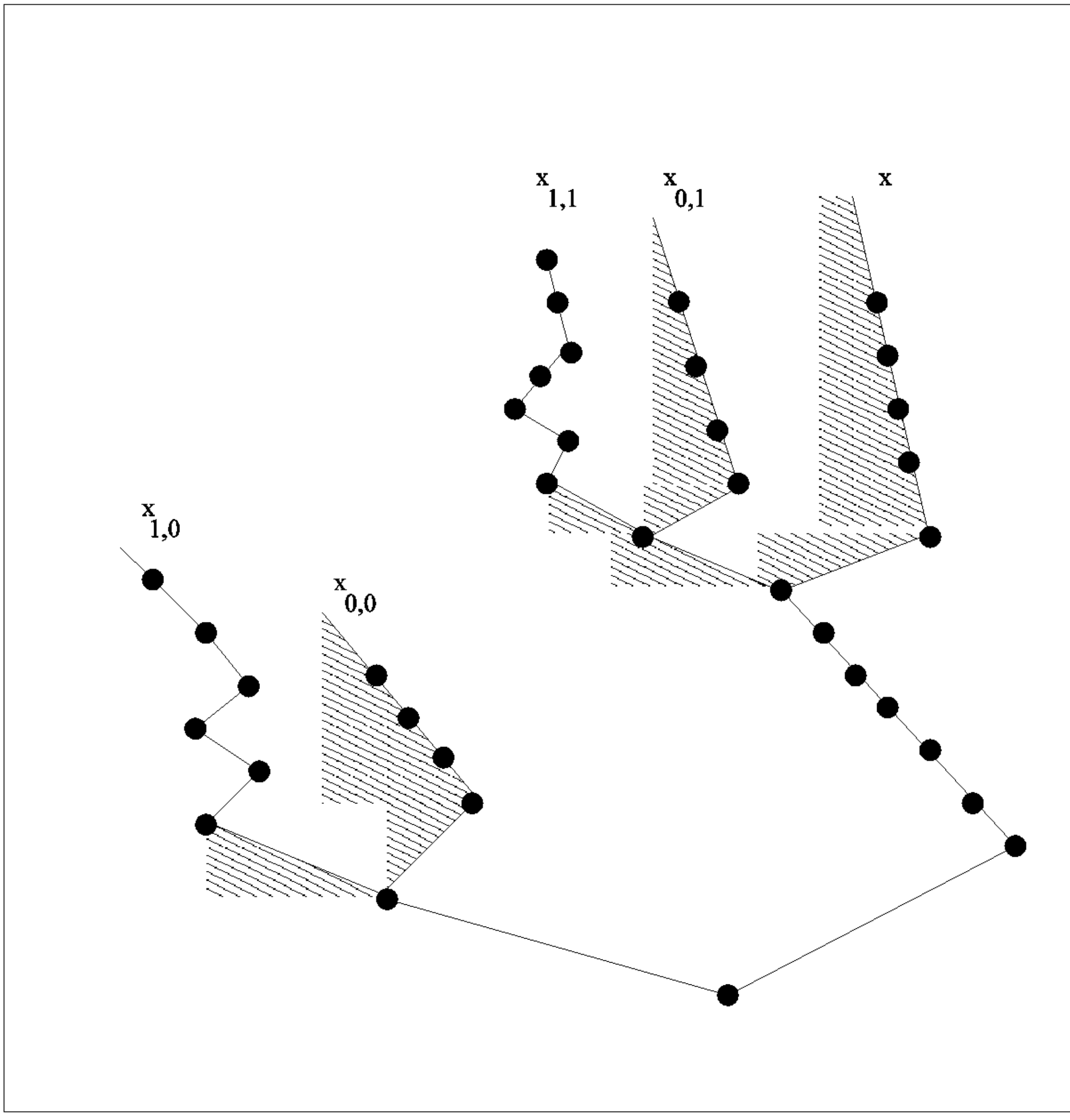
(ii) if $b > 0$, $a \leq_* b$, and $x_{b,j} \in P_e$, then $x_{a,k} \in P_e$ for all $k \geq j$.

(Recall that $P_e$ is the $e$-th $\Pi^0_1$ class.) Let us demonstrate that these requirements imply the condition $(*)$ given above.

Suppose therefore that $B = A \cup \{b\}$ and that $P_e \cap (Q_B - Q_A)$ is infinite and let $a \in B$. This means that $x_{b,j} \in P_e$ for infinitely many $j$ and thus for some $j \geq e$. Then the requirement $R_{b,j,e}$ implies that $x_{a,k} \in P_e$ for all but finitely many $k$. Thus $Q_B - P_e$ is finite as desired.

We will show below that these requirements also imply that $x$ is the unique limit point of $Q$ and that $x$ is not computable.

Priority is assigned to the requirements as follows. $R_{a,i,d}$ has higher priority than $R_{b,j,e}$ if either $i < j$, or $i = j$ and $d < e$, or $i = j$ and $d = e$ and $a < b$.


x
1,1
x
0,1
x
x
1,0
x
0,0

It remains to construct the set $Q$ by a finite injury argument. The construction will proceed in stages. At stage $s$ we will have, for $e \leq s$, strings $\sigma_e^s \prec \mu_{0,e}^s$, containing at least $e$ 1's, such that, for all $e < s$, $\sigma_e^s{}^\frown 1 \prec \sigma_{e+1}^s$, together with strings $\mu_{b,k}^s$ for $1 \leq b \leq m$ and $k < s$ such that $\mu_{b,k}$ extends $\sigma_k^s$ but does not extend $\sigma_{k+1}^s$. The construction will ensure the existence of the limits $\sigma_e = \lim_s \sigma_e^s$ for each $e$. The unique limit point $x$ of $Q$ will the union of $\{\sigma_e : e \in \omega\}$. For each $b$ and $k$, the element $x_{b,k}$ of $Q$ will be the limit of the strings $\mu_{b,k}^s$ in the sense that $x_{b,k}(i) = lim_s \mu_{b,k}^s(i)$ for each $i$. At the same time we will be defining a sequence $n(0) < n(1) < \dots$ so that $s \leq n(s)$ and constructing a computable tree $T$ in stages $T^s$. At stage $s$, we will have decided whether each finite sequence of length $n(s)$ is in $T$. This will ensure that $T$ is computable.

We first give an outline of the construction for the case when $L$ is a chain with 3 nodes 0,$\{0\}$ and $\{0,1\}$. We will build a computable tree $T$ with $Q = [T] = \{x\} \cup \{x_{0,j} : j < \omega\} \cup \{x_{1,j} : j < \omega\}$, where $x$ is the unique limit path of $Q$, also called the *main path*. Thus $Q_0 = \emptyset$, $Q_{\{0\}} = \{x\} \cup \{x_{0,j} : j < \omega\}$ and of course $Q_{\{0,1\}} = Q$. The main path will have the form $0^{n_0} * 1 * 0^{n_1} * \dots$, while the isolated paths will each end in $0^\omega$. The paths $x_{1,j}$ will each have as a *label* the substring (11), while the other paths will not have this label. The isolated paths $x_{a,j}$ will agree with $x$ at least as far as $0^{n_0} * 1 * \dots * 1 * 0^{n_j}$.

We achieve the requirements $R_{b,j,e}$ by working on the converses. That is, if it looks like $x_{0,j} \notin P_e$ but $x \in P_e$, then we move $x$ to $x_{0,j}$ (by making $\mu_{0,j}^s \prec \sigma_e^{s+1}$) to ensure that $x \notin P_e$. Similarly, if for some $j < k$ it looks like $x_{0,j} \in P_e$ but $x_{0,k} \notin P_e$, then we move $x_{0,j}$ to $x_{0,k}$ to ensure that $x_{0,j} \notin P_e$. The other cases move $x_{1,j}$ to $x_{1,k}$ or move $x_{1,k}$ to $x_{1,j}$ for $j \leq k$. The restriction that $e \leq j$ will ensure that the construction converges.

To see that these requirements lead to the desired conclusion, we suppose now that some $P_e \subset Q$ and show that $Q$ is equal (modulo finite difference) to one of the three sets $Q_A$ defined above. If $P_e$ is finite, then clearly $P_e = Q_0$ (modulo finite). If $P_e$ is infinite, then it has a limit point, so that $x \in P_e$ and therefore, by part (i) of the Requirement, $x_{0,j} \in P_e$ for all $j \geq e$, so that $Q_{\{0\}} \subset P_e$ (modulo finite). In this case, if $P_e$ contains just finitely many $x_{1,j}$, then $P_e = Q_{\{0\}}$ (modulo finite). If $P_e$ contains infinitely many $x_{1,j}$, then, by part (ii), it must contain all points $x_{1,j}$ and $x_{0,j}$ for $j \geq e$, so that $P_e = Q$ (modulo finite).

We begin the construction by setting $n(0) = 0$ and letting $\sigma_0^0$ be the null string.

Now suppose we have completed the construction as far as stage $s$. Thus we have defined $n(s) > s$ and decided whether $\sigma \in T$ for all strings $\sigma$ of length $\leq n(s)$. We have also defined $\sigma_e^s$ for all $e \leq s$ and also $\mu_{a,k}^s$ for all $a \leq m$ and $k < s$ as described above.

At stage $s+1$, the triple $(b, j, e)$ with $j \geq e$ and $b > 0$ requires action if we have $\mu_{b,j}^s \in T_e$ and we have some $a \leq_* b$ and some $k > j$ such that $\mu_{a,k}^s \notin T_e$.

The triple $(0, j, e)$ with $j \geq e$ requires action if $\sigma_j^s \in T_e$ and there is some $k > j$ such that $\mu_{0,k}^s \notin T_e$.

If no triple requires action at stage $s+1$, then we simply extend the tree as follows.

For each $a \leq m$, let $0 = a_0 <_* a_1 <_* \cdots <_* a_n = a$ list the nodes below or equal to $a$ in $S$, let

$$\ell(a) = a + 1 + a_0 + 2 + \cdots + a_n + 2$$

and let $\ell = max\{\ell(a) : a \leq m\}$ and $n(s+1) = n(s) + \ell$.

For all $a \leq m$ and all $k < s$, let $\mu^{s+1}_{a,k} = \mu^s_{a,k} * 0^\ell$. Let $\sigma^{s+1}_k = \sigma^s_k$ and let $\sigma^{s+1}_{s+1} = \sigma^s_s * 1 * 0^{\ell-1}$. Finally, let

$$\mu^{s+1}_{a,s} = \sigma^s_s * 0^{a+1} * 1^{a_0+1} * 0 * 1^{a_1+1} * 0 * \cdots * 0 * 1^{a_n+1} * 0^{\ell-\ell(a)}.$$

Otherwise, let $(b, j, e)$ be the triple with highest priority which requires action at stage $s+1$ and do the following.

**Case I**: $b = 0$. Then we have $\sigma^s_j \in T_e$ and $k > j$ such that $\mu^s_{0,k} \notin T_e$. Now the idea is to move $\sigma_j$ to $\mu^s_{0,k}$, to abandon the part of the tree which branches off between $\sigma^s_j$ and $\mu^s_{0,k}$ and restart the construction above the new $\sigma^{s+1}_j$. The details follow.

Define $\ell$ as above and let $n(s+1) = n(s) + (s+1-j)(\ell)$. For $i \leq s+1-j$, let

$$\sigma^{s+1}_{j+i} = \mu^s_{0,k} * (1 * 0^{\ell-1})^i.$$

For $a \leq m$ as above and for $i \leq s-j$, let

$$\mu^{s+1}_{a,j+i} = \sigma^{s+1}_{j+i} * 0^{a+1} * 1^{a_0+1} * 0 * 1^{a_1+1} * 0 * \cdots * 0 * 1^{a_n+1} * 0^{(s+1-j-i)\ell-\ell(a)}.$$

For $i < j$, let $\sigma^{s+1}_i = \sigma^s_i$ and for each $a$, let $\mu^{s+1}_{a,i} = \mu^s_{a,i} * 0^{(s+1-j)\ell}$.

**Case II**: $b > 0$. Then we have $\mu^s_{b,j} \in T_e$ and we have some $c \leq_* b$ and $k > j$ such that $\mu^s_{c,k} \notin T_e$. Now the idea is to move $\mu_{b,j}$ to $\mu^s_{c,k}$, move $\sigma_{j+1}$ to $\sigma_{k+1}$ and to abandon the part of the tree which branches off between $\sigma^s_j$ and $\sigma^s_k$, except for the $\mu_{a,j}$ with $a \neq b$. The tree above $\sigma^s_{k+1}$ is relabeled and the construction is restarted above $\sigma^s_s$. The details follow.

Define $\ell$ as above and let $n(s+1) = n(s) + (k-j)\ell$. Let $c = c_0 <_* c_1 <_* \cdots <_* c_r = b$ list the nodes of $T$ between $c$ and $b$ and let

$$\mu^{s+1}_{b,j} = \mu^s_{c,k} * 1^{c_1+1} * 0 * \cdots * 0 * 1^{c_r+1} * 0^q,$$

where $q$ is chosen so that $|\mu^{s+1}_{b,j}$ has length $n(s+1)$. For $a \neq b$, let

$$\mu^{s+1}_{a,j} = \mu^s_{a,j} * 0^{(k-j)\ell}.$$

Let $\sigma^{s+1}_j = \sigma^s_j$. For $0 < i \leq s-k$, let

$$\sigma^{s+1}_{j+i} = \sigma^s_{k+i}$$

and for $i$ with $0 < i < k-j$ and for any $a \leq m$, let

$$\mu^{s+1}_{a,j+i} = \mu^s_{a,k+i} * 0^{(k-j)\ell}.$$

For $i \leq k - j + 1$, let

$$\sigma^{s+1}_{s+j-k+i} = \sigma^s_s * (1 * 0^{\ell-1})^i$$

and for $a \leq m$ and $0 < i < k - j + 1$, let

$$\mu^{s+1}_{a,s+j-k+i} = \sigma^{s+1}_{s+j-k+i} * 0^{a+1} * 1^{a_0+1} * 0 * 1^{a_1+1} * 0 * \cdots * 0 * 1^{a_n+1} * 0^{(k-j-i)\ell-\ell(a)}.$$

Finally, for $i < j$, let $\sigma^{s+1}_i = \sigma^s_i$ and for each $a$, let $\mu^{s+1}_{a,i} = \mu^s_{a,i} * 0^{(k-j)\ell}$.

In each case, a string $\sigma$ of length $\leq n(s+1)$ is in $T$ if and only if either $\sigma \prec \sigma^{s+1}_k$ for some $k$, or $\sigma \prec \mu^{s+1}_{a,k} * 0^t$ for some $a, k, t$.

**Claim 16.3.5.** *For every $k$, the sequence $\sigma^s_k$ converges to some limit $\sigma_k$ and for every $a \leq m$ and every $e$, there is a stage $s$ such that, for all $t \geq s$, $\mu^{t+1}_{a,k}$ is an extension of $\mu^t_{a,k}$ by a string of 0's.*

*Proof.* It follows from the construction that we only have $\sigma^{s+1}_k \neq \sigma^s_k$ when we take action on a requirement $R_{b,j,e}$ with $j \leq k$ and similarly we only move $\mu^s_{a,k}$ when we take action on $R_{b,j,e}$ with $j \leq k$.

Thus it suffices to show that for each $k$, there is a stage after which we never again take action on any requirement $R_{b,j,e}$ with $j \leq k$. We proceed by induction on $k$. For $k = 0$, the only possible requirements have the form $R_{b,0,0}$. For $b = 0$, one action at stage $s$ will put $\sigma^s_0 \notin T_0$ and no later action can injure this requirement. Now suppose that we have reached a stage $s_0$ such that we never act on requirement $R_{0,0,0}$ after stage $s_0$. Then for each $b, c \leq m$ with $b, c \neq 0$, observe that action taken on requirement $R_{b,0,0}$ does not move $\mu_{c,0}$, so that one action taken on requirement $R_{c,0,0}$ will move $\mu_{c,0}$ out of $T_0$ and no further action will be required.

Now suppose that we have reached a stage $s_{k-1}$ such that no action is ever taken on any requirement $R_{b,j,e}$ with $j < k$ after stage $s_{k-1}$. Then we see as in the $k = 0$ case above that there will be a stage after which we never act on requirement $R_{0,k,0}$ and then a stage $s_{k,0}$ after which we never act on requirement $R_{b,k,0}$ for any $b$. Since we always have $e \leq k$ in requirement $R_{b,k,e}$, we can show by induction on $e \leq k$ that there are stages $s_{k,e}$ after which we never act on requirement $R_{b,k,e}$ for any $b$. Thus after stage $s_k = s_{k,k}$, we never act on any requirement $R_{b,j,e}$ with $j \leq k$. $\square$

Since $\sigma^s_e \prec \sigma^s_{e+1}$ for all $s$ and $e$, it follows that $\sigma_e \prec \sigma_{e+1}$ for all $e$. Thus we can define the limit point $x$ of $Q$ to be $x = \cup_e \sigma_e$.

For each $a \leq m$ and each $k$, the sequence $\mu^s_{a,k}$ likewise converges to a path $\mu_{a,k} \in Q$. Since all other paths are eventually terminated, $Q$ consists of precisely the elements $x_{a,k}$ and the elements $x_e$. For each $k \geq e$, $x_{0,k}$ is an extension of $\sigma_e$, so that $x$ is a limit point of $Q$. It is clear that each $x_{a,k}$ is isolated in $Q$

since $\mu_{a,k}$ is eventually only extended by 0's in $T$. Thus $x$ is the unique limit point of $Q$.

It remains to verify the requirements $R_{b,j,e}$ given above. Note that for each $A \in L$, $Q_A = \{x\} \cup \{x_{a,k} : a \in A \;\&\; k < \omega\}$.

**Claim 16.3.6.** *Let $e \leq j$.*

*(i) If $b = 0$ and if $x \in P_e$, then $x_{0,j} \in P_e$.*

*(ii) If $b > 0$, and $B = A \cup \{b\}$ for some $A, B \in S$ and $x_{b,j} \in P_e$, then $x_{a,k} \in P_e$ for all $a \in B$ and all $k \geq j$.*

*Proof.* For the first part, suppose that $x \in P_e$ and let $s$ be a stage such that no action is taken on any requirement of priority less than or equal to $R_{0,e,j}$ after stage $s$. Then the condition must never require action at any stage $t+1 > s$. It follows that $\sigma_j = \sigma_j^s$. Since $x \in P_e$, it follows that $\sigma_j \in T_e$, so that for all $t \geq s$, $\mu_{0,k}^t \in T_e$ for all $k \geq j$. It follows that $x_{0,j} \in P_e$.

For the second part, assume the hypothesis of part (ii) and let $s$ be a stage such that no action is taken on any requirement of priority less than or equal to $R_{b,e,j}$ after stage $s$. Then the condition must never require action at any stage $t+1 > s$. It follows that $\mu_{a,j} = \mu_{a,j}^s$ for all $a \in B$. Since $x_{b,j} \in P_e$, it follows that $\mu_{b,j} \in T_e$, so that for all $a \in B$ and all $t \geq s$, $\mu_{a,k}^t \in T_e$ for all $k \geq j$. It follows that $x_{a,k} \in P_e$, as desired. □

It is important to note that these requirements, now verified, imply that the limit point $x$ is not computable. If it were, then $\{x\}$ would be a $\Pi_1^0$ class, say $P_e$. But then we would have $x_{0,j} \in P_e$ for all $j \geq e$, which is a contradiction.

Finally, we consider the furthermore clause of the theorem, that is, that the theory of $\mathcal{L}(Q)$ is decidable. By a theorem of Lachlan [111], if a lattice $L \subset \mathcal{P}(\mathbb{N})$ is closed under finite differences, then the theory of $L$ is many-one reducible to the theory of $L^*$.

**Lemma 16.3.7.** *Suppose that $P$ is a countable $\Pi_1^0$ class such that every computable member of $P$ is isolated. Then the lattice $\mathcal{L}(P)$ of $\Pi_1^0$ subclasses of $P$ is isomorphic to a sublattice $L$ of $\mathcal{P}(\mathbb{N})$ which is closed under finite differences.*

*Proof.* Let $A = \{\alpha_n : n < \omega\}$ be a list of the isolated points in $P$. It is sufficient to show that a $\Pi_1^0$ subclass of $P$ is determined by its intersection with $A$. To see this, suppose that $Q_1$ and $Q_2$ are two $\Pi_1^0$ classes having the same intersection with $A$. We first show by induction on the rank of $x \in P$ that if $x \in Q_i$ (where $i = 0, 1$), then for any open neighborhood $U$ of $x$, there is an element of $A$ which belongs to $Q_i \cap U$. The hypothesis covers the case of rank zero. Now suppose that $x \in Q_i$ and that $x$ has rank $\alpha$ in $P$. Let $U$ be an open set such that $x \in U$ and such that $U$ contains no points of rank $\geq \alpha$ in $P$ other than $x$. Since $x$ is not computable, $Q_i \cap U$ must contain some point $y \neq x$ and necessarily $y$ has rank $< \alpha$. It follows by induction that $Q_i \cap U$ contains an element of $A$. Now if $x \in Q_1$, then every neighborhood of $x$ contains an element of $A \cap Q_1$ and therefore, by assumption, an element of $A \cap Q_2$. Since $Q_2$ is closed, it follows that $x \in Q_2$. Similarly, $x \in Q_2 \to x \in Q_1$. □

The $\Pi^0_1$ class constructed above is certainly countable and the finitely many limit points are each non-computable. Thus by Lemma 16.3.7 the theory of $\mathcal{L}(Q)$ is many-one reducible to the theory of $\mathcal{L}(Q)^*$. But the latter is the theory of a finite structure and is therefore decidable.

This completes the proof of Theorem 16.3.3. $\square$

For the remainder of the section, we will show that the construction of Theorem 16.3.3 may not, in general, be achieved with a decidable class $P$. In Section 3, we will present a stronger result, that the theory of $\mathcal{L}(P)$ is undecidable, and in fact interprets the theory of arithmetic, whenever $P$ is decidable and $\mathcal{L}(P)$ is not a Boolean algebra. We include the next result since it gives a more direct proof that no decidable $\Pi^0_1$ class $P$ can have $\mathcal{L}(P)^*$ isomorphic to a finite lattice, such as the three-point lattice $\{0, 1, 2\}$, which is not a Boolean algebra.

**Theorem 16.3.8.** *If $P$ is a countably infinite, decidable $\Pi^0_1$ class, and $\mathcal{L}(P)^*$ is not a Boolean algebra, then $\mathcal{L}(P)^*$ is infinite.*

*Proof.* First suppose that $P$ has infinitely many limit points. This condition alone implies that $\mathcal{L}(P)^*$ is infinite; by the countability of $P$, there must be infinitely many $\{x_0, x_1, \dots\}$ which have rank one. This means that for each $n$, there is an interval $U_n$ such that $P \cap U_n$ contains $x_n$ and contains no other limit point of $P$. We claim that the sets $P \cap U_n$ are distinct modulo finite difference. Suppose by way of contradiction that $P \cap U_m$ and $P \cap U_n$ had a finite difference. Since $x_m$ is a limit point of $P$, there is a sequence $y_0, y_1, \dots$ of (isolated) elements of $P \cap U_m$ which converges to $x_m$. Then all but finitely many of these $y_k$ would belong to $P \cap U_n$ and therefore $x_m$ would be in $P \cap U_n$, contradicting the assumptions above.

Now suppose that $P$ has only finitely many limit points $\{x_0, \dots, x_k\}$. As above, we can separate them by intervals $U_n$ so that $P \cap U_n$ contains $x_n$ and no other limit point. Since $P - (U_0 \cup \dots \cup U_n)$ contains no limit points and is therefore finite, we may assume that the sets $P \cap U_n$ partition $P$. The assumption that $\mathcal{L}(P)^*$ is not a Boolean algebra thus implies that $\mathcal{L}(P \cap U_n)^*$ is not a Boolean algebra for some $n$. Thus we may assume without loss of generality that $P$ has a unique limit point.

Since $\mathcal{L}(P)^*$ is not a Boolean algebra, there must be some infinite subset $P_0$ of $P$ such that $P - P_0$ is also infinite. Assuming that $\mathcal{L}(P)^*$ is finite, we may take $P_0$ to be minimal and $P$ to be a minimal extension of $P_0$. That is, we may assume, without loss of generality, that $\mathcal{L}(P)^*$ has exactly 3 nodes, corresponding to $\emptyset$, $P_0$ and $P$. Now let $P = [T]$ where $T$ has no dead ends, let $P_0 = [T_0]$, and let $x$ be the unique limit point of $P$. Of course $x \in P_0$ since $P_0$ is infinite and $x$ is the only limit point of $P$. Observe that for any $\sigma \in T - T_0$, $\sigma$ has only finitely many extensions in $P$, since otherwise $P - P_0$ would contain a limit point of $P$.

Then we can recursively define a sequence $\sigma_0, \sigma_1, \dots$ of pairwise incompatible nodes in $T - T_0$, as follows. Let $\sigma_0$ be the least element of $T - T_0$. Given $\sigma_0, \dots, \sigma_n \in T - T_0$, there exists an element $y \in P - P_0$ which does not extend any of $\sigma_0, \dots, \sigma_n$ since $P - P_0$ is infinite and each $\sigma_i$ has only finitely many extensions in $P$. Thus there exists some initial segment $\sigma \in T - T_0$ of $y$ which is incompatible with each of $\sigma_0, \ldots, \sigma_n$. Just take $\sigma_{n+1}$ to be the least such $\sigma$ (first under length and then lexicographically). The key conclusion now is that since $T$ has no dead ends, each interval $I(\sigma_n)$ must contain a point $x_n$ of $P - P_0$. Now consider the $\Pi^0_1$ class $P_1 = \{x \in P : (\forall n)\neg(\sigma_{2n} \prec x)\}$. Since each $\sigma_k \notin T_0$, we have $P_0 \subset P_1$ and in addition $x_{2n-1} \in P_1$ for each $n$. Thus $P_1$ is distinct modulo finite difference from the three subclasses which make up $\mathcal{L}(P)^*$. This contradiction demonstrates the result. □

Note that we are not assuming in the previous theorem that $\mathcal{L}(P)$ is closed under finite differences. In particular, we are not assuming that every limit point of $P$ is non-computable.

For the special case of a single node, there does exist a computable tree $T$ with no dead ends such that $P = [T]$ is a minimal $\Pi^0_1$ class.

In the next section, we consider the general problem of a decidable $\Pi^0_1$ class where $\mathcal{L}(P)$ is not a Boolean algebra.

## 16.4 Decidable $\Pi^0_1$ classes

In this section, we consider in more detail the theory of the lattice $\mathcal{L}(P)$ of $\Pi^0_1$ subclasses of a decidable $\Pi^0_1$ class when $\mathcal{L}(P)$ is not a Boolean algebra. By Theorem 16.3.8 this means that either $P$ is uncountable or $\mathcal{L}(P)^*$ is infinite. We prove the following theorem.

**Theorem 16.4.1.** *Suppose that $P$ is a decidable $\Pi^0_1$-class such that $\mathcal{L}(P)$ is not a Boolean algebra. Then $Th(\mathcal{L}(P))$ interprets $Th(\mathbb{N})$.*

Let $\mathcal{D}$ be the computable dense Boolean algebra. For ease of notation and also to conform with Nies [148], we will use the language of c.e. ideals of $\mathcal{D}$ under inclusion instead of $\Pi^0_1$-classes under inclusion. If $H$ is a c.e. ideal of $\mathcal{D}$, let $\mathcal{L}(H)$ be the lattice of c.e ideals of $\mathcal{D}$ *containing* $H$.

**Theorem 16.4.2.** *Suppose $H$ is a computable ideal of $\mathcal{D}$ and $\mathcal{L}(H)$ is not a Boolean algebra. Then $Th(\mathcal{L}(H))$ interprets $Th(\mathbb{N})$.*

The equivalence of Theorem 16.4.1 and the preceding theorem can be obtained using effective Stone duality. See Cenzer and Remmel [36] for details. It will be clear from the proof how the decidability of $H$ is used: this enables one to see that a requirement is satisfied permanently, when it depends on the fact that a certain element of $\mathcal{D}$ which has been enumerated into an ideal is not in $H$.

We first need some terminology and notation. A c.e. Boolean algebra is given by a model $(\mathbb{N} \preceq, \vee, \wedge)$ such that $\preceq$ is a c.e. relation which is a pre-ordering, $\vee, \wedge$ are total computable binary functions, and the quotient structure $\mathcal{B} = (\mathbb{N}, \preceq, \vee, \wedge)/\approx$ is a Boolean algebra (where $n \approx m \iff n \preceq m \wedge m \preceq n$). We can suppose that $0 \in \mathbb{N}$ names the least and $1 \in \mathbb{N}$ the greatest element of $\mathcal{B}$. For $\Sigma^0_k$-Boolean algebras, one requires that $\preceq$ be $\Sigma^0_k$ and that $\wedge, \vee$ be computable in $\emptyset^{(k-1)}$. For a $\Sigma^0_k$ Boolean algebra $\mathcal{B}$, let

$$\mathcal{I}(\mathcal{B}) \ := \ \text{the lattice of } \Sigma^0_k\text{–ideals of } \mathcal{B}.$$

Clearly c.e. Boolean algebras correspond to c.e. ideals of $\mathcal{D}$, and similarly for computable. In this language, Theorem 16.4.2 can be restated a further time as follows: for a computable Boolean algebra $\mathcal{C}$, if $\mathcal{I}(\mathcal{C})$ is not a Boolean algebra, then $\mathrm{Th}(\mathcal{I}(\mathcal{C}))$ interprets $\mathrm{Th}(\mathbb{N})$.

*Proof.* We will first prove the weaker result that $\mathrm{Th}(\mathcal{L}(H))$ is undecidable, and then obtain the full result by an extra argument. We use a result from Nies [148]. A c.e. Boolean algebra $\mathcal{B}$ is called *effectively dense* [148] if there is a computable $F$ such that $\forall x\ [F(x) \prec x]$ and

$$\forall x \not\approx 0 \ \ [0 \prec F(x) \prec x]. \tag{16.1}$$

More generally, a $\Sigma^0_k$ Boolean algebra $\mathcal{B}$ is effectively dense if the above holds with some $F \leq_T \emptyset^{(k-1)}$. In [148], it is proved that, for any effectively dense $\Sigma^0_k$ Boolean algebra $\mathcal{C}$, $\mathrm{Th}(\mathcal{I}(\mathcal{C}))$ is hereditarily undecidable (i.e., all subtheories containing the valid sentences are undecidable). By the standard methods to transfer hereditary undecidability (see e.g. [15]), it suffices to give a coding in $\mathcal{L}(H)$ with parameters of $\mathcal{I}(\mathcal{C})$, for an effectively dense $\Sigma^0_3$ Boolean algebra $\mathcal{C}$.

In the following we describe how to determine $\mathcal{C}$ and how to do the coding. We first need some more notation.

**Definition 16.4.3.** *1. For $S \subset \mathcal{D}$, let $\mathcal{I}(\mathcal{D})S$ be the ideal of $\mathcal{D}$ generated by $S \cup H$.*

2. *A enumeration of an ideal $X$ of $\mathcal{B} = \mathcal{D}/_{\mathbb{N}^*H}$ is given by a c.e. subset $\widetilde{X} = \bigcup_s \widetilde{X}_s$ of $\mathcal{D}$ such that $X = \mathcal{I}(\mathcal{D})\widetilde{X}$. We let $X_s = \mathcal{I}(\mathcal{D})\widetilde{X}_s$ (thereby slightly deviating from the notation in [149], where $H$ is usually not decidable). We let $(V_e)$ be a uniform enumeration of all c.e. ideals containing $H$.*

3. *For a c.e. ideal $X$, we let*

$$x_0 = 0, \ x_s = \sup_{\mathcal{D}} X_s - \sup_{\mathcal{D}} X_{s-1} (s > 0). \tag{16.2}$$

*Thus, $(x_n)_{n\in\mathbb{N}}$ is an effective "partition" generating $X$.*

4. *Capital letters $A, \ldots, E, X, Y, V, W$ range over c.e. ideals of $\mathcal{D}$ containing $H$.*

5. *An element $b$ of $\mathcal{B}$ is identified with the corresponding principal ideal $\mathcal{I}(\mathcal{D})\{b\}$.*

6. *(Splittings of ideals) We write $B \sqcup C = A$ if $B \cap C = H$ and $B \vee C = A$. In this case we denote $C$ by $Cpl_A(B)$. We write $B \sqsubset A$ if $\exists C\ B \sqcup C = A$.*

Fix $A \in \mathcal{I}(\mathcal{B})$ and choose a $\emptyset''$-listing $(X_i)$ of $\mathcal{B}(A)$, where

$$\mathcal{B}(A) = \{X : X \sqsubset A\}.$$

Since $\mathcal{I}(\mathcal{B})$ is a distributive lattice, $(\mathcal{B}(A), \cap, \vee, \mathrm{Cpl}_A, H, A)$ is a $\Sigma^0_3$-Boolean algebra (with the presentation determined by that listing). We consider ideals of $\mathcal{B}(A)$. To avoid confusion, we will write "IDEAL " when we mean such a level 2 ideal. For certain $A, E$ such that $A \subset E$, we will view

$$\mathcal{R}_E(A) = \{X \sqsubset E : X \subset A\}$$

as the IDEAL of negligible splittings of $A$. Note that $\{e : X_e \in \mathcal{R}_E(A)\}$ is a $\Sigma^0_3$-set. Let

$$\mathcal{C} = \mathcal{B}(A/_E) = \mathcal{B}(A)/_{\mathbb{N}^*\mathcal{R}_E(A)}. \tag{16.3}$$

We first give an outline of the coding. Under certain conditions on $A$ and $E$ (for instance, if $E$ is nonprincipal), we will be able to show that $\mathcal{C}$ is effectively dense as a $\Sigma^0_3$ Boolean algebra. Then, to give the coding of $I(\mathcal{C})$ in $\mathcal{L}(H)$, we represent a $\Sigma^0_3$-IDEAL $I \in \mathcal{I}(\mathcal{C})$ by (any) $C \in \mathcal{L}(H)$ such that, for $X \sqsubset A$, $X \in \mathcal{I}(\mathcal{C})$ if and only if $X \cap C$ is negligible, that is, $X \cap C \subset R$ for some $R \in \mathcal{R}_E(A)$. Clearly, any subset of $\mathcal{B}(A)$ represented in this way is a $\Sigma^0_3$-IDEAL containing $\mathcal{R}_E(A)$. The main technical result, proved in [149], is that also each such IDEAL $I$ can be represented.

Then with the listing $(X_i/_E)_{i\in\omega}$, $\mathcal{B}(A/_E)$ becomes a $\Sigma^0_3$ Boolean algebra. To obtain the desired $\Sigma^0_3$ Boolean algebra, we require that $E$ is a nonprincipal ideal, $A \subset E$ is not a split of $E$ and $A$ also satisfies the following property.

**Definition 16.4.4.** *We say that $A$ is* locally principal *(l.p.) in $E$ if $A \subset E$ and*

$$\forall e \in E[e \cap A \textit{ is principal }].$$

Note that this property of $A, E$ can be expressed in $\mathcal{I}(\mathcal{B})$ in a first-order way, since the principal ideals are just the complemented elements of $\mathcal{I}(\mathcal{B})$. The motivation is that the situation $A \subset E$ is in a sense similar to an inclusion of sets: whenever $e \in E$, the intersection $A \cap e$ has only a finite amount of information.

Locally principal ideals were introduced by Nies in [149].

Since $\mathcal{L}(H)$ is not a Boolean algebra, a nonprincipal $E$ exists. We first supply the fact that an $A \subset E$ as required also exists.

**Lemma 16.4.5.** *For any $E \not\sqsubset 1$, there is $A \subset E$, $A \not\sqsubset E$ such that $A$ is l.p. in $E$.*

*Proof.* Since $E \not\sqsubset 1$, we can fix an enumeration $E_s = \mathcal{I}(\mathcal{D})\{e_n : n < s\}$, where $(e_n)$ is a u.c.e. sequence of elements of $D - H$ which have pairwise meet $H$. It suffices to meet the requirements

$$R_n : \neg(A \sqcup V_n = E).$$

To do so, we reserve $e_n$ for $R_n$. At stage $s$, for each $n < s$, if now $e_n \in V_{n,s}$, we put $e_n$ into $A$ (precisely speaking, into $\widetilde{A}$).

Clearly $A$ is l.p. in $E$. Moreover, each requirement is met: If $A \vee V_n = E$, then, since we threaten to keep $e_n$ out of $A$, $e_n \in V_{n,s}$ for some $s$. Then the construction ensures $V_n \cap A \neq H$. $\square$

In the following we fix $A, E$ with the properties as above. We prove that the $\Sigma^0_3$ Boolean algebra $\mathcal{B}(A/_E)$ is effectively dense. Clearly if $A$ is l.p. in $E$ and $Y \sqsubset A$, then so is $Y$. So the following is sufficient.

**Lemma 16.4.6.** *Suppose $Y$ is l.p. in $E$. Then one can effectively obtain a splitting $Y = Y_0 \sqcup Y_1$ such that $Y \not\sqsubset E$ implies $Y_i \not\sqsubset E$ $(i = 0, 1)$.*

*Proof.* Let $E = \mathcal{I}(\mathcal{D})\{e_n : n \in \mathbb{N}\}$ as above. We call $S \subset E$ *small* if

$$\exists n S \subset e_0 \vee \ldots \vee e_n.$$

For c.e. ideals $C, D$ let $C \searrow D$ be the ideal $X$ given by enumerating (into a set $\tilde{X}$) at stage $s$ those $x$ such that

$$x \in C_{s-1} \ \wedge \ x \notin D_{s-1} \ \wedge \ x \in D_s,$$

(and, as always, letting $X_s$ be the ideal of $\mathcal{D}_s$ generated by $\tilde{X}_s \cup H$). Similarly to the proof of the Friedberg Splitting Theorem [181], we meet the requirements

$$P_{e,i} : V_e \searrow Y \text{ not small } \Rightarrow \ V_e \searrow Y_i \not\subset H,$$

while ensuring that $Y = Y_0 \sqcup Y_1$.

We first verify that this is sufficient. Suppose $Y_i \sqsubset E$. Choose $k$ such that $Y_i \sqcup V_k = E \ \wedge \ Y_i \cap V_k = H$. Then $V_k \searrow Y$ is not small: assume

$$V_k \searrow Y \subset \hat{e}_n := e_0 \vee \ldots \vee e_n,$$

and let

$$V = \mathcal{I}(\mathcal{D})\{y \leq \mathrm{Cpl}_E(\hat{e}_n) : \exists s\ (y \in V_{k,s} \ \wedge \ y \notin Y_s)\}.$$

Then

$$(\hat{e}_n \vee Y) \vee V = E \ \wedge \ (\hat{e}_n \vee Y) \cap V = H.$$

Thus $(\hat{e}_n \vee Y) \sqsubset E$, and since $Y$ is l.p. in $E$, $Y \sqsubset E$.

Since $V_k \searrow Y$ is not small, $V_k \searrow Y_i \not\subset H$, contrary to $V_k \cap Y_i = H$. So it suffices to meet the requirements $P_{e,i}$.

*Construction of* $Y_0, Y_1$. At stage $s$ determine the least $\langle e, i\rangle < s$ such that $P_{e,i}$ has not been met (i.e., $V_e \searrow Y_i[s] \subset H$ and $y_s \cap V_{e,s-1} \not\subset H$). Enumerate $y_s$ into $Y_i$. If $\langle e, i\rangle$ fails to exist, put $y_s$ into $Y_0$.

Clearly, $Y = Y_0 \sqcup Y_1$. To prove that $P_{e,i}$ is met, suppose that by stage $t$, $P_k$ has been met for each $k < \langle e, i\rangle$. Since $V_e \searrow Y$ is not small, there is $s > t$ such that $y_s \notin H$, $y_s \wedge \hat{e}_t \in H$ and $y_s \cap V_{e,s-1} \not\subset H$. Then the requirement is satisfied from stage $s + 1$ on. $\square$

Since $\mathcal{B}(A/_E)$ is an effectively dense $\Sigma_3^0$ Boolean algebra, by Nies [148], the lattice $\mathcal{I}(\mathcal{B}(A/_E))$ has a hereditarily undecidable theory. Therefore it is sufficient to give a coding with parameters of $\mathcal{I}(\mathcal{B}(A/_E))$ in $\mathcal{L}(H)$. We rely on the proof of Nies [149, Lemma 6.3], where it is shown that, if $A$ is l.p. in $E$, then, for each $\Sigma_3^0$ IDEAL $I$ of $\mathcal{B}(A)$ containing $\mathcal{R}_E(A)$, there is a $C_I$ such that

$$I = \{X \in \mathcal{B}(A) : (\exists R \in \mathcal{R}_E(A))(C_I \cap X \subset R)\}. \tag{16.4}$$

(In [149] the assumption is made in the proof of Lemma 6.3 that the base Boolean algebra $\mathcal{D}/_H$ is effectively dense, but this is not needed.) Note that, conversely, each subset of $\mathcal{B}(A)$ determined by (16.4) is a $\Sigma_3^0$ IDEAL containing $\mathcal{R}_E(A)$. Since the set of these IDEALS corresponds to $\mathcal{I}(\mathcal{B}(A/_E))$, we obtain the desired coding: represent $I$ by any $C_I$, and give a first-order formula $\phi_\subset(C_1, C_2; A, E)$ expressing inclusion of the represented ideals in the obvious way.

This settles undecidability. We now give the extra argument needed to obtain an interpretation of Th($\mathbb{N}$) in Th($\mathcal{L}(H)$). First we prove a uniqueness property of $\mathcal{B}(A/_E)$.

**Proposition 16.4.7.** *Suppose that*

$$E \not\sqsubset 1,\ A \subset E,\ A \not\sqsubset E,\ A \textit{ is l.p. in } E, \tag{16.5}$$

*and the same properties also hold for* $\widetilde{E}, \widetilde{A}$. *Then* $\mathcal{B}(A/_E) \cong \mathcal{B}(\widetilde{A}/_{\widetilde{E}})$ *via an isomorphism which is computable in* $\emptyset''$.

*Proof.* A c.e. Boolean algebra $\mathcal{C}$ is called *effectively inseparable* (e.i.) if the sets $\{n \in \mathbb{N} : n \approx 0\}$, $\{n \in \mathbb{N} : n \approx 1\}$ (i.e. the sets of names for $0^{\mathcal{C}}, 1^{\mathcal{C}}$) are effectively inseparable. By the methods of Kripke and Pour-El [**?**], any two e.i. Boolean algebras are effectively isomorphic. We apply their result, relativized to $\emptyset''$. It suffices to show that under the given hypotheses $\mathcal{B}(A/_E)$ (with the presentation given at (16.3)) is $\emptyset''$-e.i.. Recall that $(X_i)$ is an $\emptyset''$-listing of $\mathcal{B}(A)$. We prove that

$$S = \{i : X_i \in \mathcal{R}_E(A)\},\ T = \{i : A - X_i \in \mathcal{R}_E(A)\} \tag{16.6}$$

are $\emptyset''$-e.i. sets. Fix a pair of $\Sigma_3^0$-sets $\widetilde{S}, \widetilde{T}$ which is e.i. relative to $\emptyset''$. It suffices to find a total $f \leq \emptyset''$ such that

$$f(\widetilde{S}) \subset S, f(\widetilde{T}) \subset T. \tag{16.7}$$

Fix a u.c.e. double sequence $Z_n^i$ of initial segments of $\mathbb{N}$ such that $i \in \widetilde{S} \iff \exists n Z_{2n}^i = \mathbb{N}$ and $i \in \widetilde{T} \iff \exists n Z_{2n+1}^i = \mathbb{N}$. For each $i$ we will effectively obtain a splitting $A = A_0 \sqcup A_1$ such that $i \in \widetilde{S} \Rightarrow A_0 \sqsubset E$ and $i \in \widetilde{T} \Rightarrow A_1 \sqsubset E$. Then $f$, given by $f(i) =$ the first $j$ such that $X_j = A_0$, is a function computable in $\emptyset''$ as desired. We employ a simple fact from Nies [149, Fact 6.1]. Recall that we are identifying elements of $\mathcal{B}$ and principal ideals.

**Fact 16.4.8.** *Suppose $B \subset E$ is a c.e. ideal such that $\forall k\ B \cap e_k = b_k$, where $b_k$ is obtained effectively from $k$. Then $B \sqsubset E$.*

*Proof of the Fact.* Let $C = \mathcal{I}(\mathcal{D})\{e_k - b_k\}_{k\in\mathbb{N}}$. Then $B \sqcup C = E$. $\square$

At stage $s$, we decide whether to put $a_s$ into $A_0$ or into $A_1$, as follows: Compute the maximal $k$ such that $a_s \wedge e_k \notin H$. Let $m$ be minimal such that $|Z^i_{m,s}| > k$. If $m$ is even or fails to exist put $a_s$ into $A_1$, else into $A_0$.

To verify (16.7), first suppose $i \in \widetilde{S}$ and let $m$ be least such that $Z^i_{2m} = \mathbb{N}$. Then Fact 16.4.8 implies $B = A_0 \sqsubset E$ as follows. Given $k$, since $B$ is l.p. in $A$ we can assume that $k > \max_{r<2m} |Z^i_r|$ (because finitely many $b_i$ can be fixed in advance). Compute $s$ such that $|Z^i_{2m,s}| > k$. Then $A_0 \cap e_k = A_{0,s} \cap e_k$, so let $b_k = \sup(A_{0,s} \cap e_k)$. If $i \in \widetilde{T}$, one proves $A_1 \sqsubset E$ in a similar fashion. This completes the proof of Proposition 3.7.

By the uniqueness up to $\emptyset''$ isomorphism of $\mathcal{B}(A/E)$, all possible structures $\mathcal{I}^* = \mathcal{I}(\mathcal{B}(A/E))$, where $E, A$ satisfy (16.5), are isomorphic. By Nies [149] and the effective density of $\mathcal{B}(A/E)$, Th$(\mathcal{I}^*)$ interprets Th$(\mathbb{N})$. But

$$\mathcal{I}^* \models \phi \iff \mathcal{L}(H) \models \exists E \exists A[(16.5)\ \wedge\ ''\mathcal{I}(\mathcal{B}(A/_E)) \models \phi''],$$

so Th$(\mathcal{I}^*)$ can be interpreted in Th$(\mathcal{L}(H))$.

This demonstrates fact 16.4.8 and completes the proof of Proposition **??**, Theorems 16.4.1 and 16.4.2. $\square$

$\square$

**Open question:** Characterize those $P$ such that Th$(\mathcal{L}(P))$ is decidable.

## 16.5 Global Properties of the Lattice

## 16.6 Almost complemented classes

## 16.7 Perfect thin classes

The proof of the Low Basis Theorem 3.1.4 shows that every nonempty c.b. $\Pi^0_1$ class contains a member of c. e. degree $\mathbf{a}$ such that $\mathbf{a}$ is low, that is, $\mathbf{a} \oplus \mathbf{0}' = \mathbf{a}' = \mathbf{0}''$. The method of Theorem 2.8.1 can be used to construct a nonempty $\Pi^0_1$ class with no computable members and no members of *high* degree, where the c. e. degree $\mathbf{a}$ is high if $\mathbf{a}' = 0''$.

**Theorem 16.7.1.** *There exists a perfect, thin, c.b. $\Pi^0_1$ class $P$ with no computable members such that if $\mathbf{a}$ is the degree of a member of $P$, then $\mathbf{a}' \leq a \oplus 0'$.*

*Proof.* Let $P$ be the $\Pi^0_1$ class constructed in Theorem 2.8.1 and let $f$ be the function defined therein. Then $f$ is the limit of a uniformly computable sequence of functions and is therefore computable in $\mathbf{0}'$ by the Limit Lemma. Now let $U_e = \{\sigma : \phi^\sigma_e(e)\uparrow\}$. It follows from the s-m-n theorem that there is a computable function $\phi$ such that $U_e = T_{\phi(e)}$ for each $e$. Then for any element $A$ of $P$ and any $e$, we have

$e \in A' \iff (\exists \sigma \prec A)\sigma \in U_e \iff x\lceil f(2e+2) \in T_{\phi(e)}$.

This shows that $A'$ is computable in $A \oplus \mathbf{0}'$. $\square$

xxx

There will be results here from the paper [45]

# Chapter 17

# Degrees of Difficulty

The Medvedev lattice was introduced in [136] to classify problems according to their degree of difficulty. A *mass problem* is a subset of $\mathbb{N}^{\mathbb{N}}$ and is thought of as representing the set of solutions to some problem. For example, the *problem of separability* of sets $A$ and $B$ is $S(A,B) = \{f : i \in A \rightarrow f(i) = 0 \ \& \ i \in B \rightarrow f(i) = 1\}$. The *coloring problem* for a given countably infinite graph $G$ may be given as a set of functions each mapping $\omega$ into $\{1,2,3,4\}$. A mass problem is said to be *solvable* if it contains a computable function. See the survey paper by Sorbi [183] for more background.

In this chapter we study the Medvedev and Muchnik degrees of nonempty $\Pi^0_1$ classes. Each of these partial orderings is in fact a distributive lattice with top element, which can be viewed as the degree of the set of completions of Peano arithemtic, and bottom element, which is the degree of any set containing a computable member.

## 17.1 Reducibility

$P$ is *Medvedev reducible* to $Q$ ($P \leq_M Q$) if there is a partial computable functional $\Phi$ which is defined for all $X \in Q$ and maps $Q$ into $P$. Thus any solution of $Q$ may be used to compute a solution of $P$, so we say that $P$ has a lower (Medvedev) degree of difficulty than $Q$. There is also a nonuniform notion, Muchnik reducibility, given by $P \leq_w Q$ if every member $X$ of $Q$ computes a member of $P$, that is, $Y \leq_T X$ for some $Y \in P$. As usual, $P \equiv_M Q$ means that both $P \leq_M Q$ and $Q \leq_M P$, $P <_M Q$ means $P \leq_M Q$ but not $Q \leq_M P$, and the Medvedev degree $\mathbf{dg}_M(P)$ of $P$ is the class of all sets $Q$ such that $P \equiv_M Q$. Similar notations applies to Muchnik reducibility. Observe that $P \leq_M Q$ implies $P \leq_w Q$, so that the Medvedev degree of $P$ is a subset of the Muchnik degree of $P$. Let $\mathcal{P}_M$ denote the lattice of Medvedev degrees of $\Pi^0_1$ classes and let $\mathcal{P}_w$ denote the lattice of Muchnik (or *weak*) degrees.

We will focus primarily on the Medvedev degrees.

First we show that only total functionals are needed for Medvedev reducibility of $\Pi^0_1$ classes.

**Lemma 17.1.1.** *For any $\Pi^0_1$ subclasses $P$ and $Q$ of $\omega^\omega$, if $P \leq_M Q$, then there exists a total computable functional $F : \mathbb{N}^\mathbb{N} \to \mathbb{N}^\mathbb{N}$ such that $F[Q] \subseteq P$.*

*Proof.* Given that $P \leq_M Q$, there is a partial computable functional $\Phi$ which maps $Q$ into $P$. This means that there is a partial computable function $\phi$ mapping finite sequences to finite sequences such that $\Phi(X) = \cup_n \phi(X\lceil n)$ and with the property that $\sigma \prec \tau$ implies $\phi(\sigma) \prec \phi(\tau)$. Now $Q$ may be expressed as the set of infinite paths through some computable tree $T$. Then we can extend the mapping $\Phi$ from $Q$ to a total mapping $F$ representing function $f$ defined recursively as follows. Let $f(\emptyset) = \emptyset$. Then for any finite sequence $\sigma$ and any $n$, define $f(\sigma^\frown n)$ in two cases. If $\sigma^\frown n \in T$, let $f(\sigma^\frown n) = \phi(\sigma^\frown n)$, which must be defined. If $\sigma^\frown n \notin T$, let $f(\sigma^\frown n) = f(\sigma)^\frown 0$. □

Simpson [178] has observed that Medvedev reducibility can be viewed as the analog for degrees of difficulty of the truth-table degrees for the Turing degrees, whereas Muchnik reducibility is the analog of Turing reducibility.

**Proposition 17.1.2.** *If $P \leq_M Q$, then for any $x \in Q$, there is a $y \in P$ such that $y \leq_{tt} x$.*

*Proof.* Suppose $P \leq_M Q$ for $\Pi^0_1$ classes $P$ and $Q$. Then by Lemma 17.1.1, there is a total computable $\Phi$ which maps $Q$ into $P$. Then for any element $x$ of $Q$, it follows from Lemma 1.9.10 that $\Phi(x) \leq_{tt} x$. □

The meet and join operations of the Medvedev lattice turn out to be the standard sum $P \oplus Q$ and product $P \otimes Q$ defined earlier. We summarize here some basic facts about these meet and join operations.

**Proposition 17.1.3.** *For any $\Pi^0_1$ classes $P$, $Q$ and $R$,*

*(i) $P \oplus Q \equiv_M Q \oplus P$ and $P \otimes Q \equiv_M Q \otimes P$ (so that also $P \oplus Q \equiv_w Q \oplus P$ and $P \otimes Q \equiv_w Q \otimes P$*

*(ii) The Medvedev (Muchnik) degree of $P \oplus Q$ is the meet, or greatest lower bound, of the Medvedev (resp. Muchnik) degrees of $P$ and $Q$;*

*(iii) The Medvedev degree of $P \otimes Q$ is the join, or least upper bound, of the Medvedev degrees of $P$ and $Q$*

*(iv) $P \otimes (Q \oplus R) \equiv_M (P \otimes Q) \oplus (P \otimes R)$ and $P \otimes (Q \oplus R) \equiv_M (P \otimes Q) \oplus (P \otimes R)$.*

*(v) If $P \leq_M Q$, then, for any $R$, $(P \otimes R) \oplus Q \equiv_M P \otimes (Q \oplus R)$; if $P \leq_w Q$, then, for any $R$, $(P \otimes R) \oplus Q \equiv_w P \otimes (Q \oplus R)$*

*Proof.* (i) is obvious since these sets are in fact computably homeomorphic.

(ii) $P \oplus Q \leq_M P$ via the map $\Phi(x) = 0^\frown x$ and similarly $P \oplus Q \leq_M Q$. Suppose now that $R \leq P$ via $\Phi$ and $R \leq Q$ via $\Psi$. Then $R \leq P \oplus Q$ via the map taking $0^\frown x$ to $\Phi(x)$ and $1^\frown x$ to $\Psi(x)$.

(iii) $P \leq_M P \otimes Q$ via the map $\Phi(x) = (x(0), x(2), \dots)$ and similarly $Q \leq_M P \otimes Q$. Suppose now that $P \leq_M R$ via $\Phi$ and $Q \leq_M R$ via $\Psi$. Then $P \otimes Q \leq_M R$ via the map taking $x$ to $\langle \Phi(x), \Psi(x) \rangle$.

(iv) To see that $P \oplus (Q \otimes R) \equiv_M (P \oplus Q) \otimes (P \oplus R)$, we define computable functionals in each direction. First define $\Phi : P \oplus (Q \otimes R) \to (P \oplus Q) \otimes (P \oplus R)$ by

$$\Phi(0^\frown X) = \langle 0^\frown X, 0^\frown X \rangle$$

and

$$\Phi(1^\frown \langle Y, Z \rangle) = \langle 1^\frown Y, 1^\frown Z \rangle.$$

Then define $\Psi : (P \oplus Q) \otimes (P \oplus R) \to P \oplus (Q \otimes R)$ as follows. Given $Z = \langle V, W \rangle \in (P \oplus Q) \otimes (P \oplus R)$, there are three cases.

$$\text{If } V = 0^\frown X, \text{ let } \Psi(Z) = V;$$
$$\text{if } V = 1^\frown Y \text{ and } W = 0^\frown X, \text{ let } \Psi(Z) = W;$$
$$\text{if } V = 1^\frown Y \text{ and } W = 1^\frown Z, \text{ let } \Psi(Z) = 1^\frown \langle Y, Z \rangle.$$

The other equivalence of (iv) is left as an exercise.

(v) Since $P \leq_M Q$, we have $P \oplus Q \equiv_M P$ and $P \otimes Q \equiv_M Q$. Then

$$(P \otimes R) \oplus Q \equiv_M (P \oplus Q) \otimes (R \oplus Q) \equiv_M P \otimes (Q \oplus R).$$

The same argument works for the Muchnik degrees. □

**Corollary 17.1.4.** *Both $\mathcal{P}_M$ and $\mathcal{P}_w$ are distributive lattices.*

Next we observe that $\mathcal{P}_M$ has both a least and a greatest element. The least element $\mathbf{0}$ consists of all classes $P$ which contain a computable element. To see this, just let $X_0$ be a computable element of $P$ and define $F(X) = X_0$ for any $X$. Then $F$ maps any class $Q$ into $P$, so that $P \leq_M Q$. In particular, the classes $\{0,1\}^\omega$ and $\{0^\omega\}$ are both in $\mathbf{0}$. Sorbi points out in [183] that this means that the solvable Medvedev degree is *definable* in the lattice (as the least element).

**Proposition 17.1.5.** *$\mathcal{P}_M$ has a greatest element.*

*Proof.* Since there is an enumeration $\{P_e\}_{e \in \omega}$ of the $\Pi^0_1$ classes, it seems natural to take the product of these classes as the univeral set and hence the top Medvedev degree. There is one hitch in that the empty set is not included in $\mathcal{P}_M$. To enumerate the *nonempty* $\Pi^0_1$ classes, first recall the usual enumeration $\{T_e\}_{e \in \mathbb{N}}$ of the primitive recursive trees in $\{0,1\}^*$ and let

$$\sigma \in T_e^+ \iff [\sigma \in T_e \ \vee \ (\forall m \leq |\sigma|)(\sigma\lceil m \notin T_e \to (\forall \tau \in \{0,1\}^m)\tau \notin T_e)].$$

Now if $P_e = \emptyset$ and $m$ is the least such that $T_e \cap \{0,1\}^{m+1} = \emptyset$, then it is clear that $P_e^+ = [T_e^+] = \bigcup\{I(\sigma) : \sigma \in \{0,1\}^m \cap T_e\}$ and is still a $\Pi^0_1$ class. If $P_e \neq \emptyset$, $P_e^+ = P_e$. We claim that $\prod_e P_e^+$ is the greatest element of $\mathcal{P}_M$ and hence also of $\mathcal{P}_w$. That is, for any nonempty $\Pi^0_1$ class $P_e$, the projection map $\pi_e$ takes $\prod_e P_e^+$ to $P_e^+$. □

## 17.2 Completeness

Let $\mathcal{B}$ denote the computable Boolean algebra of clopen sets in $\{0,1\}^{\mathbb{N}}$; recall that these are finite unions of intervals $I(\sigma)$. Note that $\mathcal{B}$ is computably isomorphic to the Boolean algebra of propositional logic over an infinite set of variables–see Section 8.1 of Chapter 8. In particular, let $b_n \in \mathcal{B}$ denote $\{x : x(n) = 1\}$.

**Definition 17.2.1.** *Let $P$ be a nonempty subset of $\{0,1\}^{\mathbb{N}}$. A* splitting function *for $P$ is a computable function $g : \mathbb{N} \to \mathcal{B}$ such that, for all $e$, if $P_e \subseteq P$ and $P_e \neq \emptyset$, then $P_e \cap g(e)$ and $P_e - g(e)$ are both nonempty. $P$ is said to be* productive *if it has a splitting function.*

Clearly a productive $\Pi^0_1$ class can have no subset that is a singleton and hence can have no computable member. It follows in particular that $P$ is nowhere dense.

We observe that the class $DNC_2$ of diagonally non-computable functions in $\{0,1\}^{\mathbb{N}}$ is productive. This will be shown in the next section.

**Theorem 17.2.2.** *(Simpson) For any productive $\Pi^0_1$ class $P \subseteq \{0,1\}^{\mathbb{N}}$ and any nonempty $\Pi^0_1$ class $Q \subseteq \{0,1\}^{\mathbb{N}}$, there is a computable functional $\Phi$ from $P$ onto $Q$. Thus any productive class is Medvedev complete.*

*Proof.* Let $P$ and $Q$ be $\Pi^0_1$ subsets of $\{0,1\}^{\mathbb{N}}$ and suppose that $P$ is productive. We will define a computable monomorphism $f : \mathcal{B} \to \mathcal{B}$ such that, for all $b \in \mathcal{B}$, $Q \cap b = emptyset \iff P \cap f(b) = \emptyset$. Then the computable map $\Phi : P \to Q$ is defined by letting $Y = \Phi(X)$ be the unique element of $Q$ such that $X \in f(Y\lceil n) \neq \emptyset$ for all $n$. The approximating function for $\Phi$ is something like the inverse of $f$. For an arbitrary element $Y \in Q$, $Q \cap I(Y\lceil n) \neq \emptyset$ for all $n$ and hence $P \cap I(f(Y\lceil n)) \neq \emptyset$ for all $n$, so that $\cap_n[P \cap I(f(Y\lceil n))] \neq \emptyset$ and any element of this set will map to $Y$. This shows that $\Phi$ will map $X$ onto $Y$.

It clearly suffices to define $f$ for intervals $I(\sigma)$ and for ease of notation we will just write $f(\sigma)$ for $f(I(\sigma))$. Then $f(\sigma) = \Phi^{-1}(I(\sigma))$ under the function $\Phi$ defined above. That is, if $Y = \Phi(X)$, then $X \in f(\sigma) \iff \sigma \preceq Y$.

The function $f$ is defined recursively beginning with $f(\emptyset) = \emptyset$.

For the recursive step, suppose that $f(\sigma) = a$ is given so that $Q \cap I(\sigma) = \emptyset \iff P \cap a = \emptyset$ and that $a \subseteq I(\tau)$ for some $\tau$ with $|\tau| \geq |\sigma|$. We will show how to compute $f(\sigma^\frown 0) = a_0$ and $f(\sigma^\frown 1) = a_1$ such that each $a_i$ is included in some $I(\tau_i)$ with $|\tau_i| > |\tau|$ and such that

$$\sigma^\frown i \in T_Q \iff P \cap a_i \neq \emptyset.$$

Since $P$ is productive, it is nowhere dense so we can partition $a$ non-trivially into $b_0 \cup b_1 \cup b_2$ so that $P \cap b_0 = P \cap b_1 = \emptyset$ and hence $P \cap a = P \cap b_2$.

By the Recursion Theorem, we can compute $e \in \mathbb{N}$ such that

$$P_e = \begin{cases} P \cap b_2 & \text{if } \ \sigma^\frown 0 \in T_Q \text{ and } \sigma^\frown 1 \in T_Q; \\ P \cap b_2 \cap g(e) & \text{if } \ \sigma^\frown 0 \notin T_Q \text{ and } \sigma^\frown 1 \in T_Q; \\ P \cap b_2 - g(e) & \text{if } \ \sigma^\frown 0 \in T_Q \text{ and } \sigma^\frown 1 \notin T_Q; \\ \emptyset & \text{if } \ \sigma^\frown 0 \notin T_Q \text{ and } \sigma^\frown 1 \notin T_Q. \end{cases}$$

Now let $a_0 = b_0 \cup (b_2 \cap g(e))$ and $a_1 = \{0,1\}^{\mathbb{N}} - a_0 = b_1 \cup (b_2 - g(e))$. We claim the following:

1. $Q \cap I(\sigma^\frown 0) = \emptyset \iff P \cap a_0 = \emptyset$;
2. $Q \cap I(\sigma^\frown 1) = \emptyset \iff P - a_1 = \emptyset$.

There are several cases to check. Suppose first that $\sigma \notin T_Q$. Then by assumption $P \cap a = \emptyset$, so that we have $P \cap a_0 = P \cap a_1 = \emptyset = Q \cap I(\sigma^\frown 0) = Q \cap I(\sigma^\frown 1)$. Now suppose that $\sigma \in T_Q$, so that $Q \cap I(\sigma)$ and $P \cap a$ are both nonempty. There are three cases. First suppose that both $\sigma^\frown 0$ and $\sigma^\frown 1$ are in $T_Q$. Then $Q \cap I(\sigma^\frown 0)$ and $Q \cap I(\sigma^\frown 1)$ are both nonempty, so that $P_e = P \cap a \neq \emptyset$. It follows that $P \cap a_0 = P \cap b_2 \cap g(e) = P_e \cap g(e) \neq \emptyset$ (since $g$ is a splitting function for $P$) and similarly $P \cap a_1 \neq \emptyset$. Next suppose that $\sigma^\frown 0 \notin T_Q$ but $\sigma^\frown 1 \in T_Q$, so that $P_e = P \cap b_2 \cap g(e) = P \cap a_0$. Then $P_e - g(e) = \emptyset$ and, since $g$ is a splitting function for $P$, it follows that $P_e = \emptyset$, and hence

$$P \cap a_1 = (P \cap b_1) \cup (P \cap b_2 \cap g(e)) = \emptyset.$$

The remaining case where $\sigma^\frown 0 \in T_Q$ and $\sigma^\frown 1 \notin T_Q$ is similar. □

This result can be improved for two productive classes.

**Theorem 17.2.3.** *(Simpson) Any two productive $\Pi^0_1$ classes $P, Q \subseteq \{0,1\}^{\mathbb{N}}$ are computably homeomorphic.*

*Proof.* We simply add a back-and-forth argument to the proof of Theorem 17.2.3 to make the monomorphism onto. That is, at stage $n$, we will have a finite isomrphism $f_n : \mathcal{B}_n \simeq \mathcal{B}'_n$ where each of $\mathcal{B}_n$ and $\mathcal{B}'_n$ are finite subalgebras including $b_0, \ldots, b_n$ such that $P \cap a = \emptyset$ if and only if $Q \cap f_n(a) = \emptyset$. We start as above with $f((0)) = a_0$ and $f((1)) = a_1$ as above so that $P \cap a_i = \emptyset \iff Q \cap I((i)) = \emptyset$. Now use the splitting function for $P$ to obtain $b_{ij} \subset I(i)$ for $i, j \in \{0,1\}$ so that $a_i \cap (j) \cap Q = \emptyset \iff b_{ij} \cap P = \emptyset$ and let $\mathcal{B}_0$ be generated by $\{b_{ij} : i, j \in \{0,1\}\}$ and $\mathcal{B}_1$ be generated by $\{I((0)), I((1)), a_0, a_1\}$. Note that $f^{-1}(I(e)) = b_{e0} \cup b_{e1}$. We leave the details to the reader. □

**Question 17.2.4.** *Suppose in general that $P$ and $Q$ are $\Pi^0_1$ subsets of $\{0,1\}^{\mathbb{N}}$ and that there there exist mappings $\Phi$ from $P$ onto $Q$ and $\Psi$ from $Q$ onto $P$. Does it follows that $P$ and $Q$ are computably homeomorphic?*

**Lemma 17.2.5.** *(Simpson) Let $P$ and $Q$ be nonempty subsets of $\{0,1\}^{\mathbb{N}}$. If $P \leq_M Q$ and $P$ is productive, then $Q$ is productive.*

*Proof.* Since $P \leq_M Q$, there is a computable function $\Phi : Q \to P$. Define $f : \mathcal{B} \to \mathcal{B}$ by $f(b) = \Phi^{-1}(b)$. Let $g : \mathbb{N} \to \mathcal{B}$ be a splitting function for $P$. By the s-m-n Theorem, let $h : \mathbb{N} \to \mathbb{N}$ be primitive recursive such that $P_{h(e)} = \Phi(P_e \cap Q)$ for all $e$. We claim that the composition $f \circ g \circ h$ is a splitting function for $Q$. To see this, suppose that $P_e \subseteq Q$ and $P_e \neq \emptyset$. Then $P_{h(e)} = \Phi(P_e) \subset P$ and $P_{h(e)} \neq \emptyset$, so that $P_{h(e)} \cap g \circ h(e) \neq \emptyset$ and $P_{h(e)} - g \circ h(e) \neq \emptyset$ and therefore $P_e \cap f \circ g \circ h(e) \neq \emptyset$. □

**Corollary 17.2.6.** *Let $P \subset \{0,1\}^{\mathbb{N}}$ be a nonempty $\Pi^0_1$ class. Then $P$ is productive if and only if $P$ is Medvedev complete.*

*Proof.* By Theorem 17.2.2, if $P$ is productive, then $P$ is Medvedev complete. Lemma 17.2.5 implies that any complete $\Pi^0_1$ class is productive. □

Simpson and Slaman (unpublished) have shown the following.

**Theorem 17.2.7.** *Every nonzero degree in $\mathcal{P}_w$ contains infinitely many Medvedev degrees from $\mathcal{P}_M$.*

*Proof.* xxxx

□

**Exercises**

17.2.1. Prove parts (ii) and (iii) of Proposition 17.1.3 for the Muchnik degrees.

17.2.2. Show that $P \otimes (Q \oplus R) \equiv_M (P \otimes Q) \oplus (P \otimes R)$.

17.2.3. Show that the map taking the Medvedev degree of $P$ to the Muchnik degree of $P$ is a lattice homomorphism of $\mathcal{P}_M$ onto $\mathcal{P}_w$.

## 17.3 Separating Classes

Here is a general result which will provide a large class of Medvedev complete $\Pi^0_1$ classes. A pair of disjoint c. e. sets $A$ and $B$ are said to be *effectively inseparable* if there is a computable function $\phi$ such that, for any $x$ and $y$, if $A \subset W_x$ and $B \subset W_y$ and $W_x \cap W_y = \emptyset$, then $\phi(x,y) \notin W_x \cup W_y$. For example, it is well known that the set $PA = A_1$ of theorems of Peano Arithmetic and the set $B_1$ of negations of theorems of Peano Arithmetic are *effectively inseparable*–see Odifreddi [151], p 356. The following lemma will then imply that $S(A_1, B_1)$ is Medvedev complete.

**Proposition 17.3.1.** *If $A$ are $B$ are effectively inseparable c.e. sets, then $S(A,B)$ is a productive $\Pi^0_1$ class.*

*Proof.* Let $P = S(A,B)$ where $A$ and $B$ are effectively inseparable c. e. sets and and let $\phi$ be given as above. Define $W_{f(e)} = \{n : (\forall X \in P_e) n \in X\}$ and $W_{h(e)} = \{n : (\forall X \in P_e) n \notin X\}$. To see that these are indeed c. e. sets, note that $W_{f(e)}$ has an alternate definition, that is,

$$n \in W_{f(e)} \iff (\forall \sigma \in \{0,1\}^{n+1})(\sigma \in T_e \implies \sigma(n) = 1),$$

Clearly $W_{f(e)} \cap W_{h(e)} = \emptyset$, and if $P_e \subset P$, then $A \subset W_{f(e)}$ and $B \subset W_{h(e)}$. Thus $\phi(f(e), h(e)) = n \notin W_{f(e)} \cup W_{h(e)}$. Hence there exist $X$ and $Y$ in $P_e$ such that $n \in X$ and $n \notin Y$. The splitting function for $P$ can thus be defined by $g(e) = \{X : \phi(f(e), h(e)) \in X\}$. □

**Proposition 17.3.2.** *The $\Pi^0_1$ class $DNC_2$ is the separating class of a pair of effectively inseparable c. e. sets.*

*Proof.* Let $A = \{e : \phi_e(e) = 0\}$ and $B = \{e : \phi_e(e) = 0\}$. Then $S(A, B) = DNC_2$. Now suppose that $A \subseteq W_x$ and $B \subseteq W_y$ and suppose that $W_x \cap W_y = \emptyset$. We will show how to compute $\phi(x, y) = e$ such that $e \notin W_x \cup W_y$. Let $\psi(e, x, y) = 1$, if $e \in W_x$ and $= 0$, if $e \in W_y$. That is, $\psi(e, x, y)$ searches for the least $s$ such that $e \in W_{x,s} \cup W_{y,s}$ and then outputs 1 if $e \in W_{x,s}$ and outputs 0 if $e \in W_{y,s} - W_{x,s}$. Let $\phi(x, y) = e$ so that $\phi_e(i) = \psi(e, x, y)$. We claim that $e \notin W_x \cup W_y$. To see this, suppose that $e \in W_x$, so that by definition of $\psi, \phi$ and $e$, $\phi_e(e) = 1$. Then $e \in B$, which implies that $e \in W_y$ and contradicts $W_x \cap W_y = \emptyset$. The argument when $e \in W_y$ is similar. □

**Corollary 17.3.3.** *$DNC_2$ is Medvedev complete.*

*Proof.* By Propositions 17.3.2 and 17.3.1, $DNC_2$ is productive and hence by Theorem 17.2.2, it is Medvedev complete. □

The family of c.e. separating classes are closed under join, since

$$S(A, B) \otimes S(C, D) = S(\langle A, C\rangle, \langle B, D\rangle).$$

However, there is no non-trivial meet for c.e. separating classes, as shown by the following.

**Lemma 17.3.4.** *For any $\Pi^0_1$ class $P$ and any clopen sets $G$, and $H$, if $P \cap G \leq_M P \cap H$, then $P \cap G \equiv_M P \cap (G \cup H)$.*

*Proof.* First, $P \cap (G \cup H) \leq_M P \cap G$ via the identity map. Fix a computable functional $\Phi : P \cap H \to P \cap G$ and define $\Psi : P \cap (G \cup H) \to P \cap G$ by

$$\Psi(X) := \begin{cases} X, & \text{if } X \in G; \\ \Phi(X), & \text{otherwise.} \end{cases}$$

Note that $\Psi$ is computable since clopen sets are simply finite unions of intervals. □

**Lemma 17.3.5.** *For any c.e. separating class $P$ and any clopen set $G$, if $P \cap G \neq \emptyset$, then $P \cap G \equiv_M P$.*

*Proof.* By Lemma 17.3.4, it suffices to prove this for intervals, and we proceed by induction on the length $n$ of $\sigma$. If $n = 0$, then $I(\sigma) = 2^\omega$, so $P \cap I(\sigma) = P$. Assume as induction hypothesis that $P \cap I(\sigma) \equiv_M P$ for some $\sigma$ of length $n$, and suppose that $P \cap I(\sigma^\frown(e)) \neq \emptyset$. If $P \cap I(\sigma^\frown(1-e)) = \emptyset$, then $P \cap I(\sigma^\frown(e)) = P$. Otherwise, $P \cap I(\sigma^\frown(e)) \equiv_M P \cap I(\sigma^\frown(1-e))$ via the computable maps $X \mapsto X \cup \{0\}$ and $X \mapsto X/\{0\}$. Then by Lemma 17.3.4 again,

$$P \cap I(\sigma^\frown(e)) \equiv_M P \cap \big(I(\sigma^\frown(e)) \cup I(\sigma^\frown(1-e))\big) = P. \qquad \square$$

**Proposition 17.3.6.** *For any $\Pi^0_1$ classes $P$ and $Q$ and any c.e. separating class $R$, if $P \oplus Q \leq_M R$, then either $P \leq_M R$ or $Q \leq_M R$.*

*Proof.* Fix a computable functional $\Phi : R \to P \oplus Q$ and set $G := \{X : \Phi(X) \in I((0))\}$. $G$ is clopen as the continuous inverse image of an interval. $P \leq_M R \cap G$ via the map $X \mapsto \big(k \mapsto \Phi(X)(k+1)\big)$. If $R \cap G \neq \emptyset$, then by Lemma 17.3.5 $R \cap G \equiv_M R$, so $P \leq_M R$. Otherwise $R \setminus G \neq \emptyset$ and we have similarly $Q \leq_M R$. □

This suggests that we should consider the sublattice of $\mathcal{P}_M$ generated by the family of c.e. separating degrees. This turns out to have a simple direct characterization.

**Definition 17.3.7.** *For any tree $T \subseteq \{0,1\}^{<\omega}$ and any $\Pi^0_1$ class $P \subseteq \{0,1\}^\omega$,*

*(i) $T$ is* homogeneous *iff* $(\forall \sigma, \tau \in T)(\forall i < 2)$,

$$|\sigma| = |\tau| \implies (\sigma^\frown i \in T \iff \tau^\frown i \in T);$$

*(ii) $T$ is* almost homogeneous *iff* $\exists n (\forall \sigma, \tau \in T)(\forall i < 2)$,

$$n \leq |\sigma| = |\tau| \wedge \sigma \restriction n = \tau \restriction n \implies (\sigma^\frown i \in T \iff \tau^\frown i \in T);$$

*The least such $n$ is called the* modulus *of $T$;*

*(iii) $P$ is* (almost) homogeneous *iff $T_P$ is (almost) homogeneous; a Medvedev degree is* (almost) homogeneous *iff it contains an (almost) homogeneous class;* **AH** *denotes the family of almost homogeneous degrees.*

**Proposition 17.3.8.** *For any $\Pi^0_1$ class $P$,*

$$P \text{ is homogeneous} \iff P \text{ is a c.e. separating class.}$$

*Proof.* If $P = S(A, B)$ for c.e. sets $A$ and $B$, then

$$T_P = \{\sigma : (\forall i < |\sigma|)\big[\sigma(i) = 0 \wedge i \notin A) \vee (\sigma(i) = 1 \wedge i \notin B)\big]\}.$$

This is clearly a homogeneous tree. Conversely, if $T_P$ is homogeneous, then $P = S(A, B)$ for

$$A = \{n : 0^{n\frown}0 \notin T_P\} \qquad \text{and} \qquad B = \{n : 0^{n\frown}1 \notin T_P\}.$$

□

**Corollary 17.3.9.** *For any $\Pi^0_1$ class $P$, if $P$ is almost homogeneous with modulus $n$, then $P$ is the disjoint union of $2^n$ c.e. separating classes.*

*Proof.* Given $P \in \mathbf{AH}$ with modulus $n$, for each sequence $\sigma$ of length $n$, let $P[\sigma] := \{X \in P : \sigma \prec X\}$. Each $P[\sigma]$ is homogeneous, so is a c.e. separating class, and clearly $P$ is the disjoint union of the $P[\sigma]$. □

**Proposition 17.3.10.** *For any $\Pi^0_1$ classes $P$ and $Q$, if $P$ and $Q$ are almost homogeneous, then also $P \oplus Q$ and $P \otimes Q$ are almost homogeneous.*

*Proof.* If $P$ and $Q$ are almost homogeneous with moduli $m$ and $n$, respectively, then easily $P \oplus Q$ is almost homogeneous with modulus $\max\{m, n\} + 1$ and $P \otimes Q$ is almost homogeneous with modulus $2\max\{m, n\}$. □

**Theorem 17.3.11.** **AH** *is the smallest sublattice of $\mathcal{P}_M$ which includes the family of c.e. separating degrees.*

*Proof.* By the preceding two propositions, **AH** is a sublattice of $\mathcal{P}_M$ which includes the family of c.e. separating degrees. Let $L$ be any other such lattice; we prove by induction that for all $n$,

$$P \text{ is almost homogeneous with modulus } n \quad\Longrightarrow\quad \mathbf{dg}_M(P) \in L.$$

For $n = 0$ this is true by Proposition 17.3.8, so assume as induction hypothesis that it holds for $n$ and that $P$ is almost homogeneous with modulus $n+1$. Then if for $i < 2$ we set $P_i := \{X : (i)X \in P\}$, $P_i$ is almost homogeneous with modulus $n$, so $\mathbf{dg}_M(P_i) \in L$ and clearly $P = P_0 \oplus P_1$ so also $\mathbf{dg}_M(P) \in L$. □

Of particular interest are the generalizations

$$DNC_k = \{X \in k^{\mathbb{N}} : (\forall n) X(n) \neq \phi_n(n)\}.$$

The next theorem is due to Jockusch [85]. We will use the following lemma.

**Lemma 17.3.12.** *(Cenzer-Hinman) For any $l < k$, any $s > 0$ and any function $F : k^s \to l$, there exists $j < l$ and a tree $T \subseteq k^{\leq s}$ such that*

*(i) for all $\sigma \in T$, there exist $i_0 \neq i_1 < k$ such that $\sigma^\frown i_t \in T$ for $i = 0, 1$;*

*(ii) for all $\tau \in T \cap k^s$, $F(\tau) = j$.*

*Proof.* The proof is by induction on $s$. For $s = 1$, this is just the pigeonhole principle. Now given $F : k^{s+1} \to l$, define $G : k^s \to l$ by

$$G(\tau) = (\text{least } j < l)[(\exists i_0 < i_1) F(\tau^\frown i_0) = F(\tau^\frown i_1) = j];$$

such a $j$ must exist for each $\tau$ by considering the map $F_\tau : k \to l$ defined by $F_\tau(i) = F(\tau^\frown i)$.

Now by induction there exists $j < l$ and a tree $T_G \subseteq k^{\leq s}$ satisfying (i) and (ii) above with respect to $G$. Let

$$T = T_G \cup \{\tau^\frown i : \tau \in T_G \cap k^s \;\&\; F(\tau^\frown i) = j\}.$$

It is easy to check that $T$ satisfies conditions (i) and (ii) with respect to $F$. □

**Theorem 17.3.13.** *For all $n > 1$, $DNC_{k+1} <_M DNC_k$.*

*Proof.* Note here that in general $\phi_e : \mathbb{N} \to \mathbb{N}$ and we let $\phi_e^k(n) = max\{k-1, \phi_e(n)\}$ to get the $e$th function in $k^{\mathbb{N}}$. $DNC_{k+1} \leq_M DNC_k$ by the the map $\Phi(X) = X$. Now suppose by way of contradiction that $DNC_k \leq_M DNC_{k+1}$ and let $\Phi : DNC_{k+1} \to DNC_k$. We will show how to use $\Phi$ to compute an element $Y$ of $DNC_k$, which is the contradiction.

Given $n$, we can compute $Y(n)$ as follows. First compute a level $s$ such that $\Phi(\sigma, n)\downarrow$ for all $\sigma \in (k+1)^s$ and consider the map $F : (k+1)^s \to k$ defined by $F(\sigma) = \Phi(\sigma, n)$. By Lemma 17.3.12, there exits $j_n < k$ and a tree $T \subseteq (k+1)^{\leq s}$ such that $F(\tau) = j_n$ for all $\tau \in T \cap (k+1)^s$ and such that any $\sigma \in T \cap (k+1)^{<s}$ has at least two extensions in $T$. We now show that there is in fact some $\tau \in T \cap (k+1)^s$ such that $I(\tau) \cap DNC_{k+1} \neq \emptyset$ and hence some $X \in DNC_{k+1}$ such that $\Phi(X)(n) = \Phi(\sigma, n) = j_n$. Since $\Phi(X) \in DNC_k$, it will follow that $\phi_n(n) \neq j_n$ and hence we can compute $Y \in DNC_k$ by taking $Y(n) = j_n$ for each $n$.

The path $\tau = (e_0, e_1, \ldots, e_{s-1} \in T \cap (k+1)^s$ exists by the following (although we cannot directly compute it). For $t = 0$, we have $(i_0) \neq (i_1)$ in $T$ and at least one of these does not equal $\phi_0(0)$; let $e_0$ be the least such. Given $\sigma = (e_0, \ldots, e_{t-1}) \in T$, again there exist $i_0 < i_1$ such that both $\sigma^\frown i_0$ and $\sigma^\frown i_0$ are in $T$ and again at least on of $i_0, i_1$ is not equal to $\phi_t(t)$ so we can choose $e_0$ to be the least such. $\square$

**Exercises**

17.3.1. Say that c. e. sets $A$ and $B$ are *weakly effectively inseparable* if there is a computable function $F$, mapping $\omega^2$ into the family of finite sets of natural numbers, such that, for any $x$ and $y$, if $A \subset W_x$ and $B \subset W_y$ and $W_x \cap W_y = \emptyset$, then $F(x, y)$ contains at least one element which is not in $W_x \cup W_y$. Show that if $S(A, B)$ is productive, then $A$ and $B$ are weakly effectively inseparable.

17.3.2. Recall from Section 2.2.9 the class $CC(\mathcal{T})$ of complete consistent extensions of a c. e. propositional theory $\mathcal{T}$. Show that for any c. e. theory $\mathcal{U}$, $CC(\mathcal{U})$ is Medvedev complete if and only if, for every c. e. theory $\mathcal{T}$, there exists a computable function $\Phi : CC(\mathcal{U}) \to CC(\mathcal{T})$.

17.3.3. Let the $e$th c. e. theory $\mathcal{T}_e = \{\gamma_i : i \in W_e\}$, where $\gamma_i : i \in \mathbb{N}\}$ enumerates the sentences of propositional logic. Let us say that a theory $\mathcal{T}$ is *effectively incompletable* if there exists a computable mapping $\theta : \mathbb{N} \to Sent$ such that for all $a$, if $\mathcal{T} \subseteq T_a$ and $T_a$ is consistent, then both $W_a \cup \{\theta(a)\}$ and $W_a \cup \{\neg\theta(a)\}$are consistent. Show that, for any c. e. theory $\mathcal{U}$, $\mathcal{U}$ is effectively incompletable if and only if $CC(\mathcal{U})$ is productive and hence $CC(\mathcal{U})$ is Medvedev complete if and only if $CC(\mathcal{U})$ is effectively incompletable.

17.3.4. For a pair $A, B$ of disjoint c. e. sets, let $S(A, B)$ represent the logical theory $\mathcal{U}(A, B)$ with axioms $\{A_i : i \in A\} \cup \{\neg A_i : i \in B\}$. Show that for any pair $A, B$ of effectively inseparable c. e. sets, $\mathcal{U}(A, B)$ is effectively incompletable. Hint: There exist computable functions $f$ and $g$ such that $W_{f(e)} = \{i : A_i \in \mathcal{T}_e\}$and $W_{g(e)} = \{i : \neg A_i \in \mathcal{T}_e\}$.

17.3.5. Let $\mathcal{U}$ be an effectively incompletable c. e. theory. Then for any c. e. theory $\mathcal{T}$, there exists a computable mapping $\Theta : Sent^3 \to Sent$ such that if both $\mathcal{T} \cup \{\phi\}$ and $\mathcal{U} \cup \{\psi\}$ are consistent, then

(a) $\mathcal{T} \cup \{\phi, \chi\}$ is consistent $\longleftarrow \mathcal{U} \cup \{\psi, \theta(\phi, \psi, \chi)\}$ is consistent;

(b) $\mathcal{T} \cup \{\phi, \neg\chi\}$ is consistent $\longleftarrow \mathcal{U} \cup \{\psi, \neg\theta(\phi, \psi, \chi)\}$ is consistent.

Hint: Use the Recursion Theorem to compute an index $a$ from $\phi, \psi, \chi$ such that $\mathcal{T}_a$ equals $Con(\mathcal{U} \cup \{\psi, \theta_a\})$ if $\mathcal{T} \cup \{\phi, \chi\}$ is consistent, equals $Con(\mathcal{U} \cup \{\psi, \neg\theta_a\})$ if $\mathcal{T} \cup \{\phi, \neg\chi\}$ is consistent, and equals $Con(\mathcal{U} \cup \{\psi\})$ otherwise.

17.3.6. Show directly that any effectively incompletable c. e. theory $\mathcal{U}$ is Medvedev complete. Hint: given any c. e. theory $\mathcal{T}$, recursively define a mapping $\psi : \{0,1\}^* \to Sent$ taking $\sigma$ to $\psi_\sigma$ by $\psi(\emptyset) = p_0 \vee \neg p_0$ and for any $\sigma$ of length $n$, $\psi(\sigma^\frown 0) = \psi_\sigma \wedge \neg\Theta(q_\sigma, \psi_\sigma, A_n)$ and $\psi(\sigma^\frown 1) = \psi_\sigma \wedge \Theta(q_\sigma, \psi_\sigma, A_n)$, where $q_\sigma$ denotes the conjunction over $i < n$ of $\{A_i : \sigma(i) = 1\} \cup \{\neg A_i : \sigma(i) = 0\}$. Then $\mathcal{T} \cup \{q_\sigma$ is consistent if and only if $\mathcal{U} \cup \{\psi(\sigma)\}$ is consistent and for $X \in CC(\mathcal{U})$, we may define $\Phi(X) \in CC(\mathcal{T})$ to be the unique $Y$ such that $X \in CC(\mathcal{U} \cup \{\psi(Y\lceil n)\}$ for all $n$.

## 17.4 Measure

In this section we give the result of Cenzer and Hinman [23] that there is no Medvedev complete $\Pi^0_1$ class of positive measure and indeed that no class of positive measure can be $\leq_M$ any separating class. The following lemma is due to Simpson [178].

**Lemma 17.4.1.** *(Simpson) Let $\{F_n\}_{n\in\omega}$ be a sequence of nonempty finite subsets of $\mathbb{N}$ of bounded cardinality and let $S = \prod_n F_n$. Let $P \subseteq \{0,1\}^{\mathbb{N}}$ have positive measure and let $Q \subseteq \mathbb{N}^{\mathbb{N}}$ be nonempty.*

1. *If $S \leq_M P \otimes Q$, then $S \leq_M Q$.*

2. *If $S \leq_w P \otimes Q$, then $S \leq_w Q$.*

*Proof.* (1) The proof is similar to that of Theorem 3.3.11. Suppose that $card(F_n) < k$ for all $n$. Let $U$ and $V$ be clopen so that $\mu(V - P) < \mu(P)$ and $\mu(V - U)$ are both $< \mu(P)/4k$ and therefore $\mu(U - P) < \mu(U)/K$. It is important here that $\mu(U)$ is rational. Let $\Phi$ be a computable function such that $\Phi(x \oplus y) \in S$ for all $x \in P$ and $y \in Q$. Given $y \in Q$ and $n \in \mathbb{N}$, we can compute $m = \Psi(y)(n)$ such that $\mu(\{x \in U : \Phi(x \oplus y)(n) = m\}) > \mu(U)/k$ and therefore $m \in F_n$. Thus $\Psi$ maps $Q$ onto $S$ and hence $S \leq_M Q$.

(2) Fix $y \in Q$ and note that $S \leq_w P \otimes \{y\}$. Now for each $x \in P$, there is a function $\Phi_{e(x)}$ such that $\Phi_{e(x)}(x, y) \in S$. By countable additivity of $\mu$, there is a single function $\Phi$ and a subset $P_g$ of $P$ of positive measure such that $S \leq_M P_g \otimes \{y\}$ Now by lemma 17.4.1, $S \leq_M \{g\}$. It follows that $S \leq_s Q$. □

Note that in particular these lemmas apply to any separating class $S$.

**Theorem 17.4.2.** *(Simpson) Let $P, Q$ be nonempty $\Pi^0_1$ subsets of $\{0,1\}^{\mathbb{N}}$ and suppose that $P$ has positive measure. If $Q <_M \mathbf{1}$, then $P \otimes Q <_M 1$ and similarly if $Q <_w \mathbf{1}$, then $P \otimes Q <_w \mathbf{1}$.*

*Proof.* This follows from Lemma 17.4.1, since there is a Medvedev complete separating class by Corollary 17.3.3. □

**Corollary 17.4.3.** *If $P \subset \{0,1\}^{\mathbb{N}}$ is a $\Pi^0_1$ class with positive measure, then $P <_w \mathbf{1}$.*

*Proof.* This follows from Theorem 17.4.2 by letting $Q = \{0,1\}^{\mathbb{N}}$. □

**Exercises**

17.4.1. Show that for any $\Pi^0_1$ classes $P$ and $Q$, if $P$ has positive measure and $DNC_k \leq_M P \otimes Q$, then $DNC_k \leq_M Q$. Conclude that $P \otimes DNC_k >_M P \otimes DNC_{k+1}$ for all $k$.

## 17.5 Randomness

Recall the notion of 1-randomness from Section 4.3. Let $R$ be the class of all 1-random reals in $\{0,1\}^{\mathbb{N}}$. It follows from Theorem 4.3.3.5 that there is a $\Pi^0_1$ class $P$ with positive measure such that $P \subset R$.

**Theorem 17.5.1.** *(Simpson) For any $\Pi^0_1$ class $P \subset R$ with positive measure, $P \equiv_w R$.*

*Proof.* Since $P \subset R$, it follows that $R \leq_w P$. On the other hand, Theorem 4.3.3.6 tells us that $P$ has elements of every 1-random degree, so for any $x \in R$, there exists $y \in P$ with $y \equiv_T x$ and hence $P \leq_w R$. □

**Corollary 17.5.2.** *If a $\Pi^0_1$ class $P$ contains a random real, then $P \equiv_w R$.* □

**Corollary 17.5.3.** *(Simpson) The Muchnik degree of $R$ can be characterized as the unique largest Muchnik degree of any $\Pi^0_1$ class $P \subseteq \{0,1\}^{\mathbb{N}}$ such that $\mu(P) > 0$.*

*Proof.* By Theorem 17.5.1, there is a $\Pi^0_1$ class $P \subseteq \{0,1\}^{\mathbb{N}}$ with positive measure and with $P \equiv_w R$. Now let $P$ be any $\Pi^0_1$ class $P$ of positive measure. It follows from Theorem 4.3.3.6 that $P \leq_w R$. □

**Exercises**

17.5.1.

## 17.6 Thin Classes

In this section, we prove the following theorem of Simpson.

**Theorem 17.6.1.** *If $Q \subseteq \{0,1\}^{\mathbb{N}}$ is a nonempty perfect thin $\Pi^0_1$ class and $R \subset \{0,1\}^{\mathbb{N}}$ is the set of all Martin-Löf random reals, then $Q$ and $R$ and Muchnik incomparable.*

The theorem is proved using a sequence of lemmas.

**Lemma 17.6.2.** *Let $Q \subset \{0,1\}^{\mathbb{N}}$ be nonempty thin $\Pi^0_1$ class, let $x$ be Martin-Löf random, and let $y \in Q$ be almost computable. Then $x$ is not Turing reducible to $y$.*

*Proof.* Suppose by way of contradiction that $x \leq_T y$. Then by Theorem 1.9.13 $x \leq_{tt} y$. Now by Theorem 1.9.10, there is a total computable functional $\Phi : \{0,1\}^{\mathbb{N}} \to \{0,1\}^{\mathbb{N}}$ mapping $y$ to $x$. Now by Lemma 2.8.4, the image $\Phi[Q]$ is a thin $\Pi^0_1$ class and hence has measure zero by Theorem 3.3.3 But $x \in \Phi[Q]$ is Martin-Löf random and hence $\mu(\Phi[Q]) > 0$ by Exercise 5. □

Since every nonempty $\Pi^0_1$ class contains an almost computable member by Theorem 3.1.5, it follows that $R$ is not Muchnik reducible to $Q$. The following lemma is due to Demuth [52].

**Lemma 17.6.3.** *Let $x \in \{0,1\}^{\mathbb{N}}$ be Martin-Löf random and let $y \leq_{tt} x$ be noncomputable. Then there exists $z \equiv_T y$ such that $z$ is Martin-Löf random.*

**Lemma 17.6.4.** *If $Q \subseteq \{0,1\}^{\mathbb{N}}$ is a nonempty perfect thin $\Pi^0_1$ class, let $y \in Q$ and let $x$ be Martin-Löf random and almost computable. Then $y$ is not Turing reducible to $x$.*

*Proof.* Suppose by way of contradiction that $y \in Q$ and $y \leq_T x$. Then $y \leq_{tt} x$ since $x$ is almost computable. It follows from Lemma 17.6.3 that there is a random $z \equiv_T y$. But then $z \leq_T x$, contradicting Lemma 17.6.2. □

To complete the proof of Theorem 17.6.1, let $x$ be random and almost computable and let $Q$ be a nonempty perfect thin $\Pi^0_1$ class. It follows from Lemma 17.6.4 that no member of $Q$ is computable from $x$. □

**Corollary 17.6.5.** *There is a $\Pi^0_1$ separating class $Q$ and a $\Pi^0_1$ class $Q'$ such that $Q < wQ'$, and furthermore, for any separating class $P$, if $P$ is Muchnik reducible to $Q'$, then $P$ is Muchnik reducible to $Q$.*

*Proof.* Let $Q$ be a perfect thin $\Pi^0_1$ class which is separating 2.8.1, let $R$ be a $\Pi^0_1$ class of randoms and let $Q' = Q \otimes R$. It follows from Theorem 17.6.1 that $Q'$ is not Muchnik reducible to $Q$. The furthermore remark follows from Lemma 17.4.1. □

# Chapter 18

# Random Closed Sets

The study of algorithmic randomness has been of great interest in recent years. The basic problem is to quantify the randomness of a single real number; here we will extend this problem to the randomness of the set of paths through a finitely-branching tree. Early in the last century, von Mises [197] suggested that a random real should obey reasonable statistical tests, such as having a roughly equal number of zeroes and ones of the first $n$ bits, in the limit. Thus a random real would be *stochastic* in modern parlance. If one considers only *computable* tests, then there are countably many and one can construct a real satisfying all tests.

An early approach to randomness was through betting. Effective betting on a random sequence should not allow one's capital to grow unboundedly. The betting strategies used are constructive martingales, introduced by Ville [196] and implicit in the work of Levy [118], which represent fair double-or-nothing gambling.

Martin-Löf [133] observed that stochastic properties could be viewed as special kinds of meaure zero sets and defined a random real as one which avoids certain effectively presented measure 0 sets. That is, a real $x \in \{0,1\}^{\mathbb{N}}$ is Martin-Löf random if for any effective sequence $S_1, S_2, \dots$ of c.e. open sets with $\mu(S_n) \leq 2^{-n}$, $x \notin \cap_n S_n$.

At the same time Kolmogorov [103] defined a notion of randomness for finite strings based on the concept of *incompressibility.* For infinite words, the stronger notion of prefix-free complexity developed by Levin [117], Gács [69] and Chaitin [43] is needed. Schnorr later proved [170] that the notions of constructive martingale randomness, Martin-Löf randomness, and prefix-free randomness are equivalent. In this chapter, we will consider algorithmic randomness on the space $\mathcal{C}$ of nonempty closed subsets $P$ of $\{0,1\}^{\mathbb{N}}$.

The betting approach to randomness is formalized as follows:

**Definition 18.0.6** (Ville [196]). *1. A* martingale *is a function* $d : n^{<\mathbb{N}} \to$

$[0,\infty)$ *such that for all* $\sigma \in n^{<\mathbb{N}}$,

$$d(\sigma) = \frac{1}{n}\sum_{i=0}^{n-1} d(\sigma^\frown i).$$

2. *A martingale* $d$ succeeds *on* $X \in n^{\mathbb{N}}$ *if*

$$\limsup_{m\to\infty} d(X\lceil m) = \infty.$$

*That is, the betting strategy results in an unbounded amount of money made on the binary string* $X$*.*

3. *The* success set *of* $d$ *is the set* $S^\infty[d]$ *of all sequences on which* $d$ *succeeds.*

That is, a martingale on $2^{<\mathbb{N}}$ is the representation of a fair double-or-nothing betting strategy. When working on $3^{<\mathbb{N}}$ the strategy is triple-or-nothing.

**Definition 18.0.7.** *A martingale* $d$ *is* constructive (effective, c.e.) *if it is* lower semi-computable*; that is, if there is a computable function* $\hat{d} : n^{<\mathbb{N}} \times \mathbb{N} \to \mathbb{Q}$ *such that*

1. *for all* $\sigma$ *and* $t$, $\hat{d}(\sigma,t) \leq \hat{d}(\sigma,t+1) < d(\sigma)$, *and*

2. *for all* $\sigma$, $\lim_{t\to\infty} \hat{d}(\sigma,t) = d(\sigma)$.

In other words, $d(w)$ is approximated from below by rationals uniformly in $w$. A sequence in $2^{\mathbb{N}}$ is considered random in this setting if no constructive martingale succeeds on it.

Martin-Löf randomness for reals, as defined above, is extended to closed sets by giving an effective homeomorphism with the space $\{0,1,2\}^{\mathbb{N}}$ and simply carrying over the notion of randomness from that space.

Prefix-free randomness for reals is defined as follows. Let $M$ be a prefix-free function with domain $\subset \{0,1\}^*$; that is, if $\sigma \sqsubseteq \tau$ are strings in the domain of $M$, then $\sigma$ must equal $\tau$. For any finite string $\tau$, let $K_M(\tau) = \min\{|\sigma|, \infty : M(\sigma) = \tau\}$. There is a *universal* prefix-free function $U$ such that, for any prefix-free $M$, there is a constant $c$ such that for all $\tau$

$$K_U(\tau) \leq K_M(\tau) + c.$$

We let $K(\sigma) = K_U(\sigma)$. Then $x$ is called *prefix-free random* if there is a constant $c$ such that $K(x\lceil n) \geq n - c$ for all $n$. This means that the initial segments of $x$ are not *compressible*.

For a tree $T$, we want to consider the compressibility of $T_n = T \cap \{0,1\}^n$. This has a natural representation of length $2^n$ since there are $2^n$ possible nodes of length $n$. We will show that any tree $T_P$ can be compressed, that is, $K(T_n) \geq 2^n - c$ is impossible for a tree with no dead ends.

## 18.1 Martin-Löf Randomness of Closed Sets

In this section, we define a measure on the space $\mathcal{C}$ of nonempty closed subsets of $\{0,1\}^{\mathbb{N}}$ and use this to define the notion of randomness for closed sets. We then obtain several properties of random closed sets.

An effective one-to-one correspondence between the space $\mathcal{C}$ and the space $3^{\mathbb{N}}$ is defined as follows. Let a closed set $Q$ be given and let $T = T_Q$ be the tree without dead ends such that $Q = [T]$.

Then define the code $x = x_Q \in \{0,1,2\}^{\mathbb{N}}$ for $Q$ as follows. Let $\emptyset = \sigma_0, \sigma_1, \sigma_2, \ldots$ enumerate the elements of $T$ in order, first by length and then lexicographically. We now define $x = x_Q = x_T$ by recursion as follows. For each $n$, $x(n) = 2$ if $\sigma_n{}^\frown 0$ and $\sigma_n{}^\frown 1$ are both in $T$, $x(n) = 1$ if $\sigma_n{}^\frown 0 \notin T$ and $\sigma_n{}^\frown 1 \in T$ and $x(n) = 0$ if $\sigma_n{}^\frown 0 \in T$ and $\sigma_n{}^\frown 1 \notin T$.

Now define the measure $\mu^*$ on $\mathcal{C}$ by

$$\mu^*(\mathcal{X}) = \mu(\{x_Q : Q \in \mathcal{X}\}).$$

Informally this means that given $\sigma \in T_Q$, there is probability $\frac{1}{3}$ that both $\sigma^\frown 0 \in T_Q$ and $\sigma^\frown 1 \in T_Q$ and, for $i = 0,1$, there is probability $\frac{1}{3}$ that only $\sigma^\frown i \in T_Q$. In particular, this means that $Q \cap I(\sigma) \neq \emptyset$ implies that for $i = 0,1$, $Q \cap I(\sigma^\frown i) \neq \emptyset$ with probability $\frac{2}{3}$.

Let us comment briefly on why some other natural representations were rejected. Suppose first that we simply enumerate all strings in $\{0,1\}^*$ as $\sigma_0, \sigma_1, \ldots$ and then represent $T$ by its characteristic function so that $x_T(n) = 1 \iff \sigma_n \in T$. Then in general a code $x$ might not represent a tree. That is, once we have $(01) \notin T$ we cannot later decide that $(011) \in T$. Suppose then that we allow the empty closed set by using codes $x \in \{0,1,2,3\}^*$ and modify our original definition as follows. Let $x(n) = i$ have the same definition as above for $i \leq 2$ but let $x(n) = 3$ mean that neither $\sigma_n{}^\frown 0$ nor $\sigma^\frown 1$ is in $T$. Informally, this would mean that for $i = 0,1$, $\sigma \in T$ implies that $\sigma^\frown i \in T$ with probability $\frac{1}{2}$. The advantage here is that we can now represent all trees. But this is also a disadvantage, since for a given closed set $P$, there are many different trees $T$ with $P = [T]$. The second problem with this approach is that we would have $[T] = \emptyset$ with positive probability. We briefly return to this subject in Section 6.

Now we will say that a closed set $Q$ is (Martin-Löf) random if the code $x_Q$ is Martin-Löf random. Let $Q_x$ denote the unique closed set $Q$ such that $x_Q = x$. Since random reals exist, it follows that random closed sets exists. Furthermore, there are $\Delta^0_2$ random reals, so we have the following.

**Theorem 18.1.1.** *There exists a random closed set $Q$ such that $T_Q$ is $\Delta^0_2$.* $\square$

Note that if $T_Q$ is $\Delta^0_2$, then $Q$ must contain $\Delta^0_2$ elements (Theorem 14.1.6). Since there exist strong $\Pi^0_2$ classes with no $\Delta^0_2$ elements, there are strong $\Pi^0_2$ classes $Q$ such that $T_Q$ is not $\Delta^0_2$.

The following lemma will be needed throughout.

**Lemma 18.1.2.** *For any $Q \subseteq 2^{\mathbb{N}}$ which is either closed or open, $\mu^*(\{P : P \subseteq Q\}) \leq \mu(Q)$.*

*Proof.* Let $\mathcal{P}_C(Q)$ denote $\{P : P \subseteq Q\}$. We first prove the result for (nonempty) clopen sets $U$ by the following induction. Suppose $U = \cup_{\sigma \in S} I(\sigma)$, where $S \subseteq \{0,1\}^n$. For $n = 1$, either $\mu(U) = 1 = \mu^*(\mathcal{P}_C(U))$ or $\mu(U) = \frac{1}{2}$ and $\mu^*(\mathcal{P}_C(Q)) = \frac{1}{3}$. For the induction step, let $S_i = \{\sigma : i^\frown\sigma \in S\}$, let $U_i = \cup_{\sigma \in S_i} I(\sigma)$, let $m_i = \mu(U_i)$ and let $v_i = \mu^*(\mathcal{P}_C(U_i))$, for $i = 0, 1$. Then considering the three cases in which $S$ includes both initial branches or just one, we calculate that

$$\mu^*(\mathcal{P}_C(U)) = \frac{1}{3}(v_0 + v_1 + v_0 v_1).$$

Thus by induction we have

$$\mu^*(\mathcal{P}_C(U)) \leq \frac{1}{3}(m_0 + m_1 + m_0 m_1).$$

Now

$$2m_0 m_1 \leq m_0^2 + m_1^2 \leq m_0 + m_1,$$

and therefore

$$\mu^*(\mathcal{P}_C(U)) \leq \frac{1}{3}(m_0 + m_1 + m_0 m_1) \leq \frac{1}{2}(m_0 + m_1) = \mu(U).$$

For a closed set $Q$, let $Q = \cap_n U_n$, with $U_{n+1} \subseteq U_n$ for all $n$. Then $P \subset Q$ if and only if $P \subseteq U_n$ for all $n$. Thus

$$\mathcal{P}_C(Q) = \cap_n \mathcal{P}_C(U_n),$$

so that

$$\mu^*(\mathcal{P}_C(Q)) = \lim_{n\to\infty} \mu^*(\mathcal{P}_C(U_n)) \leq \lim_{n\to\infty} \mu(U_n) = \mu(Q).$$

Finally, for an open set $Q$, let $Q = \bigcup_n U_n$ be the union of an increasing sequence of clopen sets. Then, by compactness,

$$\mathcal{P}_C(Q) = \cup_n \mathcal{P}_C(U_n),$$

so that

$$\mu^*(\mathcal{P}_C(Q)) = \lim_{n\to\infty} \mu^*(\mathcal{P}_C(U_n)) \leq \lim_{n\to\infty} \mu(U_n) = \mu(Q).$$

This completes the proof of the lemma. □

Next we consider the intersection of a random closed set with an interval $I(\sigma)$ and the disjoint union of random closed sets.

Let us call the coding of a closed set $Q$ by the nodes of its representative tree with no dead ends the *canonical code* of $Q$. We wish now to introduce a second method of coding, the *ghost code*. A ghost code of $Q$ is an infinite ternary string whose bits correspond to all nodes of $2^{<\mathbb{N}}$ in lexicographical order. The bits corresponding to the nodes of $Q$'s tree (the "canonical nodes") hold the same values as the corresponding bits in the canonical code; the remaining "ghost nodes" may hold any values. Ghost codes are non-unique, and every closed set has a non-random ghost code (if the closed set itself is random take the code with ghost nodes all equal to zero).

We define randomness for closed sets in the world of ghost codes as possession of a random code. This method of coding is more convenient for some purposes; for example, we will use it to show that if $Q_0, Q_1$ are closed sets and $Q = \{0^\frown x : x \in Q_0\} \cup \{1^\frown x : x \in Q_1\}$, $Q$ is random if and only if the $Q_i$ are random relative to each other. With canonical coding it is straightforward to show relative randomness of the half trees is sufficient for randomness of the full tree, but not its necessity.

However, the utility of the ghost codes rests entirely on the following correspondence.

**Theorem 18.1.3.** *The canonical code of a closed set $Q \subseteq 2^{\mathbb{N}}$ is random if and only if $Q$ has some random ghost code.*

*Proof.* ($\Leftarrow$) Suppose the canonical code of $Q$ is nonrandom. Then there is a c.e. martingale $m$ that succeeds on it. From any initial segment $\sigma$ of a ghost code $g$ for $Q$, the subsequence $\hat{\sigma}$ of exactly the canonical nodes of $\sigma$ is computable. Therefore it is computable whether the bit of $g$ after $\sigma$ is canonical or ghost. From $m$, define the martingale $m'$ which bets as follows:

$$m'(\sigma^\frown i) = \begin{cases} \frac{m(\hat{\sigma}^\frown i)}{m(\hat{\sigma})} m'(\sigma) & \text{next bit is a canonical node} \\ m'(\sigma) & \text{next bit is a ghost node.} \end{cases}$$

That is, $m'$ holds its money on ghost nodes and bets proportionally to $m$ (in fact, identically) on canonical nodes. It is clear that $m'$ succeeds on the ghost code $g$ and thus $g$ is nonrandom.

($\Rightarrow$) Now suppose the canonical code $r$ for $Q$ is random, and let $q$ be an infinite ternary string that is random relative to $r$ (so therefore $r$ is also random relative to $q$). We claim the ghost code $g$ obtained by using the bits of $r$ as the canonical nodes and the bits of $q$ in their original order as the ghost nodes is random. It is clear that $g$ is a ghost code for $Q$.

Suppose $m$ is a c.e. martingale that bets on $g$. We define two martingales from $m$, $m_r$ and $m_q$, that bet on the bits of $r$ and $q$ respectively, with oracle $q$ and $r$ respectively, according to $m$'s actions on the corresponding bits of $g$, and show that if they do not succeed, $m$ does not succeed. As $q$ and $r$ are relatively random, $m_r$ and $m_q$ cannot succeed, and so $g$ will be random.

We define $m_r$ only; for $m_q$ swap the roles of $r$ and $q$, and of canonical and ghost nodes. From a string $\sigma$ compute the unique initial segment $\tau$ of a ghost code such that

1. $\sigma$ is the substring of canonical nodes of $\tau$,
2. the node to follow $\tau$ is a canonical node, and
3. the ghost nodes of $\tau$ are an initial segment of the oracle $q$.

Then $m_r(\sigma^\frown i) = \frac{m(\tau^\frown i)}{m(\tau)} m_r(\sigma)$.

Since each of $r, q$ is random relative to the other, neither $m_r$ nor $m_q$ can succeed. That is, there is some $b$ such that $m_r(\sigma) \leq b$ for all $\sigma \subset r$ and likewise $m_q(\tau) \leq b$ for all $\tau \subset q$. We claim that the values of $m$ on initial segments of $g$ are bounded by $b^2$.

At each node of the code $r$ the martingale $m_r$ will multiply the capital held before the node by some constant factor. Let $\{c_n^r\}_{n\in\mathbb{N}}$ be the sequence of level-$n$ multiplicative factors for $m_r$. Assuming an initial capital of 1, the values $m_r$ achieves on initial segments of $r$ are

$$\prod_{k=0}^{\ell} c_k^r$$

for $0 \leq \ell < \infty$. By assumption, every product of this form is less than or equal to $b$. Substituting the corresponding definition for $m_q$, $\{c_n^q\}_{n\in\mathbb{N}}$, gives the same result.

The original martingale $m$ behaves on the subequences $r$ and $q$ exactly as the martingales $m_r$ and $m_q$ do, by construction of $m_r$ and $m_q$. Therefore it has the same collection of multiplicative factors. Again assuming initial capital 1, the values it achieves are thus products of the form

$$\left(\prod_{k=0}^{\ell} c_k^r\right) \cdot \left(\prod_{k=0}^{\ell'} c_k^q\right).$$

As each of the subproducts is bounded by $b$, the entire product is bounded by $b^2$ and $m$ does not succeed on $g$. Since $m$ was arbitrary, no c.e. martingale succeeds on $g$ and thus $Q$ has a random ghost code. □

Note that this theorem relativizes so we can assert that the canonical code is, say, $A$-random for some $A$ if and only if it has an $A$-random ghost code.

The primary purpose of the ghost codes is to remove the dependence on the particular closed set under discussion when interpreting bits of the code as nodes of the tree. This is especially useful when subdividing the tree, as in the following definition.

**Definition 18.1.4.** *The* tree join *of closed sets $P_0$ and $P_1$ is the closed set $Q = \{0^\frown x : x \in P_0\} \cup \{1^\frown x : x \in P_1\}$. Given ghost codes $r_0, r_1$ for the $P_i$, their tree join $r_0 \boxplus r_1$ is the code for $Q$ with the corresponding ghost node values.*

This is distinguished from what we will call the *recursion-theoretic join* $r_0 \oplus r_1$, where elements of $r_0$ and $r_1$ alternate.

We wish to relate the recursion-theoretic join and the tree join. First recall van Lambalgen's theorem.

**Theorem 18.1.5** (van Lambalgen [195])**.** *The following are equivalent.*

1. *$A \oplus B$ is $n$-random.*

2. *$A$ is $n$-random and $B$ is $n$-$A$-random (or vice-versa).*

3. *$A$ is $n$-$B$-random and $B$ is $n$-$A$-random.*

**Lemma 18.1.6.** *Given two ghost codes $r_0, r_1$, the tree join $r_0 \boxplus r_1$ is random if and only if the recursion theoretic join $r_0 \oplus r_1$ is random.*

*Proof.* We show a Martin-Löf test that contains one version of the join may be transformed into a test that contains the other version, and therefore if one is nonrandom, both are nonrandom. To simplify the proof we ignore the initial bit of the tree join, as it has no matching bit in the recursion-theoretic join.

It is clear that initial segments of the two versions of join may be transformed into each other via a computable algorithm that is independent of the value of the bits, provided the strings are of length $\ell_n := 2^{n+1} - 1$ (i.e., the final node is the end of a level of the tree). For strings that terminate in the middle of levels, a single initial segment $\sigma$ becomes a finite collection of strings $\tau_j$, one for each extension of $\sigma$ to a string of length $\ell_n$ for the least $n$ such that $|\sigma| \leq \ell_n$. There will be $2^{\ell_n - |\sigma|}$ such strings and hence the measure of the intervals around them will total $2^{-\ell_n} \cdot 2^{\ell_n - |\sigma|} = 2^{-|\sigma|}$, equal to the measure of the interval around $\sigma$.

Therefore, let $\{U_i : i \in \mathbb{N}\}$ be a Martin-Löf test failed by $r_0 \oplus r_1$. When $\sigma$ enters $U_i$ for some $i$, enumerate the finite collection of $\tau_j$ as described above into a new set $\hat{U}_i$. The collection $\{\hat{U}_i : i \in \mathbb{N}\}$ is clearly a Martin-Löf test, as it is enumerated simultaneously with the $\{U_i\}$ and $\mu(\hat{U}_i) = \mu(U_i)$ for all $i \in \mathbb{N}$. If $r_0 \oplus r_1$ extends $\sigma \in U_i$, $r_0 \boxplus r_1$ will extend one of the corresponding $\tau_j$ in $\hat{U}_i$. Therefore since $r_0 \oplus r_1$ fails $\{U_i\}$, $r_0 \boxplus r_1$ will fail $\{\hat{U}_i\}$. Symmetrically, if $r_0 \boxplus r_1$ is nonrandom we may construct a Martin-Löf test that $r_0 \oplus r_1$ also fails. □

We now obtain the following corollary of Theorems 18.1.3 and 18.1.5, and Lemma 18.1.6.

**Corollary 18.1.7.** *Suppose $P_i$, $i = 0, 1$, are closed sets with canonical codes $r_i$ and let $P$ be the tree join of $P_0, P_1$. Then $P$ is random if and only if $r_0 \oplus r_1$ is random.*

*Proof.* ($\Leftarrow$) Suppose that $r_0 \oplus r_1$ is random. Then by Theorem 18.1.5, the $r_i$ are mutually relatively random. By Theorem 18.1.3, there are ghost codes $g_i$ for the $P_i$ that are also mutually relatively random. Again by 18.1.5, the recursion-theoretic join $g_0 \oplus g_1$ is random; then by Theorem 18.1.6 the tree join $g_0 \boxplus g_1$ is also random, and hence $P$ possesses a random ghost code and is random.
($\Rightarrow$) Suppose now that $P$ is random, and therefore possesses a random ghost code $g$. The code $g$ may be thought of as a tree join $g_0 \boxplus g_1$, which is therefore random, and so by Theorem 18.1.6, $g_0 \oplus g_1$ is random. By Theorem 18.1.5, the individual codes $g_0, g_1$ are therefore mutually relatively random, and so by Theorem 18.1.3 the canonical codes $r_0, r_1$ for the half trees are as well. Thus again by 18.1.5, $r_0 \oplus r_1$ is random. □

## 18.2 Members of Random Closed Sets

# Bibliography


[1] T. . A. Lewis.

[2] K. Appel, W. Haken, and J. Koch, *Every planar map is 4-colorable*, Ill. J. Math., 41 (1976), pp. 469–480.

[3] K. Apt, *Logic programming*, in Handbook of Theoretical Computer Science (J. van Leeuven,ed.), MIT Press, 1990, pp. 493–574.

[4] E. Artin and O. Schreier, *Algebraische konstruktion reeller körper*, Abh. Math. Sem. Univ. Hamburg, 5 (1927), pp. 85–89.

[5] G. Barmpalias, A. Lewis, and K. Ng, *The importance of* $\pi_1^0$ *classes in effective randomness*, Journal of Symbolic Logic, 75 (2010), p. 387400.

[6] G. Barmpalias, A. Lewis, and F. Stephan, $\pi_1^0$ *classes, lr degrees and turing degrees*, Annals of Pure and Applied Logic, 156 (2008), p. 2138.

[7] D. Bean, *Effective coloration*, Journal of Symbolic Logic, 41 (1976), pp. 469–480.

[8] ——, *Recursive euler and hamiltonian paths*, Proceedings of the American Mathematical Society, 55 (1976), pp. 885–894.

[9] L. Blum, M. Shub, and S. Smale, *On a theory of computation and complexity over the real numbers*, Bulletin of the American Mathematical Society, 21 (1989), pp. 1–46.

[10] V. Brattka and K. Weihrauch, *Computability on subsets of euclidean space i: closed and compact subsets*, Theoretical Computer Science, 219 (1999), pp. 65–93.

[11] M. Braverman and S. Cook, *Computing over the reals: foundations for scientific computing*, Notices Amer. Math. Soc., 53 (2006), p. 318329.

[12] M. Braverman and M. Yampolsky, *Non-computable julia sets*, J. Amer. Math. Soc., 19 (2006), p. 551578.

[13] R. Brooks, *On colouring the nodes of a network*, Proc. Cambridge Philos. Soc., 37 (1941), pp. 194–197.

[14] J. BÜCHI, *On a decision method in restricted second order arithmetic*, Stanford Univ. Press, 1962, pp. 1–11.

[15] S. BURRIS AND H. SANKAPPANAVAR, *Lattice-theoretic decision problems in universal algebras*, Algebra Universalis, 5 (1975), pp. 163–177.

[16] H.-G. CARSTENS, *A 5-color extension theorem*, in Computation Theory and Logic, vol. 270 of Lecture Notes in Computer Science, Springer-Verlag, 1987, pp. 67–77.

[17] D. CENZER, *Effective dynamics*, in Logical Methods in honor of Anil Nerode's Sixtieth Birthday, J. Crossley, J. B. Remmel, R. Shore, and M. Sweedler, eds., Birkhäuser, 1993, pp. 162–177.

[18] D. CENZER AND P. BRODHEAD, *Effectively closed sets and enumerations*, Archive for Mathematical Logic, 45 (2008), pp. 565–582.

[19] D. CENZER, P. CLOTE, R. SMITH, R. SOARE, AND S. WAINER, *Members of countable $\pi_1^0$ classes*, Annals of Pure and Applied Logic, 31 (1986), pp. 145–163.

[20] D. CENZER, A. DASHTI, AND J. L. F. KING, *Countable symbolic dynamics*, Mathematical Logic Quarterly, 54 (2008), pp. 524–533.

[21] D. CENZER, A. DASHTI, F. TOSKA, AND S. WYMAN, *Computability of countable subshifts in one dimension*, Theory of Computing Systems, 51 (2012), pp. 352–371.

[22] D. CENZER, R. DOWNEY, C. G. JOCKUSCH, AND R. SHORE, *Countable thin $\pi_1^0$ classes*, Annals of Pure and Applied Logic, 59 (1993), pp. 79–139.

[23] D. CENZER AND P. G. HINMAN, *Density of the medvedev lattice of $\pi_1^0$ classes*, Archive for Mathematical Logic, 42 (2003), pp. 583–600.

[24] D. CENZER AND E. HOWORKA, *On vertex partitions of certain infinite graphs*, Discrete Mathematics, 23 (1978), pp. 105–113.

[25] D. CENZER, W. MAREK, AND J. REMMEL, *Index sets for finite predicate logic programs*, in FLOC '99 Workshop on Complexity-theoretic and Recursion-theoretic methods in Databases, Artificial Intelligence and Finite Model Theory, T. Eiter and G. Gottlob, eds., Technische Universitat Wien, 1999, pp. 72–80.

[26] D. CENZER AND R. D. MAULDIN, *On the borel class of the derived set operator*, Bulletin Societe Math. France, 110 (1982), pp. 357–380.

[27] ——, *On the borel class of the derived set operator ii*, Bulletin Societe Math. France, 111 (1983), pp. 367–372.

[28] D. CENZER AND A. NIES, *Initial segments of the lattice of $\Pi_1^0$ classes*, Journal of Symbolic Logic, 66 (2001), pp. 1749–1765.

[29] ———, *Global properties of the lattice of $\Pi^0_1$ classes*, Proceedings of the American Mathematical Society, 132 (2004), pp. 239–249.

[30] D. Cenzer and J. B. Remmel, *Polynomial-time versus recursive models*, Annals of Pure and Applied Logic, 54 (1991), pp. 17–58.

[31] ———, *Polynomial time abelian groups*, Annals of Pure and Applied Logic, 56 (1992), pp. 313–363.

[32] ———, *Recursively presented games and strategies*, Math. Social Sciences, 24 (1992), pp. 117–139.

[33] ———, *Feasible graphs and colorings*, Methods of Logic in Computer Science, 41 (1995), pp. 327–352.

[34] ———, *Complexity theoretic model theory and algebra*, in Handbook of Recursive Mathematics, Y. Ersov, S. Goncharov, A. Nerode, and J. B. Remmel, eds., Studies in Logic Found. Math #138, Elsevier, 1998, pp. 381–513.

[35] ———, *Index sets for $\pi^0_1$ classes*, Annals of Pure and Applied Logic, 93 (1998), pp. 3–61.

[36] ———, *$\pi^0_1$ classes in mathematics*, in Handbook of Recursive Mathematics, Y. Ersov, S. Goncharov, A. Nerode, and J. B. Remmel, eds., Studies in Logic Found. Math #139, Elsevier, 1998, pp. 623–821.

[37] ———, *Index sets in computable analysis*, Theoretical Computer Science, 219 (1999), pp. 111–150.

[38] ———, *Effectively closed sets and graphs of computable real functions*, Theoretical Computer Science, 284 (2002), pp. 279–318.

[39] ———, *Index sets in $\omega$-languages*, Mathematical Logic Quarterly, 49 (2002), pp. 22–33.

[40] ———, *Complexity, completeness and decidability*, Journal of Symbolic Logic, 71 (2006), pp. 399–424.

[41] D. Cenzer and F. Riazati, *Minimal extensions of $\pi^0_1$ classes*, Mathematical Logic Quarterly, 51 (2005), pp. 206–216.

[42] D. Cenzer and R. Smith, *The ranked points of a $\pi^0_1$ set*, Journal of Symbolic Logic, 54 (1989), pp. 975–991.

[43] G. Chaitin, *Information-theoretical characterizations of recursive infinite strings*, Theoretical Computer Science, (1976), pp. 45–48.

[44] K. Chen, *Recursive well-founded orderings*, Annals of Pure and Applied Logic, 13 (1978), pp. 117–147.

[45] P. Cholak, R. Coles, R. Downey, and E. Herrmann, *Automorphisms of the lattice of $\pi_1^0$ classes*, Transactions of the American Mathematical Society, 353 (2001), pp. 4899–4924.

[46] P. Cholak and R. Downey, *On the cantor-bendixson rank of recursively enumerable sets*, Journal of Symbolic Logic, 58 (1993), pp. 629–640.

[47] A. Church and S. Kleene, *Formal definitions in the theory of ordinal numbers*, Fund. Math., 28 (1937), pp. 11–21.

[48] P. Clote, *On recursive trees with a unique infinite branch*, Proceedings of the American Mathematical Society, 93 (1985), pp. 335–342.

[49] P. J. Cohen, *Set Theory and the Continuum Hypothesis*, Benjamin, 1966.

[50] T. Craven, *The boolean space of orderings of a field*, Transactions of the American Mathematical Society, 209 (1975), pp. 225–235.

[51] J. Dekker and J. Myhill, *Retraceable sets*, Canad. J. Math., 10 (1985), pp. 357–373.

[52] O. Demuth, *A notion of semigenericity*, Commenationes Mathematicae Universitatis Carolinae, 28 (1987), pp. 71–84.

[53] R. P. Dilworth, *A decomposition theorem for partially ordered sets*, Annals of Math., 51 (1950), pp. 161–166.

[54] R. Downey, *Abstract dependence, recursion theory and the lattice of recursively enumerable filters.* Ph.D. Thesis (Monash University), 1982.

[55] ———, *On $\pi_1^0$ classes and their ranked points*, Notre Dame J. of Formal Logic, 32 (1991), pp. 499–512.

[56] R. Downey, *On presentations of algebraic structures*, in Complexity, Logic and Recursion Theory, A. Sorbi, ed., Lecture Notes in Pure and Appl. Math. #197, Marcel Dekker, 1997, pp. 157–206.

[57] R. Downey, *Recursion theory and linear orderings*, in Handbook of Recursive Mathematics, Y. Ersov, S. Goncharov, A. Nerode, and J. B. Remmel, eds., Studies in Logic Found. Math #139, Elsevier, 1998, pp. 823–975.

[58] R. Downey and D. Hirschfeldt, *Algorithmic Randomness and Complexity*, Springer-Verlag, 2010.

[59] R. Downey, C. Jockusch, and M. Stob, *Array non-recursive sets and multiple permitting arguments*, in Recursion Theory Week (Proceedings Oberwolfach 1989), K. Ambos-Spies, G. Muller, and G. Sacks, eds., Lecture Notes in Mathematics #1432, Springer-Verlag, 1990, pp. 141–173.

[60] R. Downey and S. Kurtz, *Recursion theory and ordered groups*, Annals of Pure and Applied Logic, 32 (1986), pp. 137–151.

[61] R. Downey and S. Lempp, *Proof-theoretic strength of the dushnik-miller theorem on countable linear orderings*, in Recursion Theory and Complexity (Proc. 1997 Kazan conference), M. Arslanov and S. Lempp, eds., W. de Gruyter, 1999, pp. 55–57.

[62] B. Dushknik and E. W. Miller, *Partially ordered sets*, Amer. J. Math, 1941 (63), pp. 600–610.

[63] A. Ehrenfeucht, *Separable theories*, Bull. Acad. Polon. Sci. Sér. Sci. Math. Astronom. Phys., 9 (1961), pp. 17–19.

[64] R. M. Friedberg, *Three theorems on recursive enumeration*, Journal of Symbolic Logic, 23 (1958), pp. 309–316.

[65] H. Friedman, *Systems of second order arithmetic with restricted induction i, ii (abstracts)*, Journal of Symbolic Logic, 41 (1976), pp. 557–559.

[66] H. Friedman, S. Simpson, and R. Smith, *Countable algebra and set existence theorems*, Annals of Pure and Applied Logic, 25 (1983), pp. 141–181.

[67] H. Friedman, S. Simpson, and R. Smith, *Countable algebra and set existence theorems*, Annals of Pure and Applied Logic, 25 (1983), pp. 141–181.

[68] H. Friedman, S. Simpson, and R. Smith, *Addendum to countable algebra and set existence theorems*, Annals of Pure and Applied Logic, 28 (1985), pp. 319–320.

[69] P. Gacs, *On the symmetry of algorithmic information*, Soviet Mathematics–Doklady, 15 (1974), pp. 1477–1480.

[70] D. Gale and F. M. Stewart, *Infinite games with perfect information*, Ann. Math. Studies, 28 (1953), pp. 245–266.

[71] R. O. Gandy, *Proof of mostowski's conjecture*, Bull. Acad. Polon. Sci., 8 (1960), pp. 571–575.

[72] W. Gasarch and G. Martin, *Index sets in recursive combinatorics*, in Logical Methods in honor of Anil Nerode's Sixtieth Birthday, J. Crossley, J. B. Remmel, R. Shore, and M. Sweedler, eds., Birkhäuser, 1993, pp. 352–385.

[73] M. Gelfond and V. Lifschitz, *The stable semantics for logic programs*, in Proceedings 5th Int'l. Symp. Logic Programming, MIT Press, 1988, pp. 1070–1080.

[74] ——, *Logic programs with classical negation*, in Proceedings 7th Int'l. Conference on Logic Programming, MIT Press, 1990, pp. 579–597.

[75] S. Grigorieff, *Every recursive linear ordering has a copy in dtime(n)*, Journal of Symbolic Logic, 55 (1990), pp. 260–276.

[76] M. Hall, *Distinct representatives of subsets*, Bulletin of the American Mathematical Society, 54 (1948), pp. 922–926.

[77] D. Harel, *Hamiltonian paths in infinite graphs*, Israel J. Math., 76 (1991), pp. 317–336.

[78] L. Hay, A. Manaster, and J. Rosenstein, *Concerning partial recursive similarity transformations of linearly ordered sets*, Pacific J. Math., 71 (1977), pp. 57–70.

[79] E. Hermann, *Unpublished manuscript.*

[80] P. G. Hinman, *Recursion-Theoretic Hierarchies*, Perspectives in Mathematical Logic, Springer-Verlag, 1978.

[81] J. Hirst, *Marriage theorems and reverse mathematics*, in Logic and Computation (Pittsburgh 1987), Contemporary Mathematics #106, American Mathematical Society, 1990, pp. 181–196.

[82] J. E. Hopcroft and J. D. Ullman, *Formal Languages and Their Relations to Automata*, Addison-Wesley, 1969.

[83] T. Jech, *Set Theory*, Academic Press, 1978.

[84] C. G. Jockusch, *$\pi_1^0$ classes and boolean combinations of recursively enumerable sets*, Journal of Symbolic Logic, 39 (1974), pp. 95–96.

[85] ——, *Degrees of functions with no fixed points*, in Logic, Methodology and Philosophy of Science VIII, Studies in Logic and Foundations of Mathematics, Elsevier, 1989, pp. 191–201.

[86] C. G. Jockusch, A. Lewis, and J. B. Remmel, *$\pi_1^0$ classes and rado's selection principle*, Journal of Symbolic Logic, 56 (1991), pp. 684–693.

[87] C. G. Jockusch and T. G. McLaughlin, *Countable retracing functions and $\pi_1^0$ predicates*, Pacific J. Math., 30 (1969), pp. 67–93.

[88] C. G. Jockusch and R. Shore, *Pseudo-jump operators ii*, Journal of Symbolic Logic, 49 (1984), pp. 1205–1236.

[89] C. G. Jockusch and R. Soare, *A minimal pair of $\pi_1^0$ classes*, Journal of Symbolic Logic, 36 (1971), pp. 66–78.

[90] ——, *Degrees of members of $\pi_1^0$ classes*, Pacific J. Math., 40 (1972), pp. 605–616.

[91] ——, *$\pi_1^0$ classes and degrees of theories*, Transactions of the American Mathematical Society, 173 (1972), pp. 33–56.

[92] K. Gödel, *Über die lange der beweise*, Ergeb. Math. Koll., 7 (1936), pp. 23–24.

[93] H. Keirstead, *Recursive and on-line graph coloring*, in Handbook of Recursive Mathematics, Y. Ersov, S. Goncharov, A. Nerode, and J. B. Remmel, eds., Studies in Logic Found. Math #139, Elsevier, 1998, pp. 1233–1269.

[94] H. Kierstead, *An effective version of dilworth's theorem*, Transactions of the American Mathematical Society, 268 (1981), pp. 63–77.

[95] H. Kierstead, G. McNulty, and W. Trotter, *Recursive dimension for partially ordered sets*, Order, 1 (1984), pp. 67–82.

[96] N. Klarlund, *Progress measures and finite arguments for infinite computation.* Ph.D. Thesis (Cornell University), 1990.

[97] S. C. Kleene, *Recursive predicates and quantifiers*, Transactions of the American Mathematical Society, 53 (1943), pp. 41–73.

[98] ——, *A symmetric form of Gödel's theorem*, Ind. Math., 12 (1950), pp. 244–246.

[99] ——, *Recursive functionals and quantifiers of finite types i*, Transactions of the American Mathematical Society, 91 (1959), pp. 193–218.

[100] J. Knight, A. Lachlan, and R. Soare, *Two theorems on degrees of models of true arithmetic*, Journal of Symbolic Logic, 49 (1984), pp. 425–436.

[101] K. Ko, *Complexity Theory of Real Functions*, Birkhauser, 1991.

[102] ——, *On the computability of fractal dimensions and julia sets*, Annals of Pure and Applied Logic, 93 (1998), pp. 195–216.

[103] A. N. Kolmogorov, *Three approaches to the quantitative defintion of information*, Problems of Information Transmission, 1 (1965), pp. 1–7.

[104] G. Kreisel, *A variant to hilbert's theory of the foundations of arithmetic*, British J. Philosophy of Science, 4 (1953), pp. 107–129.

[105] ——, *Analysis of the cantor-bendixson theorem by means of the analytic hierarchy*, Bull. Acad. Polon. des Sciences Ser. Math. Astronom. et Phys, 7 (1959), pp. 621–626.

[106] A. Kucera, *Measure, $\pi^0_1$ classes and complete extensions of pa*, in Recursion Theory Week (Proceedings Oberwolfach 1984), H. D. Ebbinghaus, G. Muller, and G. Sacks, eds., Lecture Notes in Mathematics #1141, Springer-Verlag, 1985, pp. 245–259.

[107] ———, *An alternative, priority free, solution to post's problem*, in Mathematical Foundations of Computer Science (Bratislava 1986), vol. 233 of Lecture Notes in Computer Science, Springer-Verlag, 1986, pp. 493–500.

[108] ———, *Randomness and generalizations of fixed point free functions*, in Recursion Theory Week (Proceedings Oberwolfach 1989), K. Ambos-Spies, G. Muller, and G. Sacks, eds., Lecture Notes in Mathematics #1432, Springer-Verlag, 1990, pp. 245–254.

[109] K. Kuratowski, *Some problems concerning semi-continuous set-valued mappings*, in Set-Valued Mappings, Selections and Topological Properties of $2^X$, Lecture Notes in Mathematics #171, Springer-Verlag, 1970, pp. 45–48.

[110] K. Kuratowski and A. Mostowski, *Set Theory*, Studies in Logic #86, Elsevier North-Holland, 1968.

[111] A. H. Lachlan, *On the lattice of recursively enumerable sets*, Transactions of the American Mathematical Society, 130 (1968), pp. 1–37.

[112] D. Lacombe, *Les ensembles recursivement ouverts ou fermes, et leurs applications a l'analyse recursive i*, Comptes Rendus Hebdomaires des Sciences, 245 (1957), pp. 1040–1043.

[113] ———, *Les ensembles recursivement ouverts ou fermes, et leurs applications a l'analyse recursive ii*, Comptes Rendus Hebdomaires des Sciences, 246 (1958), pp. 405–418.

[114] L. Lawton, *Almost complemented $\pi_1^0$ classes*, Archive for Mathematical Logic, 45 (2006), pp. 555–568.

[115] S. Lempp, *Hyperarithmetical index sets in recursion theory*, Transactions of the American Mathematical Society, 303 (1987), pp. 559–583.

[116] F. W. Levi, *Arithmetische gesetze im gebiete diskreter gruppen*, Rend. Circ. Mat. Palermo, 35 (1913), pp. 225–236.

[117] L. A. Levin, *The concept of a random sequence*, Dokl. Akad. Nauk SSSR, 212 (1973), pp. 548–550.

[118] P. . Lévy, *Théorie de l'Addition des Variables Aleatoires*, Gauthier-Villars, 1937 (second edition 1954).

[119] J. Lloyd, *Foundations of Logic Programming*, Springer-Verlag, 1989.

[120] A. Manaster and J. Rosenstein, *Effective matchmaking*, Proceedings London Mathematical Society, 25 (1972), pp. 615–654.

[121] ———, *Effective matchmaking and $k$-chromatic graphs*, Proceedings of the American Mathematical Society, 39 (1973), pp. 371–378.

[122] V. W. Marek, *Nonmonotonic Logic*, Springer-Verlag, 1993.

[123] W. Marek, A. Nerode, and J. B. Remmel, *Nonomonotonic rule systems i*, Annals of Mathematics and Artificial Intelligence, 1 (1990), pp. 241–273.

[124] ——, *How complicated is the set of stable models of a recursive logic program?*, Annals of Pure and Applied Logic, 56 (1992), pp. 119–135.

[125] ——, *Nonomonotonic rule systems ii*, Annals of Mathematics and Artificial Intelligence, 5 (1992), pp. 229–264.

[126] ——, *A context for belief revision: Forward-chaining normal nonmonotonic rule systems*, Annals of Pure and Applied Logic, 67 (1994), pp. 269–324.

[127] ——, *The stable models of a predicate logic program*, Journal of Logic Programming, 21 (1994), pp. 129–154.

[128] ——, *Nonmonotonic rule systems with recursive sets of restraints*, Archive for Mathematical Logic, 36 (1997), pp. 339–384.

[129] D. Marker, *Degrees of models of arithmetic*, in Logic Colloquium '81 (Proceedings of the Herbrand Symposium), North Holland, 1982, pp. 233–243.

[130] D. Martin, *Classes of recursively enumerable sets and degrees of unsolvability*, Zeitschrift für Mathematische Logik und Grundlagen der Mathematik, 12 (1966), pp. 295–310.

[131] D. Martin and M. Pour-El, *Axiomatizable theories with few axiomatizable extensions*, Journal of Symbolic Logic, 35 (1970), pp. 205–209.

[132] G. Martin, *Cantor singletons, rank-faithful trees, and other topics in recursion theory.* Ph.D. Thesis, University of Maryland, 1993.

[133] P. Martin-Lof, *The definition of random sequences*, Information and Control, 9 (1966), pp. 602–619.

[134] D. McDermott, *Nonmonotonic logic ii: nonmonotonic modal theories*, Journal of the ACM, 29 (1982), pp. 33–57.

[135] D. McDermott and J. Doyle, *Nonmonotonic logic i*, Artificial Intelligence, 13 (1980), pp. 41–72.

[136] Y. Medvedev, *Degrees of difficulty of the mass problem*, Doklady Akad. Nauk SSSR, 104 (1955), pp. 501–504.

[137] G. Metakides and A. Nerode, *Recursion theory and algebra*, in Algebra and Logic, J. N. Crossley, ed., Lecture Notes in Math. #450, Springer-Verlag, 1975, pp. 209–219.

[138] G. Metakides and A. Nerode, *Effective content of field theory*, Annals of Pure and Applied Logic, 17 (1979), pp. 289–320.

[139] J. Miller, $\pi_1^0$ *classes in analysis and topology.* Ph.D. thesis, Cornell University, 2002.

[140] M. Minsky, *A framework for representing knowledge*, in The Psychology of Computer Vision, McGraw-Hill, 1975, pp. 211–272.

[141] Y. Moschovakis, *Elementary Induction on Abstract Structures*, Studies in Logic #100, North-Holland, 1974.

[142] ——, *Descriptive Set Theory*, Studies in Logic #100, North-Holland, 1980.

[143] A. Nerode, *General topology and partial recursive functionals*, in Summaries of talks presented at the Summer Institute for Symbolic Logic, Cornell University, 1957, pp. 247–251.

[144] A. Nerode and W. Huang, *Applications of pure recursion theory to recursive analysis*, Acta Sinica, 28 (1985).

[145] A. Nerode and J. B. Remmel, *Complexity theoretic algebra ii, the free boolean algebra*, Annals of Pure and Applied Logic.

[146] ——, *Complexity theoretic algebra I, vector spaces over finite fields*, Computer Science Press, 1987, pp. 218–239.

[147] A. Nerode and J. B. Remmel, *Complexity theoretic algebra: vector space bases*, in Feasible Mathematics, S. Buss and P. Scott, eds., Progress in Computer Science and Applied Logic #9, Birkhäuser, 1990, pp. 293–319.

[148] A. Nies, *Intervals of the lattice of computably enumerable sets*, Bull. London Math. Soc., 29 (1997), pp. 683–692.

[149] ——, *Effectively dense boolean algebras and their applications*, Transactions of the American Mathematical Society, 352 (2000), pp. 4989–5013.

[150] A. Nies, *Computability and randomness*, vol. 51 of Oxford Logic Guides, Oxford University Press, 2009.

[151] P. Odifreddi, *Classical Recursion Theory*, Elsevier, 1989.

[152] V. P. Orevkov, *A constructive mapping of the square onto itself displacing every constructive point*, Sov. Math. Dokl., 4 (1963), pp. 1253–1256.

[153] J. Owings, *Rank, join and cantor singletons*, Archive for Mathematical Logic, 36 (1997), pp. 313–320.

[154] C. Papadimitriou, *Computational Complexity*, Addison-Wesley, 1994.

[155] E. L. Post, *Recursively enumerable sets of positive integeres and their decision problems*, Bulletin of the American Mathematical Society, 50 (1944), pp. 284–316.

[156] M. B. Pour-El and J. I. Richards, *Computability in Analysis and Physics*, Perspectives in Mathematical Logic, Springer-Verlag, 1989.

[157] R. Rado, *Axiomatic treatment of rank in infinite sets*, Canadian J. Math., 1 (1949), pp. 337–343.

[158] R. Reiter, *A logic for default reasoning*, Artificial Intelligence, 13 (1980), pp. 81–132.

[159] J. B. Remmel, *Recursively enumerable boolean algebras*, Annals of Pure and Applied Logic, 14 (1978), pp. 75–107.

[160] ——, *On the effectiveness of the schroder-bernstein theorem*, Proceedings of the American Mathematical Society, 83 (1981), pp. 379–386.

[161] ——, *Graph colorings and recursively bounded $\pi^0_1$ classes*, Annals of Pure and Applied Logic, 32 (1986), pp. 185–194.

[162] ——, *When is every recursive linear ordering of type $\mu$ recursively isomorphic to a p-time linear order over the binary representation of the natural numbers?*, in Feasible Mathematics, S. Buss and P. Scott, eds., Prog. in Comp. Science and Appl. Logic #9, Birkhäuser, 1990, pp. 321–341.

[163] ——, *Polynomial-time categoricity and linear orderings*, in Logical Methods, J. Crossley, J. B. Remmel, R. Shore, and M. Sweedler, eds., Prog. in Comp. Science and Appl. Logic #12, Birkhäuser, 1993, pp. 321–341.

[164] F. Riazati, *The lattice of $\pi^0_1$ classes.* Ph.D. thesis, University of Florida, 2001.

[165] J. Rosenstein, *Linear Orderings*, Acad. Press, 1982.

[166] B. Rosser, *Extensions of some theorems of Gödel and church*, Journal of Symbolic Logic, 1 (1936), pp. 87–91.

[167] G. Sacks, *Degrees of Unsolvability*, Annals of Mathematics Studies #55, Princeton University Press, 1963.

[168] J. Schmerl, *Recursive colorings of graphs*, Canadian J. Math., 32 (1980), pp. 821–830.

[169] ——, *The effective version of brooks' theorem*, Canadian J. Math, 34 (1982), pp. 1036–1046.

[170] C. P. Schnorr, *A unified approach to the definition of random sequences*, Mathematical Systems Theory, 5 (1971), pp. 246–258.

[171] D. Scott, *Algebras of sets binumerable in complete extensions of arithmetic*, in Recursive Function Theory, J. Dekker, ed., Proceedings of Symposia in Pure Mathematics #5, American Mathematical Society, 1962, pp. 117–121.

[172] D. Scott and S. Tennenbaum, *On the degrees of complete extensions of arithmetic*, Notices Amer. Math. Soc., 7 (1960), pp. 242–243.

[173] J. Shoenfield, *Degrees of formal systems*, Journal of Symbolic Logic, 23 (1958), pp. 389–392.

[174] ——, *Degrees of models*, Journal of Symbolic Logic, 25 (1960), pp. 233–237.

[175] ——, *Mathematical Logic*, Addison-Wesley, 1967.

[176] S. G. Simpson, *Subsystems of Second Order Arithmetic*, Springer-Verlag, 1999.

[177] ——, *$\pi^0_1$ sets and models of $wkl_0$*, in Reverse Mathematics, Lecture Notes in Logic, Association for Symbolic Logic, 2004. to appear.

[178] ——, *Mass problems and randomness*, Bulletin of Symbolic Logic, (2005). to appear.

[179] R. Soare, *Cohesive sets and recursively enumerable dedekind cuts*, Pacific J. Math., 31 (1969), pp. 215–231.

[180] ——, *Recursion theory and dedekind cuts*, Transactions of the American Mathematical Society, 139 (1969), pp. 271–294.

[181] ——, *Recursively Enumerable Sets and Degrees*, Springer-Verlag, 1987.

[182] D. R. Solomon, *$\pi^0_1$ classes and orderable groups*, Annals of Pure and Applied Logic, 115 (2002), pp. 279–302.

[183] A. Sorbi, *The medvedev lattice of degrees of difficulty*, in Computability, Enumerability, Unsolvability, London Math. Soc. Lecture Notes #224, Cambridge University Press, 1996, pp. 289–312.

[184] C. Spector, *Recursive well-orderings*, Journal of Symbolic Logic, 20 (1955), pp. 151–163.

[185] ——, *Hyperarithmetic quantifiers*, Fundamenta Mathematica, 48 (1959), pp. 313–320.

[186] ——, *Inductively defined sets of natural numbers*, in Infinitistic Methods (Proceedings of Warsaw Symposium), Pergamon Press, 1961, pp. 301–384.

[187] L. Staiger, *Hierarchies of recursive $\omega$-languages*, J. Inform. Process. Cybernetics EIK, 22 (1986), pp. 219–241.

[188] ———, *Recursive automata on infinite words*, in Proc. STACS '93, vol. 665 of Lecture Notes in Computer Science, Springer-Verlag, 1993, pp. 629–639.

[189] ———, *$\omega$-languages*, in Handbook of Formal Languages, Volume 3, Springer-Verlag, 1997, pp. 339–387.

[190] L. Staiger and K. Wagner, *Recursive $\omega$-languages*, in Fundamentals of Computation Theory, vol. 56 of Lecture Notes in Computer Science, Springer-Verlag, 1977, pp. 532–537.

[191] A. Tarski, *Logic, semantics, metamathematics*, Oxford University Press, 1956.

[192] A. Tarski, A. Mostowski, and R. M. Robinson, *Undecidable Theories*, North-Holland, 1953.

[193] S. Tennenbaum, *Non-archimedean models for arithmetic*, Notices Amer. Math. Soc., 6 (1959), p. 270.

[194] B. A. Trakhtenbrot, *Tabular representation of recursive operators*, Doklady Acad. Nauk, 101 (1955), pp. 417–420.

[195] M. van Lambalgen, *Random sequences.* Ph.D. thesis, University of Amsterdam, The Netherlands, 1987.

[196] J. Ville, *Étude Critique de la Notion de Collectif*, Gauthier-Villars, Paris, 1939.

[197] R. von Mises, *Grundlagen der wahrscheinlichkeitsrechnung*, Math Zeitschrift, 5 (1919), pp. 52–99.

[198] K. Weihrauch, *An Introduction to Computable Analysis*, Springer-Verlag, 2000.