Fakultät für Mathematik und Informatik

Ruprecht-Karls-Universität Heidelberg

Π_1^0 Classes - Boundedness and Degrees

Diplomarbeit

im Studiengang Mathematik

vorgelegt von

Lutz Büch

aus Arnsberg (Nordrheinwestfalen)

März 2012

unter der Betreuung von

Herrn Priv.-Doz. Dr. Wolfgang Merkle

Arbeitsgruppe Mathematische Logik und Theoretische Informatik

Contents

1 Introduction

In this thesis we consider Π^0_1 class, which can roughly be defined as the sets of infinite paths through computable trees. Historically, Π^0_1 classes have first been considered by Shoenfield ([Sho60]) in an investigation of the complexity of complete extensions of computably axiomatizable first-order theories, such as Peano arithmetic.

We aim for a comprehensive access to the notion of Π^0_1 classes. First of all we motivate the notion of tree used in the context of Π^0_1 classes, which is a special case of the common graph-theoretic notion. We find, that while formally a special case, its definition is reasonable, since it captures all considered structural phenomena. Also, we do not only consider Π^0_1 classes in 2^ω, but in all of ω^ω. However, we restrict ourselves to Π^0_1 classes that are bounded in some sense. For this purpose, we investigate different notions of boundedness before identifying two somewhat universal classes of bounded Π^0_1 classes, then simply called *bounded* and *computably bounded Π^0_1 classes*, and a nice way of representing them. After that, we turn to the core of the thesis.

That is, we investigate what spectra of degrees the members of a Π^0_1 class of a given kind can bear. On the one hand, we try to find members of particularly low complexity in some sense. In order to do this, we establish *basis theorems* that tell us, that any Π^0_1 class of a class $\mathcal{C}$ has a member of a class of functions $\mathcal{B}$, called basis. On the other hand, we showcase some Π^0_1 classes, which by example show that any according basis, in return, has to contain a member of some property.

Returning to the initial motivation for our consideration, we apply the results to logical theories, and Peano arithmetic in particular, thereby nourishing the findings of Gödel's Incompleteness Theorem. An important result is that the class of complete extensions of Peano arithmetic form a *universal* computably bounded Π^0_1 class. That means that every such extension computes some member of every computably bounded Π^0_1 class. The degrees of theses extensions, called PA degrees, can even be characterized by this property. We assemble other characterizations and results on PA degrees.

A notable feature of this thesis is the generalization of Shoenfield's construction for the improvement of the Kreisel Basis Theorem. This generalization seems not to have been done yet. The originally resulting Kreisel-Shoenfield Basis Theorem is properly weaker than the Low Basis Theorem of Jockusch and Soare, but the generalization has more ramifications than just that theorem. For one, it implies that one can delete any maximal elements of a wide range of bases for any of the considered classes of Π^0_1 classes. This then shows that there are chains of low degrees of arbitrary finite length. The same is implied for hyperimmune-free degrees. As another application, the generalization of Shoenfield's construction sharpens Solovay's characterization of the PA degrees. By this tightened characterization, it provides an immediate alternative proof of Scott and Tennenbaum's result that there is no minimal degree that is PA. And in fact, it implies the apparently novel result that there is an infinite chain of PA degrees below every PA degree.

2 Preliminaries

First of all, we agree on some notation in order for this thesis to be accessible for readers of other texts on recursion theory. Readers not familiar with recursion theory are referred to the comprehensive introductory book [Odi92]. After that, we list some basic results of recursion theory that will be required in later chapters. If not specified differently, these can be found in [Odi92].

As it is general convention in recursion theory, we will not explicate the effective parts of every proof by means of, say, defining a specific Turing machine. Rather, we will outline an algorithm or an argument that such an algorithm exists, and refer to the Church-Turing thesis.

We assume familiarity with the basics of recursion theory, especially relative computability, Turing degrees and the arithmetical and analytical hierarchies.

2.1 Notation and definitions

Most of the following notions are ubiquitous and we will only give little concrete specification and properties.

$\omega, \mathbb{N}$	the set of natural numbers Natural numbers will usually be denoted by lower-case Latin letters $e, i, j, k, l, m, n, s, t$ and sets of natural numbers by upper-case Latin letters A, B, C, M
$c_A \colon \mathbb{N} \to \{0,1\}$	the total characteristic function of a set A
$\chi_A \colon \mathbb{N} \to \{1\}$	the partial characteristic function of a set A
$\sigma = (k_0, \ldots, k_n) \in \omega^{n+1} \subset \omega^*$	a string consisting of n natural numbers Finite strings of length greater zero

	will usually be denoted by ρ, σ, τ.
λ	the empty string
	Sets of strings, especially trees (to be defined) will usually be denoted by S, T.
$\lvert\sigma\rvert = \lvert(k_0, \dots, k_n)\rvert = n+1$	the length of a string
$\sigma(i) = k_i$	the ith number of the string $\sigma \in \omega^*$
$x = (k_0, k_1, \dots) \in \omega^\omega$	an infinite string or sequence
	Infinite strings will also be denoted by f, g as they will often be identified with functions.
ω^ω	Baire space
$2^\omega = \{0,1\}^\omega$	Cantor space
$\sigma \circ \tau$	concatenation of strings $\sigma, \tau \in \omega^*$
$\sigma \circ x$	concatenation of a string $\sigma \in \omega^*$ with an infinite string $x \in \omega^\omega$
$\sigma \restriction j$, $x \restriction j$	the prefix of length j of a string σ (x); $(\sigma \restriction j = \sigma$ if $\lvert\sigma\rvert \leq j)$
	The term *initial segment* may be used interchangeably with *prefix*
$I(\sigma) = \{x \in \omega^\omega \mid x \restriction \lvert\sigma\rvert = \sigma\}$	the interval of infinite strings extending σ
$\sigma <_{lex} \tau$	lexicographical (linear) order on strings of a fixed length; $\sigma, \tau \in \mathbb{N}^{n+1}$
	$\exists 0 \leq j < n \; \sigma \restriction j-1 = \tau \restriction j-1 \wedge \sigma(j) < \tau(j)$
$\tau \prec \sigma$	σ is a prefix of σ
$Pref(S) = \{\tau \mid \exists \sigma \in S \; \tau \prec \sigma\}$	the set of prefixes of a set of strings $S \subseteq \omega^*$
$\tau \colon \mathbb{N}^2 \to \mathbb{N}$	a binary bijective primitive recursive coding function
	$\tau(x, y) = div(x^2 + 2xy + y^2 + x + 3y, 2)$
$\pi_0, \pi_1 \colon \mathbb{N} \to \mathbb{N}$	the associated primitive recursive projection functions:
	$\forall m, n \in \mathbb{N} m = \pi_0(\tau(m,n)) \wedge n = \pi_1(\tau(m,n))$
$\langle\cdot\rangle \colon \mathbb{N}^* \to \mathbb{N}$	a binary bijective coding function for arbitrary long finite strings
	$\langle\sigma\rangle = \tau(\lvert\sigma\rvert - 1, \tau(\sigma(0), \tau(\sigma(1), \tau \dots \sigma(\lvert\sigma\rvert - 1) \dots)),$

	$\langle\lambda\rangle = 0$ We may by $\langle\cdot\rangle$ sometimes identify sets of strings with sets of natural numbers, when conventient and appropriate.

We assume a standard enumeration of the partial computable functions by Turing machines. Thereby, we may clearly speak of the computation of a partial function on a given input for a number of steps. We will identify oracle Turing machines with ordinary Turing machines, as in [DH10].

φ_e	the eth partial computable function
$\varphi_e(x)\downarrow$, $\varphi_e(x)[t]\downarrow$	convergence on input x (after t many steps)
$\varphi_e(x)\uparrow$, $\varphi_e(x)[t]\uparrow$	divergence on input x (after t many steps)
$W_e = \{x \mid \varphi_e(x)\downarrow\}$	the eth computably enumerable set
$\varphi_e^f(x)$	the eth computation with oracle $f \in \omega^\omega$ on input x
φ_e^f	the eth partial f-computable function We may also define sets of natural numbers as oracles by identifying them with their characteristic function.
$\varphi_e^\sigma(x)$	the eth computation with finite oracle $\sigma \in \omega^*$ on input x A computation with finite oracle also diverges, when it makes such a request that the oracle is too short.
$K = \{e \mid \varphi_e(e)\downarrow\} = \emptyset'$	the (diagonal) halting problem
$A' = \{e \mid \varphi_e^A(e)\downarrow\}$	the jump of a set A
$f' = \{e \mid \varphi_e^f(e)\downarrow\}$	the jump of a function f
$\leq_T$	Turing reducibility, arbitrarily between sets and functions
$\leq_m$	m-reducibility In this thesis, when we speak of *degrees*, we will always mean Turing degrees. They will be denoted by boldface lower-case Latin letters $\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}, \mathbf{p}$.
$\mathbf{a} \leq \mathbf{b}$	the partial order of T-degrees, induced by $\leq_T$
$A \oplus B = \{2n \mid n \in A\}$ $\cup\{2n+1 \mid n \in B\}$	the join of two sets A, B
$\mathbf{a} \vee \mathbf{b}$	the join of two T-degrees, induced by $\oplus$

Remark. Note that *Baire space* stands for two different notions; one topologic and applicable to many spaces, while the other one is set-theoretic and is just the name for ω^ω. It is also a Baire space in the topologic sense.

The Baire space $\mathbb{N}^\mathbb{N}$ is often considered a topological space. Although in this thesis most of the time we do not need that, at times it proves very useful. One does not equip the space with just any topology, of course, but with a very natural one. It can be defined equivalently in at least three ways. The most economic definition of that topology is that it is the product topology of $\mathbb{N}^\mathbb{N}$, where $\mathbb{N}$ has the discrete topology.

By defining $d\colon \mathbb{N}^\mathbb{N} \times \mathbb{N}^\mathbb{N} \to \mathbb{R}, d(x, y) := 2^{-n}$, where $n = \max\{|\sigma| \mid x, y \in I(\sigma)\}$, one gets what easily can be verified to be a metric for $\mathbb{N}^\mathbb{N}$, which induces the same topology.

And finally, the family $(\{x|x(m) = n\})_{m,n\in\mathbb{N}}$, when used as a sub-basis for a topology, makes the family $(I(\sigma))_{\sigma\in\mathbb{N}^*}$ a basis of that topology. But that is exactly the family of open balls of the above metric. So this sub-basis induces also the same topology.

2.2 The notion of tree

In this section we take a look at the notion of tree, which is crucial in the treatment of Π^0_1 classes. We will discuss its specific definition in the context of Π^0_1 classes in contrast to the more common graph-theoretic definition.

Definition 2.2.1. A set $T \subseteq \mathbb{N}^*$ is a *tree* if it is closed under initial sequences, i.e. if for all strings $\sigma \in T$, all initial segments $\tau \prec \sigma$ are also members of that set: $\tau \in T$. The empty string λ is the *root* of T, and we call elements $\sigma \in T$ *nodes* of T.

Note the difference between this definition and the more general and more common definition of a tree as a simple acyclic connected (countable) undirected graph.

A tree in our sense may of course also be considered an undirected graph. We define an appropriate edge relation:

Definition 2.2.2. For a tree T we define the edge relation $E_T := \{(\sigma, \tau) | \exists k \in \mathbb{N}\ \sigma = \tau \circ k \vee \sigma \circ k = \tau\}$.

It is $(\sigma, \tau) \in E_T$ iff $(\tau, \sigma) \in E_T$, so E_T is undirected. Also, $(\sigma, \sigma) \notin E_T$ for any σ, i.e. (T, E_T) is a simple graph.

Throughout this thesis, we will only consider undirected simple graphs and whenever we speak of *graphs* or *trees* in the graph-theoretical sense, we will hence always mean only undirected simple graphs or trees.

We define some basic notions for graphs we want to consider in trees.

Definition 2.2.3. In a graph (V, E), a *walk* is a sequence of vertices $v_0, \dots, v_n \in V$, so that any two successive vertices are adjacent, i.e. connected by an edge: For all $0 \leq i < n$, $(v_i, v_{i+1}) \in E$. Here, n is the *length* of the walk $v_0, \dots, v_n$. A *path* is a walk so that its vertices are mutually different, $v_i \neq v_j$ for all $i \neq j$. An *infinite path* with one or without any *end-point*, respectively, is an infinite sequence $v_0, v_1, v_2, \dots$ or $\dots, v_{-1}, v_0, v_1, \dots$ so that every subsequence $v_i, v_{i+1}, \dots, v_{i+n}$ is a path. In the first case, v_0 is the end-point. A *cycle* is a walk $v, v_0, \dots, v_n, v$ with $n > 0$, such that $v, v_0, \dots, v_n$ is a path.

Although our definition of a tree T with the according edge relation E_T satisfies also the graph-theoretical definition - as we will see now -, a general graph need not have a node defined as a root. But our main interest will be in infinite paths with one end-point through trees. Most of the time, we will be interested only in their complexity in terms of Turing-degrees. And because this notion is robust against finite differences, we may as well define that any such infinite path should have a certain node defined as the root of the tree as its end-point.

We will now observe that our definition of a tree is a special case of the more general definition, but already captures all relevant phenomena.

Lemma 2.2.1. *Every tree T is also a countable tree in the graph-theoretical sense.*

Proof. Let T be a tree. Since $T \subseteq \mathbb{N}^*$, T is countable. Using the coding function $\langle\rangle$ we can even consider it a subset of $\mathbb{N}$. Note that this is a recursive operation in both ways.

Because T is closed under initial segments, for every node there is a path to the root λ. Adjoining these paths one to the other accordingly, one sees that there is a path from every node to any other node, i.e. T is connected.

Suppose, there were two different strings $\rho, \tau \in T$ connected by two different paths $\rho, \sigma_1, \ldots, \sigma_n, \tau$ and $\rho, \sigma'_1, \ldots, \sigma'_{n'}, \tau$. Then $\sigma_1 \neq \sigma'_1$, without loss of generality; otherwise set ρ as σ_i so that $\sigma_j = \sigma'_j$ for all $1 \leq j \leq i < \min(n, n')$ and $\sigma_{i+1} \neq \sigma'_{i+1}$, and rename the vertices appropriately. If no such i exists, then $\sigma_j = \sigma'_j$ for all $1 \leq j \leq \min(n, n')$, with $\min(n, n') = n'$ without loss of generality. But then $\tau, \sigma'_{n'} = \sigma_{n'}, \sigma_{n'+1}, \ldots, \sigma_n, \tau$ is a cycle.

Since both are paths and the direct predecessor of a string is unique, either $\sigma_1 \prec \rho$ and $\rho \prec \sigma'_1$, or $\rho \prec \sigma_1$ and $\sigma'_1 \prec \rho$, or $\rho \prec \sigma_1$ and $\rho \prec \sigma'_1$. In any case, one of the two strings σ_1 and σ'_1 is a direct successor of ρ. Let it be σ_1 without loss of generality.

Since $\rho \prec \sigma_1$ and the direct predecessor of a string is unique and in a path there are no repeated nodes, $\sigma_1 \prec \sigma_2$. By induction, $\sigma_i \prec \sigma_{i+1}$ for all $1 \leq i \leq n-1$ and $\sigma_n \prec \tau$. The same argumentation works for the other path, if $\rho \prec \sigma'_1$. But, again, since the direct predecessor of a string is unique, a path with given end-node τ going successively over the direct predecessors is unique and then both paths would be identical. So, $\sigma'_1 \prec \rho$.

There must be $2 \leq j < n'$ so that $\sigma'_{j-1} \prec \sigma'_j$, because otherwise the length of the σ'_i would always decrease. And that cannot be, because $|\rho| \leq |\tau|$ which is shown by the first path. Then, like above, $\sigma'_i \prec \sigma'_{i+1}$ for all $j-1 \leq i \leq n'-1$. But then, again because the predecessors of τ of given length are unique, $\sigma'_i = \rho$ for some $i \geq j$. And so $\tau, \sigma'_1, \ldots, \sigma'_{n'}, \tau$ cannot be a path.

So, there are no cycles in T and therefore T is a tree in the graph theoretic sense. □

Now, consider the inverse direction. Since we are concerned with computability, we assume G not only to be computable, but also that the nodes of G form a subset of $\mathbb{N}$, without loss of generality.

Lemma 2.2.2. *For any graph-theoretical tree $G = (V, E)$, $G \subseteq \mathbb{N}$, there is a tree that is isomorphic to G as a graph.*

Proof. If G is empty, the claim is true. Otherwise, choose $r \in G$. Define $d_r : G \to \mathbb{N}$ as the length of the unique path from a given node of G to r. Since G is connected, d_r is well-defined.

Define
$T := \{\sigma \in \mathbb{N}^* | \forall 0 \leq i < |\sigma| \; d_r(\sigma(i)) = i, \forall 0 \leq j < |\sigma| - 1 \; (\sigma(j), \sigma(j+1)) \in E\}$
T is obviously a tree, since the defining condition holds for a string if it holds for a successor.

Together with the relation $E_T := \{(\sigma, \tau) \in T \times T | \exists k \in \mathbb{N} \; \sigma = \tau \circ k \vee \sigma \circ k = \tau\}$, T then also is a graph-theoretical tree, as we have seen in Lemma 2.2.1.

Now the isomorphism f works as follows: $f \colon G \to T, f(v) := \sigma$ with $\sigma = (r, v_0, \ldots, v_k, v)$ as the unique path from the root r to v.

Because T consists of all paths from the root r to the other nodes, f is well-defined.

And f is obviously injective, since the reverse image is equal to the last entry of the image.

Also, f is surjective, because every path in a tree is unique.

Finally, f is a graph-isomorphism.
Let $(v, w) \in E$ and let $r, v_0, \ldots, v_k, v$ and $r, w_0, \ldots, w_l, w$ be the respective paths from the root. If $v \in \{w_o, \ldots, w_l\}$, then $w_l = v$, because otherwise there would be a shorter path from r to w. Analogously, if $w \in \{v_0, \ldots, v_k\}$, then $v_k = w$. If neither $v \in \{w_o, \ldots, w_l\}$ nor $w \in \{v_0, \ldots, v_k\}$, then $r, v_0, \ldots, v_k, v$ and $r, w_0, \ldots, w_l, w, v$ are different paths from r to v, which contradicts the assumption that G be acyclic.
It follows that $w_l = v$, without loss of generality, and therefore $(f(v), f(w)) = ((r, v_0, \ldots, v_k, v), (r, w_o, \ldots, w_l, w)) \in E_T$.

Reversely, let for two vertices $v, w \in G$, their images be adjacent: $(f(v), f(w)) \in E_T$. Then either $f(w) = (r, v_0, \ldots, v_k, v, w)$ or $f(v) = (r, w_0, \ldots, w_l, w, v)$. Either way, it holds that $(v, w) \in E$. $\square$

In conclusion, we see that a general tree (with root) can be replaced by a tree in the sense of our definition. This is still true, when we shift our interest to computable trees.

Definition 2.2.4. A tree $T \in \mathbb{N}^*$ is called *computable* if it is a computable set.

In the more general case of graph-theoretical trees this translates to graphs $G = (V, E), V \in \mathbb{N}$ with V, E computable sets, because when we consider a computable tree T a graph-theoretic tree, both T and E_T are computable. Consider again the embedding of a - now computable - graph-theoretic tree into $\mathbb{N}^*$ like above. Because paths are unique in any tree, that isomorphism f was in fact effective, in the sense, that one can effectively decide the membership of a node in the isomorphic tree T. Hence, for a computable graph G, its isomorphic image T is also computable. The edge relation E_T is always computable.

At last, we have to make sure that we get the same complexity results when we restrain ourselves to infinite paths from the root through trees instead of abitrary infinite paths through graph-theoretical trees.

To this end, let $(v_0, v_1, \ldots)$ be an infinite path through some graph-theoretic tree G with root r. If $r = v_i$ for some $i \in \mathbb{N}$ then $(v_i, v_{i+1}, \ldots)$ is a path of same Turing-degree, since they differ only in finitely many places. If $r \notin \{v_i | i \in \mathbb{N}\}$, then choose $i \in \mathbb{N}$ such that the path $(r, p_0, \ldots, p_k, v_i)$ from r to v_i is the shortest. Then there is no v_j with $i < j$ on that path, otherwise its path to r would have been shorter. Therefore, $(r, p_k, \ldots, p_0, v_i, v_{i+1}, \ldots)$ is an infinite path and has the same degree as the original path, because they have only different initial segments. Note, that the equivalent path beginning at r can in any case be uniformly computed from the original path.

So let us consider a graph-theoretic tree G with root r, an infinite path through G starting at r, and the isomorphic image of G that is a tree in our sense. Then the corresponding infinite path has the same degree: One can compute the infinite path through G with its image T as an oracle by mapping to the last entry of each of its node. And one can compute the infinite path through T with the reverse image as an oracle by mapping each node to the path from r to that node.

That is why it is legitimate and reasonable to only consider trees in our restrictive sense and to only consider infinite paths with end-point through these trees that begin at the root λ. And from here on, we will denote with *(infinite) path* through a tree only the infinite paths with end-point λ, if not stated otherwise. As one easily observes, for every infinite path through a tree, there is a unique function $x \in \mathbb{N}^{\mathbb{N}}$, such that the path is equal to $(\lambda, x \upharpoonright 1, x \upharpoonright 2, ...)$. Because of this great redundance, we will further identify every infinite path with this corresponding function.

2.3 Definition and characterizations of Π^0_1 classes

Let us capture the previous observations and conventions in an important definition.

Definition 2.3.1. Let T be a tree. Then $[T] := \{f \in \omega^\omega | \forall n \in \omega \; f \upharpoonright n \in T\}$ is the *class of infinite paths* through T.

Definition 2.3.2. $P \subseteq \omega^\omega$ is called a Π^0_1 *class* if there is a computable tree T such that $P = [T]$. If $P \subseteq 2^\omega$, we may speak of a *class of sets*, otherwise of a *class of functions*. If P contains no computable functions, it is a *special* Π^0_1 class.

Remark. The notion of special Π^0_1 class is due to Jockusch and Soare. The paper [JS72a] is denoted to special Π^0_1 class exclusively. Kreisel and Shoenfield historically achieved the first results on Π^0_1 classes, captured in [Sho60], but the pathbreaking works were done by Jockusch and Soare in the 1970s ([JS72a], [JS72b] amongst others).

The following observation explains why Π^0_1 classes are often called *effectively closed sets*. That way, Π^0_1 classes can be considered part of the effective version of the Borel hierarchy of ω^ω. In contrast to the general Borel hierarchy, one uses lightface characters to denote the classifications. That is why the effective version is also called *lightface Borel hierarchy.*

Theorem 2.3.1. *A subset C of $\mathbb{N}^\mathbb{N}$ is closed if and only if $C = [T]$ for a tree $T \subseteq \mathbb{N}^*$.*

Proof. Suppose $C \subseteq \mathbb{N}^\mathbb{N}$ is closed. Then the complement $U := \overline{C}$ relative to $\mathbb{N}^\mathbb{N}$ is open. Because $(I(\sigma))_{\sigma\in\mathbb{N}^*}$ is a basis for the topology of $\mathbb{N}^\mathbb{N}$, there is some $I \subseteq \mathbb{N}^*$, such that $U = \bigcup_{\sigma\in I} I(\sigma)$. Now we can assume, without loss of generality, that I is chosen to be maximal, i.e. for all $\sigma \in \mathbb{N}^*$, if $I(\sigma) \subseteq U$, then $\sigma \in I$. But that means, that I is closed under successors, because $I(\sigma) \subseteq I(\tau)$, if $\tau \prec \sigma$. By contraposition, this shows that the complement $T := \overline{I}$ relative to $\mathbb{N}^*$ is closed under predecessors, i.e. a tree. So then $C = \overline{U} = \overline{\bigcup_{\sigma\in I} I(\sigma)} = \overline{\{x|\ \exists k \in \mathbb{N}\ x \upharpoonright k \in I\}} = \{x|\ \forall k \in \mathbb{N} x \upharpoonright k \in T\} = [T]$.

Now suppose $C = [T]$ for some tree $T \subseteq \mathbb{N}^*$.
Then $\overline{C} = \overline{[T]} = \overline{\{x|\ \forall k \in \mathbb{N}\ x \upharpoonright k \in T\}} = \{x|\ \exists k \in \mathbb{N}\ x \upharpoonright k \notin T\} = \bigcup_{\sigma\notin T} I(\sigma)$, which is clearly open. So C is closed. □

We will now observe that we can vary the complexity of trees involved in the definition of Π^0_1 classes in two directions, i.e. more complex or less complex, while conserving the same notion. That makes it possible to better get a handle on the notion on the one hand, e.g. when trying to introduce an index for Π^0_1 classes. This can be done by an effective enumeration of primitive recursive sets. And on the other hand, that makes it easier to show that some class is a Π^0_1 class. Also, the equivalent notion of a class defined via a formula with one universal quantifier over the natural numbers and a computable matrix finally explains the name *Π^0_1 class.*

For results on index sets for Π^0_1 classes, see for instance [CR03], [CJ99] or chapter VI of [Cen10].

Lemma 2.3.2 ([CR98]). *For a class $P \subseteq \mathbb{N}^{\mathbb{N}}$, the following are equivalent:*

i) *P is a Π^0_1 class*

ii) *$P = [T]$, for some recursive tree T*

iii) *$P = [T]$, for some primitive recursive tree T*

iv) *$P = [T]$, for some Π^0_1 tree T*

v) *$P = \{x|\forall n\ R(n, x)\}$, for some recursive relation R*

Proof. By definition, i) and ii) are equivalent.

Now suppose, $P = [T]$ for some recursive tree T. Then there is a total computable $\{0, 1\}$-valued function ϕ_e such that $c_T \equiv \phi_e$.
Define $S := \{\sigma|\forall n \le |\sigma|\varphi_e(\sigma \restriction n)[|\sigma|] \ne 0\}$, which is a primitive recursive tree. Obviously, all strings in T satisfy the condition to be member of S, so $T \subseteq S$ and hence $[T] \subseteq [S]$. Assume conversely that $x \notin [T]$. Then there is a prefix $x \restriction n \notin T$. So the according computation converges at some time t: $\varphi_e(x \restriction n)[t] \downarrow= 0$. But then $\varphi_e(x \restriction (\max\{n, t\} + 1)) \notin S$, and therefore $x \notin [S]$. This shows the implication ii) $\Rightarrow$ iii), while the converse is obviously true.

Let now $P = [T]$ with T a Π^0_1 tree. So there is a recursive relation R, such that $\sigma \in T \Leftrightarrow \forall n\ R(n, \sigma)$.
Define the set S by $\sigma \in S \Leftrightarrow \forall m, n \le |\sigma|\ R(n, \sigma \restriction m)$. If $\sigma \in T$, then $\forall n\ R(n, \sigma)$, especially $\forall n \le |\sigma|, m = |\sigma|\ R(n, \sigma \restriction m)$. But since T is a tree, it also holds that $\forall n \le |\sigma|, m < |\sigma|\ R(n, \sigma \restriction m)$. So, $\sigma \in S$, and therefore $T \subseteq S$ and $[T] \subseteq [S]$. Now let $x \in [S]$. Then $\forall k x \restriction k \in S$. By definition of S and because $(x \restriction k) \restriction m = x \restriction m$ for $m \le k$, it holds that $\forall k \forall m, n \le k\ R(n, x \restriction m)$. By choosing $k = \max(m, n)$, we get $\forall m, n\ R(n, x \restriction n)$. And finally by choosing $m = n$, $\forall n\ R(n, x \restriction n)$ and hence $x \in [T]$. This shows the nontrivial direction of ii) $\Leftrightarrow$ iv).

Suppose $P = [T]$ for a recursive tree T. Then define the relation R by $R(n, x) \Leftrightarrow x \restriction n \in T$, which is recursive.
Then $x \in [T] \Leftrightarrow \forall n\ x \restriction n \in T \Leftrightarrow \forall n\ R(n, x)$. That shows ii) $\Rightarrow$ v).

Finally, suppose that $P = \{x | \forall n\ R(n,x)\}$ for some recursive relation R. Then there is a recursive functional φ_e such that $R(n,x) \Leftrightarrow \varphi_e^x(n) = 1$ and $\neg R(n,x) \Leftrightarrow \varphi_e^x(n) = 0$.
Then define $T := \{\sigma | \forall n \varphi_e^\sigma(n)[n] \downarrow \rightarrow \varphi_e^\sigma(n) = 1\}$. One easily sees that T is a Π_1^0 tree. If $x \in P$, then obviously $\varphi_e^{x\restriction k}(n) = 1$ for all n, so $P \subseteq [T]$. Now let $x \in [T]$. This means that for all numbers k and n, if $\varphi_e^{x\restriction k}(n)$ converges after n steps, then $\varphi_e^{x\restriction k}(n) = 1$. But since φ_e^x is a total function, there is, for every n, a use u and a time t_n, such that the corresponding computation with oracle $x \restriction u$ with input n halts after t_n many steps. So for all n, $\varphi_e^x(n) = 1$, and therefore $x \in P$. By this argument, v) $\Rightarrow$ iv).

By transitivity, all five statements are equivalent. □

There are other interesting ways of characterizing the Π_1^0 classes. When considering problem classes arising in combinatorics, one often sees that solvability of finite instances extend to solvability of infinite instances by some compactness argument. But these extended results cannot always be effectivized. For example, [Cen10] mentions a 3-colorable, computable, connected graph which has no computable k-coloring for any k.

But solution sets of infinite instances of combinatorial problem classes can often be represented by Π_1^0 classes and the degrees of those solutions can hence be explored with the results of Chapter 4. Some of these problem classes already characterize the (bounded, computably bounded) Π_1^0 classes, in the sense that each (bounded, computably bounded) Π_1^0 class can be represented that way. So, by other results in Chapter 4, the existence of particularly pathological problem instances can be shown. We will define the notion of (computably) bounded Π_1^0 classes at the end of the next chapter.

Our consideration of the two notions of trees can be seen as such a representation result providing a characterization. It was introduced to narrow down our notion of trees and infinite paths to a reasonably simple, but comprehensively representational class. For an overview over other representation results, see [CR98], [CJ99] and Part B of [Cen10].

Finally, like in the Chomsky Hierarchy for formal languages, there are also machine characterization for languages of words of infinite length, so-called ω-languages. The first approach to automata accepting sets of infinite strings was done by Büchi in the 1960s. Later, other authors considered different acceptance modes. There is a treatment in [CR03]. There, the authors find that the class of ω-languages accepted by a strictly computable deterministic automaton via e-acceptance, is exactly the class of Π^0_1 classes. It is also possible to define grammars that generate ω-languages. However, both approaches are not very handy and are only mentioned to offer a more comprehensive view.

3 Notions of boundedness

3.1 The partial order of notions of boundedness for trees

Now let us consider the following notions of boundedness for trees. We will investigate which ones of them are really different and which ones in fact coincide. This will be done for general trees and for computable trees in parallel. That is for economic reasons and because one notion is identical for both general and computable trees, i.e. every tree bounded in the sense of that notion is already computable.

Definition 3.1.1 (Notions of boundedness for trees)**.** Let T be a tree.

i) T is *finitely branching* if there is a function $f\colon \mathbb{N}^* \to \mathbb{N}$, such that for every σ in T,

$$|\{i \mid \sigma \circ i \in T\}| < f(\sigma).$$

T is *computably finitely branching* if f is computable.

ii) T is *bounded* by a function $g\colon \mathbb{N}^* \to \mathbb{N}$ if for every σ in T and every $i \in \mathbb{N}$,

$$\sigma \circ i \in T \Rightarrow i < g(\sigma).$$

T is *computably bounded* if g is computable.

iii) T is *uniformly bounded* by a function $h\colon \mathbb{N} \to \mathbb{N}$ if for all $\sigma \in T$ and all $i < |\sigma|$

$$\sigma(i) < h(i).$$

T is *computably uniformly bounded* if h is computable.

iv) T is *stagewise enumerable* by a function $k\colon \mathbb{N}^* \to \mathbb{N}$ if for each $\sigma \in T$

$$k(\sigma) = \langle i_1, \dots, i_m \rangle,$$

where $i_1, \dots, i_m$ are exactly the successors of σ: $\{i_1, \dots, i_m\} = \{i : \sigma \circ i \in T\}$. T is *computably stagewise enumberable* if k is computable. This function k is unique, if it exists.

Remark. Note that *stagewise enumerability* is stronger than general enumerability, since every tree $T \subseteq \mathbb{N}^*$ is enumerable. Also, stagewise enumerability by a computable function is stronger than computably enumerability. In fact, it is an easy observation that computably stagewise enumerable trees are already computable, which will be part of our following investigation.

Lemma 3.1.1. *For any tree T, the following are equivalent:*

i) T is finitely branching

ii) T is bounded

iii) T is uniformly bounded

iv) T is stagewise enumberable

Proof. Let T be a finitely branching tree. Then, for every $\sigma \in T$, there are only finitely many successors. So there is a function k that maps each node $\sigma \in T$ to the tuple of its successors.

The other directions follow from Lemma 3.1.3 below. □

Almost every tree we will be considering is bounded and infinite. And for infinite bounded trees, the important Kőnig's Lemma applies.

Lemma 3.1.2. *(Kőnig's Lemma) Every infinite, finitely branching tree has an infinite path.*

Proof. Let T be an infinite tree. Then $\lambda \in T$, because T is not empty and closed under initial segments. Define $x(0) := \lambda$.

By Lemma 3.1.1, T is bounded, and therefore there is a function $g : \omega^* \to \omega$, such that if $\sigma \circ i \in T$, then $i < g(\sigma)$, for all $\sigma \in T, i \in \omega$. By applying the pigeonhole principle to the infinitely many successors of $\sigma = \lambda$ and their finitely many mutually distinct initial segments bounded by $g(\sigma)$, there must be at least one number i, such that $(i) \in T$ has infinitely many successors. Define $x(1) := i$ as the least such number i. And then, inductively for each n, define $x(n+1)$ to be the least number i such that T has infinitely many elements with $(x(0), x(1), \ldots, x(n), i)$ as an initial segment. Again, you see that such a number exists, by applying the pigeonhole principle to the infinitely many successors of $\sigma = (x(0), \ldots, x(n))$ that exist by the induction hypothesis and the bound $g(\sigma)$ for the number of possible initial segments of length $n + 1$ extending σ. □

Remark. In this proof of Kőnig's Lemma we use some form of the axiom of choice, but Kőnig's Lemma is strictly weaker than the axiom of choice, in terms of provability in a subsystem of second order arithmetic over RCA_0. Kőnig's Lemma is equivalent to what is called arithmetic comprehension ([Cen10]). RCA_0 together with arithmetic comprehension is called ACA_0. The same statement for infinite trees in 2^ω is called Weak Kőnig's Lemma (WKA_0). For an elaboration of the relations of these axioms to others, see chapter VII about reverse mathematics of [Cen10] or section three of [CJ99].

A recursive version of Kőnig's Lemma stating *Every infinite computably finite branching computable tree has an infinite computable path* does *not* hold. An investigation of Π^0_1 classes without computable members, so-called *special* Π^0_1 classes, will be done in Section 4.2.7.

Remark. In section 4.2 (Basis Theorems), we will revisit the idea of the proof of Kőnig's Lemma and effectivize it, to obtain, amongst others, the Kreisel Basis Theorem.

Lemma 3.1.3. *Let T be any tree.*

If T is stagewise enumerable by a function k, T is also uniformly bounded by a function h, which is computable from k.

If T is uniformly bounded by a function h, it is also bounded by a function g, which is computable from h.
And if T is bounded by a function g, T is also finitely branching by a function f, which is computable from g.

Proof. Let the function $k(\sigma) = \langle i_1, \ldots, i_m \rangle$ enumerate the successors of the nodes of T, define
$h(1) := \max\{i \mid \exists j \in \mathbb{N}\; k(\lambda) = \langle i_1, \ldots, i_m \rangle \wedge 1 \leq j \leq m \wedge i = i_j\}$ and
$h(n+1) := \max(h(n), \max\{i \mid \exists \sigma \in \prod_{k=1}^{n} \{0, \ldots, h(k)\} \exists j \in \mathbb{N}\; k(\sigma) = \langle i_1, \ldots, i_m \rangle \wedge 1 \leq j \leq m \wedge i = i_j\})$ for all $0 < n$. Then T is uniformly bounded by h, by construction, and h is computable from k.

Given a uniform bound h for T, T is bounded by $g(\sigma) := h(|\sigma| + 1)$, which can be computed from h.

Suppose T is bounded by the function g. Then T is also finitely branching with the bound $f(\sigma) := g(\sigma) + 1$ computable from g. □

Remark. Note that Lemma 3.1.1 does not hold for computable bounding functions. In fact, for arbitrary trees, all these effective boundedness notions are strictly different and stronger than noncomputable boundedness.

Lemma 3.1.4. *There is a tree that is finitely branching, but not computably finitely branching.*
There is a tree that is computably finitely branching, but not computably bounded.
There is a tree that is computably bounded, but not computably uniformly bounded.
There is a tree that is computably uniformly bounded, but not computably stagewise enumerable.

Proof. Let $K = \{e : \varphi_e(e) \downarrow\} = \{k_0, k_1, ...\}$ be the diagonal halting problem with $k_i < k_{i+1}$ for every i. This is not a computable enumeration, of course.

$T_1 := \{\lambda\} \cup \{(t_0, \ldots, t_n) \mid \forall\, 0 \leq i \leq n\; (\varphi_{k_i}(k_i)[t_j] \uparrow \text{ or } t_j = 0)\}$
This tree is finitely branching, but not computably finitely branching. Suppose, f

was a computable function with $|\{i \mid \sigma \circ i \in T_1\}| < f(\sigma)$ for every $\sigma \in T_1$. Compute $t(e) := \max\{f(0^j) : 0 \leq j \leq e\}$. Since e is not larger than the $(e+1)$th index $k_{e+1} \in K$ in the halting problem, $e \in K$ iff $\varphi_e(e)[t(e)] \downarrow$, which is computable. This would be a contradiction and so f cannot be computable.

$T_2 := \{0\}^* \cup \{0^e \circ (s+1) \mid \varphi_e(e)[s] \downarrow \& \varphi_e(e)[s-1] \uparrow\}$
The tree T_2 is effectively finitely branching ($f \equiv 2$), but not computably bounded. Suppose, there were a computable function g such that $i < g(\sigma)$ if $\sigma \circ i \in T_2$, one could compute the halting problem: $K = \{e \mid \varphi_e(e)[g(0^e)] \downarrow\}$. This tree is also computable, which will be relevant in the upcoming Lemma 3.1.6.

To define the next example, we first define the following:
$T_K := \{(k_0, \dots, k_i) \mid i \in \mathbb{N}\} \cup \{\lambda\}$
$T_S := \{(s_0, \dots, s_i) \mid i \in \mathbb{N} \;\&\; \forall\, 0 \leq j \leq i\ \varphi_{k_j}(k_j)[s_j] \downarrow \;\&\; \varphi_{k_j}(k_j)[s_j - 1] \uparrow\} \cup \{\lambda\}$
Let p be a computable function s.t. $\varphi_e \equiv \varphi_{p(e)}$ and $p(e) > e$; i.e. a padding function.

Then define
$T_3 := \{\rho \in \omega^* \mid \sigma \in T_k \wedge \tau \in T_S \wedge \forall 0 \leq i < |\sigma| = \lceil \frac{|\rho|}{2} \rceil\ (\rho(2i) = \sigma(i)) \wedge \forall 0 \leq j < |\tau| = \lfloor \frac{|\tau|}{2} \rfloor\ (\rho(2j+1) = \tau(j))\}$.
This tree is computably bounded, but not computably uniformly bounded. Any string in T_3 consists of the indices in the halting problem, ordered by size, and the times by which their respective computations halt, alternately. This is a computable bound for T_3:

$$g(\sigma) := \begin{cases} p^{[\frac{|\sigma|}{2}]}(e_0) + 1 & \text{, if } |\sigma| \text{ even} \\ \min\{s : \varphi_{\sigma(|\sigma|-2)}(\sigma(|\sigma|-2))[s] \downarrow\} + 1 & \text{, if } |\sigma| \text{ odd,} \end{cases}$$

where $p^{[n]} = \underbrace{p \circ \cdots \circ p}_{n \text{ times}}$ denotes the nth composition of p with itself. For the verification, observe that $p^{[i]}(e_0)$ is in fact greater or equal than e_i. Because for every $e \in K$, also $p(e) \in K$. Further $e < p(e)$. But e_{i+1} is defined as the lowest such number, i.e. both in K and greater than e_i. For strings of odd length the bound works obviously.
But T_3 is not computably uniformly bounded. Suppose that T was computably uniformly bounded by h. Then you would be able to derive from it an effective decision method for the halting problem. If $e \in K$, then it is among the first $e+1$

indices $k_0, \dots k_e \in K$. To decide whether $e \in K$, it is hence sufficient to compute $s(e) := \max\{h(2i+1) \mid 0 \leq i \leq e\}$, because then $K = \{e \mid \varphi_e(e)[s(e)]\downarrow\}$. So the halting problem would be computable, which is a contradiction. So there is no uniform computable bound h for T_3.

And finally, T_K is an example of a computably uniformly bounded tree that is not computably stagewise enumerable. It is computably bounded, as can be argued similarly to above. And since there is exactly one string in T_K of any given length, it is obviously also computably uniformly bounded. A computable uniform bound is, for example, $h(i) := p^{[i]}(e_0) + 1$. But it is an easy observation that every tree, that is computably stagewise enumerable, is already computable. Let T be a tree that is computably stagewise enumerable by the function k. Then clearly $\sigma \in T$ iff $\sigma(i)$ is a number encoded in $k(\sigma \upharpoonright i)$, which is computable. Thus, since T_K is not computable, it cannot be computably stagewise enumerable. □

Remark. Some of these separation results are easy corollaries of theorems in Chapter 4, but we choose to do our investigation in a bottom-up manner. Also, one might find the employed miniature examples illustrative.

Now let us consider computable trees. As with arbitrary trees, the notion of computably finitely branching is stronger than that of general finitely branching.

Lemma 3.1.5. *There is a computable tree, which is finitely branching, but not computably finitely branching.*

Proof. Consider $T := \{0\}^* \cup \{0^e \circ (t+1) \mid \varphi_e(e)[t]\downarrow \& \varphi_e(e)[\lfloor t/2 \rfloor]\uparrow\}$.

T is computable and is bounded by a function g, because for every index e in the halting problem, there are only finitely many steps, for which $\varphi_e(e)$ does not converge. But T is not effectively finitely branching. Suppose, f would give a computable bound on the number of successors for 0^e in T. If $\varphi_e(e)$ converges first at step s, then $0^e \circ t \in T$ for all $s < t < 2s-1$. So $f(0^e) > 2s-1-s-1 = s-2$. Then one would be able to decide whether $e \in K$, depending on $\varphi_e(e)[f(0^e)+1]\downarrow$. □

Apart from that, the situation with computable trees is different:

Lemma 3.1.6. *For computable trees, the notions of computably stagewise enumerability, computably uniformly boundedness and computably boundedness coincide. The notion of computably finite branching is still properly weaker with respect to computable trees.*

Proof. Let T be a computable tree. Because of Lemma 3.1.3, for the first statement it suffices to show that you can effectively enumerate the children of a node of T, given a computable bound for T. Let g be such a bound and $\sigma \in T$. Simply decide for every $i < g(\sigma)$ whether $\sigma \circ i \in T$. This can effectively be done since T and g are computable. So it is obvious that the successor function, which enumerates the children of a given node of T, is computable.

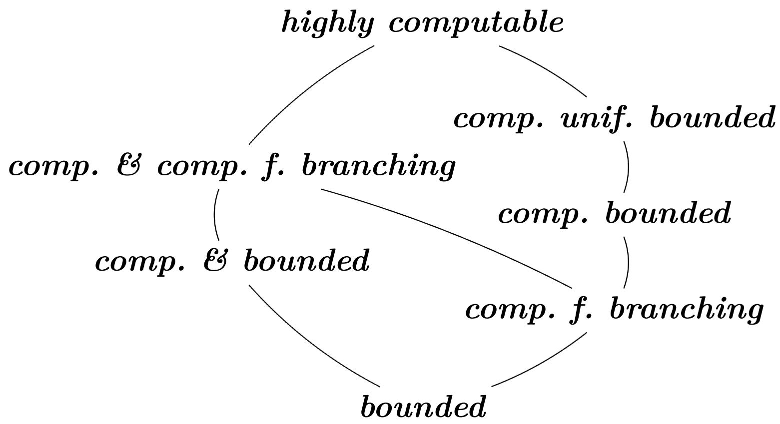
To see that the class of computable trees that are computably finitely branching is strictly greater, consider example T_2 from Lemma 3.1.4. That tree T_2 was computable although this was not required there. □

Definition 3.1.2. The class of computable trees that are computably bounded is called *highly computable trees*.

Since all of the effective versions of boundedness notions (except for computably finitely branching) coincide for computable trees, these notions indicate exactly the highly computable trees.

Combining the above results about boundedness notions we get the partial order of boundedness notions for trees.

Corollary 3.1.7. *The* 16 *formal definitions using computable/noncomputable trees and the effective or general version of the boundedness definitions collapse into 7 mutually distinct notions as follows:*



In this graphical representation of the partial order of notions of boundedness, exactly the shown inclusions and their transitive closure are valid. Every edge directed generally upwards indicates an inclusion.

Proof. Here is an overview how this corollary follows from the above lemmas: First of all, there are three notions which represent four equivalent notions, respectively. That is, *boundedness* is equivalent to the notion *finitely branching* and is equivalent to *uniformly boundedness* and is equivalent to *stagewise enumerability*. This holds by Lemma 3.1.1 for general trees on the one hand and computable trees on the other. Further, the class of highly computable trees can be equivalently characterized as the class of trees that are either computably stagewise enumerable (and hence automatically computable as in the proof of Lemma 3.1.4) or computable and either computably uniformly bounded or computably bounded. This follows from the Lemmata 3.1.3 and 3.1.6.

Then there are $16 - 3 - 3 - 3 = 7$ remaining classes that we see in the figure. So, there are $\binom{7}{2} \cdot 2 = 7 \cdot 6 = 42$ possible inclusive relations between classes. We will now account for the validity of the respective implications.

There are 8 implications displayed in the figure. There are 8 other implications that follow via transitivity and therefore need not be considered. Three of the shown implications follow by mere formal restriction, that is: Every computable bounded tree is bounded; every computable and computably finitely branching tree is computably finitely branching; and every computable and computably uniformly bounded tree (i.e., highly computable tree) is computably uniformly bounded. Another two follow by formal restriction when applying Lemma 3.1.1: Every computable tree that is computably finitely branching is computable and bounded and every computably finitely branching tree is bounded.

Lemma 3.1.3 gives us the remaining three valid implications. Every highly computable tree is computable and computably finitely branching; every computably uniformly bounded tree is computably bounded; and every computably bounded tree is computably finitely branching.

Now for the $42 - 16 = 26$ invalid implications.

By considering the tree of all initial segments of an noncomputable string in 2^ω, one sees that some computably uniformly bounded trees are not computable. And by transitivity and contraposition, this shows that neither of the four classes of computably uniformly bounded, computably bounded, computably finitely branching or bounded trees is a generalization of any of the three different notions for computable trees. That shows that 12 implications are not valid.

Lemma 3.1.4 separates the four notions for general trees. Note, that the most restrictive one, computably stagewise enumerability, already implies that the tree is computable and hence highly computable. This gives us proof for the invalidity of four implications directly, one of which we already have proved in the last paragraph. And for four other implications, the invalidity follows by transitivity and contraposition from Lemma 3.1.4, one of which has already been accounted for. So, Lemma 3.1.4 shows that another 6 implications do not hold.

From Lemma 3.1.6, we know that there are computable trees that are computably finitely branching, but not computably bounded. That shows that the class of computable trees that are computably finitely branching is neither a generalization of the highly computable trees nor of the classes of general trees with an effective boundedness of some kind other than computably finitely branching. And then, by transitivity and contraposition, the same holds for computable, bounded trees. These are immediate proofs for the invalidity of 6 more implications, that have still been unaccounted for.

Finally, by Lemma 3.1.5, we know that there are trees that are computable and bounded, but not computably bounded, which shows the invalidity of the two remaining implications. □

3.2 Considering boundedness for Π^0_1 classes

Now, that we understand which different boundedness notions there really are, we focus on computable trees, since they can serve to define our objects of interest, the Π^0_1 classes.

Definition 3.2.1. If $P = [T]$ for some highly computable tree T, P is a *computably bounded* Π^0_1 class.
Analogously, we may speak of *bounded* or *computably finitely branching* Π^0_1 classes.

Remark. Here lies the source of a possible misunderstanding due to linguistic ambiguity. This definition of *computably bounded* and *bounded* only applys to Π^0_1 class, as it relates to the trees generating the elements of the class. This is quite obvious in the case of *computably finitely branching*. Out of the context of trees, *bounded* or *computably bounded* may be mistaken as a notion relating to the members of the class themselves only. That is, they would be uniformly bounded by some function. In the case of computably boundedness, these two notions even coincide. The members of a computably bounded Π^0_1 class are obviously uniformly bounded by a computable function. And conversely, one can add to the definition of a computable tree a constraint that bounds the strings by a computable function. That disguises only the subtle difference of the two notions. In the case of boundedness, the notion unrelated to trees is properly weaker. In [Cen10], the author calls it *topological boundedness* and gives an example separating the two notions.

We find, that the three different boundedness notions for computable trees do not collapse into one or two classes of Π^0_1 classes when we look at the infinite paths through the trees in these classes. Let us rephrase this in the following Lemma.

Lemma 3.2.1. *There is a bounded Π^0_1 class that is not a computably finitely branching Π^0_1 class.*
There is a computably finitely branching Π^0_1 class that is not a computably bounded Π^0_1 class.

Proof. For the first part, consider this tree, which is a variant of that in Lemma 3.1.5:
$T := \{\sigma : \forall i < |\sigma| \ \sigma(i) = 0 \vee (\sigma(i) = t+1 \wedge \varphi_i(i)[t] \downarrow \wedge \varphi_i(i)[\lfloor t/2 \rfloor] \uparrow)\}$
Obviously, T is computable and bounded. Suppose now, there is a computable tree $\tilde{T}$, that is finitely branching with the bound f, so that $[T] = [\tilde{T}]$. The trees T and $\tilde{T}$ have a common subtree $Ext(T)$ of the initial segments of $[T]$. But that tree is equal to T, because every node in T is extendible. Hence, one can apply the argument of Lemma 3.1.5 to show that one can compute K from f. Therefore f cannot be computable.

For the second part, consider a slight variation of T_2 in Lemma 3.1.4:
$T := \{\sigma : \forall i < |\sigma| \ \sigma(i) = 0 \vee (\sigma(i) = t \wedge \varphi_i(i)[t] \downarrow \wedge \varphi_i(i)[t-1] \uparrow)\}$ Similar to above, a computable bound g on the children of the nodes of T especially works for the tree of infinite paths through T, such as this one:
$$x_0(e) = \begin{cases} 0 & , e \notin K \\ t & , e \in K, \ \varphi_e(e)[t] \downarrow \ \& \ \varphi_e(e)[t-1] \uparrow \end{cases}$$
So $e \in K$ iff $\varphi_e(e)[g(e-1)] \downarrow$. Hence g cannot be computable. □

Remark. Note that the separation of computably bounded Π^0_1 classes from bounded Π^0_1 classes is implied by the upcomping Theorem 4.2.4 in combination with the Kreisel Basis Theorem (4.2.3). The separation of bounded Π^0_1 class from computably finitely branching Π^0_1 classes, however, cannot be done in such a way, because we will find in Corollary 4.1.8 that every computably finitely branching Π^0_1 class has the same degree spectrum as some bounded Π^0_1 class. That means, when reduced to their degree spectra, the three notions of boundedness for Π^0_1 classes do collapse into two; bounded and computably bounded Π^0_1 classes.

A very natural and common example for computably bounded Π^0_1 classes are classes of *separating sets*.

Definition 3.2.2. Let A, B be infinite c.e. sets, that are mutually disjoint. $S(A, B) = \{C \subseteq \mathbb{N} | A \subseteq C, B \cap C = \emptyset\}$ is the class of *separating sets* for the pair A, B. When $S(A, B)$ contains no computable set, the pair A, B is called *recursively inseparable.*

Note that there is also a definition of *effectively inseparable sets*, which is properly stronger. See [Odi92] for details.

The following statement is to be understood by identifying sets with their characteristic functions.

Proposition 3.2.2. *Let A, B be infinite c.e. sets, that are mutually disjoint. Then the class of separating sets $S(A, B)$ is a nonempty computably bounded Π^0_1 class.*

Proof. Because A and B are disjoint, $A \in S(A, B)$ and so $S(A, B)$ is nonempty.

Since A and B are c.e. sets, there are partial computable functions with indices e_A and e_B such that A and B coincide with their respective domains, i.e. $A = W_{e_a}, A = W_{e_B}$.

Define the set
$T := \{\sigma \in 2^* \mid \forall 0 \leq i < |\sigma| \ ((\varphi_{e_A}(i)[|\sigma|] = 0 \rightarrow \sigma(i) = 1) \wedge (\varphi_{e_B}(i)[|\sigma|] = 1 \rightarrow \sigma(i) = 0))\}$.

This set T is a tree, because for longer strings, only longer computations are considered, and once a computation yields a result, this result is fixed. This tree is obviously computable. So $[T]$ is a Π^0_1 class bounded by the computable function $g \equiv 2$.

And finally, any infinite path through T has to satisfy all requirements that its finite initial segments meet. So, if $x \in [T]$, then $i \in W_{e_A} = A$ implies $x(i) = 1$. And $i \in W_{e_B} = B$ implies $x(i) = 0$. So there is a separating set S such that $x = c_S$. And since there are no further constraints on the paths, the characteristic function to every separating set in $S(A, B)$ is in $[T]$. $\square$

4 Degree spectra of Π^0_1 classes

In this chapter we want to take a general look at the degree spectra of certain kinds of Π^0_1 classes. The *degree spectrum* of a class of functions is the class of degrees of elements of the class; that is T-degrees in our context. As we consider Π^0_1 class in this thesis, we might also speak in short of the degree spectrum of a tree. Thereby, we refer to the degree spectrum of the class of infinite paths through that tree.

In computability theory, it is common practice to regard a set or function as a representative of a degree, because one is interested in its complexity, not its specific form. Similarly, we wish to consider Π^0_1 classes mere representatives of their respective degree spectra. Note, that in articles like [KL10], where this notion is actually defined and not just implied, degree spectra are a collection of degrees without multiplicity. This is reasonable when discussing basis results. But that also means that you can only infer little information about the cardinality of a Π^0_1 class by its degree spectrum; see one result in section 4.2.7. However, it is impossible to separate, say, a Π^0_1 class with a single element, which is computable, from a Π^0_1 class consisting of computable functions with any other countable cardinality.

The first section can be thought of as an ansatz to find, for a given Π^0_1 class, a somewhat nicer Π^0_1 class representing the same degree spectrum. Jockusch and Soare have coined the term *degree isomorphic* for the concept of Π^0_1 classes with the same degree spectrum. Then, in section two, we shed light on the bounding conditions for degree spectra. That is, what kinds of degrees can always be found in the degree spectrum of some kind of Π^0_1 class and which cannot.

For a treatment of the structure of degree spectra of Π^0_1 classes in 2^ω ordered by inclusion, see [KL10] and [Cen10]. The authors prove, amongst other results, that the partial order of degree spectra of computably bounded Π^0_1 classes, ordered by inclusion, is a lattice. Note that the observation, that this structure is an uppersemilattice is trivial. For computable trees $T, \tilde{T}$ bounded by computable functions $g, \tilde{g}$, respectively, the tree $\{\lambda\} \cup 0 \circ T \cup 1 \circ \tilde{T}$ is bounded by the function $g_\cup(1 \circ \sigma) := g(\sigma), g_\cup(1 \circ \sigma) := \tilde{g}(\sigma), g_\cup(\lambda) := 2$ and has a path of same degree for any member in the union $[T] \cup [\tilde{T}]$.

4.1 Reduction of Π^0_1 classes with respect to degree spectra

The aim will be to construct, for a given Π^0_1 class of a certain kind, a degree-preserving bijection to a different Π^0_1 class of a nicer kind representing the same degree spectrum. Sometimes we will only be able to construct an embedding which is not onto. But this will still be useful to generalize results in the upcoming section 4.2 about basis theorems.

Theorem 4.1.1. *Let $T \subseteq \omega^\omega$ be an abitrary tree, $P = [T]$. Then there exists a tree $\tilde{T} \subseteq 2^*$ and an effective, degree-preserving functional $S : P \to \tilde{P} = [\tilde{T}]$ such that:*
$\tilde{T}$ is computably enumerable from T and S is a one-to-one correspondence between the noncomputable paths in P and $\tilde{P}$.
If T is bounded by a function g, $\tilde{T}$ is even computable from the join of T and g, and S is even a one-to-one correspondence between all of P and $\tilde{P}$.

Proof. To define $\tilde{T}$, we use the function f that gives us the rest of a division by 2:

$$f : \mathbb{N} \to \{0,1\}, f(n) = \begin{cases} 0 & , n \equiv 0(2) \\ 1 & , n \equiv 1(2), \end{cases}$$

and a function s, which encodes nodes of T:

$s : \omega^* \to \{0,1\}^*$,
$\sigma \mapsto 0^{\sigma(0)+1} \circ 1^{\sigma(1)+1} \circ \cdots \circ f(i)^{\sigma(i)+1} \circ \cdots \circ f(|\sigma|-1)^{\sigma(|\sigma|-1)+1}$

Then $\tilde{T}$ consists of all the images of nodes in T, and prefixes of these, i.e. $\tilde{T} := Pref(S(T))$

Obviously, $\tilde{T}$ is closed under initial segments, i.e., a tree. Since it is sufficient for a node τ to have one witness $\sigma \in T$ to be a member of $\tilde{T}$, it follows that $\tilde{T}$ is computably enumerable in T.

To develop a complete decision procedure for the membership of a string τ in $\tilde{T}$, we have to make two observations.

Firstly, when τ has got a witness σ for its membership in $\tilde{T}$, it has a minimal one. The reason for this is, that any witness σ of τ remains a witness for τ if you extend it by one number i to $\sigma \circ i$.

Secondly, if there is any witness, the minimal ones are identical in length, and may only differ in their last entry. This is due to the manner in which t encodes nodes: If τ is of the form $1 \circ \rho$, it cannot be in $\tilde{T}$, because all of these nodes begin with a zero, except for λ. And if τ is of the form $0^{i_1+1} \circ 1^{i_2+1} \circ 0^{i_3+1} \circ \cdots \circ b^{i_k+1}$ for $b \in \{0, 1\}, k \in \mathbb{N}$ and $i_1, \ldots, i_k \in \mathbb{N}$, any possible witness for τ obviously would begin with $i_1 i_2 \ldots i_{k-1}$ and its next entry would need to be greater or equal than i_k.

Taking this in consideration, one can easily see, that one simply has to exhaust all nodes of T of length $k+1$ with that given prefix. And for finitely branching T, there are only finitely many such nodes,. Then one can give a negative answer for the membership of τ in $\tilde{T}$, when all of them fail to be witnesses for τ.

So if T is finitely branching, $\tilde{T}$ is computable in the join of T and its boundary function g.

Now let us look at the injective functional between the infinite paths through T and those through $\tilde{T}$.

By extending the function s to the domain of infinite strings with a range in the infinite binary strings, one gets the desired functional. This works in the obvious way:

$S : \omega^\omega \to [\tilde{T}] \subseteq \{0,1\}^\omega$,
$\sigma \mapsto 0^{\sigma(0)+1} \circ 1^{\sigma(1)+1} \circ \cdots \circ f(i)^{\sigma(i)+1} \circ \ldots$

The functional S is well-defined, since s is order-preserving in the sense, that the prefix relation is respected by s. That way, for any infinite path x through T, s maps all of its finite initial segments to such finite strings, that there is at most one of any length and one is the prefix of another, if and only if it is shorter than the other. So these segments result in an infinite path through $\tilde{T}$, which is exactly $S(x)$.

S is injective, as one easily checks. If x and $\tilde{x}$ are different infinite paths through T, they differ at a least position n; $x(n) < \tilde{x}(n)$, without loss of generality. But then the nth change of ones and zeros in $S(\tilde{x})$ happens later than in $S(x)$. So, $S(x) \neq S(\tilde{x})$.

Also, S is a computable functional. That is, you can uniformly compute an infinite path through $\tilde{T}$, given an infinite path through T as an oracle. This is obvious from the definition of S.

And not only can you compute $S(x)$ uniformly from x, for a given image $y \in S([T])$ you can also compute its reverse image uniformly. Therefore, S is degree-preserving.

And lastly, let us investigate under which circumstances S would map $[T]$ onto $[\tilde{T}]$.

An infinite path through T is mapped by S to a path through 2^*. More precisely, its image is a binary path, which changes from zero to one infinitely often (and vice versa). But there may still be paths through $\tilde{T}$ that do not change forever, but become constant eventually. Consider such a path $y = \rho \circ b^\omega$, with $b \in \{0,1\}$ and $\rho(|\rho| - 1) \neq 1 - b$, without loss of generality. This is the case if and only if there are $\tau \in \tilde{T} : \tau = \rho \circ 0^i$ for all $i \in \mathbb{N}$. And because $\tilde{T} = \{\tau \in 2^* | \exists \sigma \in T : \tau \prec s(\sigma)\}$, this holds iff there are $\sigma = \theta \circ k$ with $s(\theta) = \rho$ and $k >= i$ for all $n \in \mathbb{N}$. In other words, there are infinite paths through $\tilde{T}$ that are eventually constant, iff some node in T branches infinitely wide. So there are no infinite paths through $\tilde{T}$ that are not being mapped to by S, iff T is finitely branching.

So S is a one-to-one correspondence between $[T]$ and $[\tilde{T}]$ if and only if T is finitely branching.

However, in the general case, the relation S is still a one-to-one correspondence between the noncomputable infinite paths through the two trees, since any eventually constant path is computable. $\square$

Corollary 4.1.2. *If T is a tree which is highly computable in some degree $\mathbf{d}$, then it has the same degree spectrum as some tree $\tilde{T} \subseteq 2^*$ computable in $\mathbf{d}$. There is even an effective degree-preserving one-to-one correspondence between the respective infinite paths.*

Corollary 4.1.3. *If T is a highly computable tree, then it has the same degree spectrum as some computable tree $\tilde{T} \subseteq 2^*$. There is even an effective degree-preserving one-to-one correspondence between the respective infinite paths.*

In [JS72a], Jockusch and Soare show the following result on special Π^0_1 classes.

Theorem 4.1.4 ([JS72a])**.** *For any special Π^0_1 class P, there is a special computably bounded Π^0_1 class $\tilde{P}$ such that for every member $f \in P$ there is a member $\tilde{f} \in \tilde{P}$ with $f \equiv_T \tilde{f}$.*

Theorem 4.1.5 ([JLR91])**.** *For every tree T that is highly computable in $\mathbf{0}'$ there exists a computably finitely branching computable tree $\tilde{T}$ such that there is an effective, degree-preserving 1:1 correspondence between $[T]$ and $[\tilde{T}]$.*

Proof. At first, we use Corollary 4.1.2 to reduce the general case to trees in 2^* that are computable in $\mathbf{0}'$. So, let $T \subseteq 2^*$ be a tree computable in $\mathbf{0}'$.

Since T is computable in $\mathbf{0}'$, its total characteristic function c_T is computable in $\mathbf{0}'$. By the Shoenfield Limit Lemma (see [Odi92], e.g.) we know there exists a total computable function $g : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ such that $\lim_{s\to\infty} g(x, s)$ exists for all $x \in \mathbb{N}$ and is equal to the characteristic function of T: $\forall x \in \mathbb{N}\ c_T(x) = \lim_{s\to\infty} g(x, s)$. Based on that, we define the computable function

$$\tilde{g}(x, s) = \begin{cases} \prod_{\tau \prec \sigma} g(\langle\tau\rangle, s), & x = \langle\sigma\rangle \text{ for some } \sigma \in 2^*, \\ 0, & \text{otherwise} \end{cases}$$

By construction, $\tilde{g}(\cdot, s)$ is the total characteristic function of a tree $T_s \subseteq 2^*$, for every s. We now show, that it has also the limit property that g has, characterized above. In a formula, that is
$\forall x \in \mathbb{N}\ \exists s(x) \in \mathbb{N}\ \forall s \in \mathbb{N}\ s > s(x) \rightarrow c_T(x) = g(x, s)$
By defining $\tilde{s}(x) := \begin{cases} \max\{s(\langle\tau\rangle) \mid \tau \prec \sigma\}, & x = \langle\sigma\rangle \text{ for some } \sigma \in 2^* \\ 0, & \text{otherwise} \end{cases}$
we get the analogous result for $\tilde{g}$:
$\forall x \in \mathbb{N}\ \forall s \in \mathbb{N}\ s > \tilde{s}(x) \rightarrow c_T(x) = g(x, s)$.
The reason for this is, that for s above the threshold $\tilde{s}(\langle\sigma\rangle)$, for every input $y = \langle\tau\rangle$ with $\tau \prec \sigma$ all the computations $g(y, s)$ must return the same value. That is, also the value of $\prod_{\tau \prec \sigma} g(\langle\tau\rangle, s)$ becomes constant.

The limit value $\lim_{s\to\infty} \tilde{g}(x, s)$ is therefore everywhere defined. And it equals that of g, because for whatever input $x = \langle\sigma\rangle$ the value of $g(x, s)$ becomes constantly 0, the product $\prod_{\tau \prec \sigma} g(\langle\tau\rangle, s)$ also becomes 0, at the latest at the same point in time. And wherever the value of $g(x, s)$ becomes eventually 1, the product $\prod_{\tau \prec \sigma} g(\langle\tau\rangle, s)$ also becomes 1, because the limit of g is the characteristic function of a tree. So we can assume, without loss of generality, that $g(x, s)$ is the characteristic function of a computable tree in 2^*, say T_s, for every s.

Now, we define the tree $\tilde{T}$:
$\tilde{T} := \{\tau \in \omega^* \mid \forall n \in \mathbb{N}\ \forall i \in \mathbb{N}\ n < |\tau| \rightarrow ((2 \mid n \rightarrow \tau(n) \in \{0, 1\})$
$\wedge (2 \mid n \rightarrow (n < \tau(n)\ \wedge g(\langle\tau(0), \tau(2), \ldots, \tau(n-1)\rangle, \tau(n)) = 1 \wedge (n \leq i < \tau(n) \rightarrow$
$g(\langle\tau(0), \tau(2), \ldots, \tau(n-1)\rangle, i) = 0))))\}$

In words, $\tilde{T}$ consists of all the nodes which have arbitrary values in $\{0, 1\}$ at even positions and at odd positions $n = 2i + 1$ they have the first number y bigger than i so that the values at the first $i + 1$ even positions $0, 2, 4, \ldots, 2i$ form a node of the tree T_y. The relation that defines $\tilde{T}$ is obviously computable. Since the relation puts only restraints on the values at odd positions which depend exclusively on the values at lower positions, and lets the other values be arbitrary in $\{0, 1\}$, $\tilde{T}$ is closed under initial segments and is hence a computable tree.

Also, every node has at most two successors, since at odd positions the value is uniquely defined by the values at the positions below, and at even positions, only the values 0 and 1 occur. So, $\tilde{T}$ is computably finitely branching with the constant upper bound 3 on the number of successors of nodes.

We now define the one-to-one correspondence between T and $\tilde{T}$ as the computable functional $S : \omega^\omega \to \omega^\omega, S(x)(i) := x(2i)$, hence mapping an infinite path through $\tilde{T}$ to the sequence of values at the even positions.

Note that S maps to 2^ω since any node in $\tilde{T}$ has only values in $\{0, 1\}$ at even positions. In fact, S maps infinite paths through $\tilde{T}$ to infinite paths through T. To understand that, one has to observe that paths through $\tilde{T}$ have the property, that every value at an odd position shows that the binary string of values at even positions below form a node of some T_s. Let us call the binary string of values at even positions of a string the *even part*. Considering the prefix $x \upharpoonright 2i + 2$ of an infinite path x through $\tilde{T}$, we then see that the even part of $x \upharpoonright 2i + 2$ is a node on $T_{x(2i+1)}$. By the definition of $\tilde{T}$, $x(2i + 1)$ has the unbounded linear function $n \mapsto n$ as a lower bound. Since the characteristic functions $c_{T_s} = g(\cdot, s)$ of the trees T_s converge for all nodes, the even part of τ is in every tree T_s with $s > s_0$ for some s_0. This means, the even part of τ is in T itself. Since this holds for all initial segments τ of x, all initial segments of the even part of x, i.e. $S(x)$, are in T and hence $S(x)$ is an infinite path through T.

Now let us consider the inverse relation. That is, show how to effectively reconstruct an inverse image for some infinite path y through T and proving that it is unique, thereby showing that S is in fact onto, one-to-one, and degree-preserving.

Let y be an infinite path through T. Then every inverse image x would have to start with $y(0)$. The next value would be uniquely determined by that. It is the least number $s_1 > 0$ such that $(y(0))$ is a node of the tree T_{s_1}. For all values at higher positions n, x is uniquely determined as well. If n is even, $n = 2i$, every inverse image x must satisfy $x(2i) = y(i)$. If n is odd, $n = 2i+1$, $x(n)$ is bound to be the least number greater than i such that $(y(0), y(1), y(2), \ldots, y(i))$ is a node of the tree $T_{x(n)}$. So, an infinite path x that meets these requirements exists, is uniquely defined by y and uniformly computable in y. □

The inverse also holds. For this, we do not even have to construct any new tree and consequently no computable correspondence:

Lemma 4.1.6. *Every finitely branching computable tree T is highly computable in $\mathbf{0}'$.*

Proof. Every computable tree is computable in $\mathbf{0}'$. So it is sufficient to show that for a computable tree T, which is bounded by g, there is a function $\tilde{g}$ computable in $\mathbf{0}'$, so that T is also bounded by $\tilde{g}$.

Let $g : \omega^* \to \omega$ be any function such that for all $\sigma \in T$ it holds $\sigma \circ i \in T \Rightarrow i < g(\sigma)$.

Consider the following algorithm `IsBoundBy`:

```
IsBoundBy(σ,m):
For i = m, m+1, ...:
Check whether σ ∘ i ∈ T. If it does not hold, continue.
If it does hold, exit the loop and return 1.
```

Note that this algorithm halts always never. Now fix a $\sigma \in \omega^*$ and consider the following algorithm which may ask the halting problem H.

```
Bound(σ):
If σ ∉ T, stop and return 0, otherwise continue.
For m = 1, 2, ...:
Ask H whether IsBoundBy(σ,m) converges. If it does, continue.
Else, exit the loop and return m.
```

`Bound` does always halt eventually, because T is finitely branching. It computes a function $\tilde{g}$ relative to $\mathbf{0}'$. This is a bound for T, because if $\sigma \in T$ and $\tilde{g}(\sigma) = m$, it means that `Bound` must have halted at stage m of its loop, which means that `IsBoundBy`(σ, m) does not halt. This means that there is no number n greater or equal than m so that $\sigma \circ n \in T$. In fact, $\tilde{g}$ is even the minimal bound by construction. □

Equivalently, but less instructively, you can prove this in the following way:

Proof. By defining $R(\sigma, m) :\Leftrightarrow \forall n \geq m : \sigma \circ n \notin T \; \forall m' \in \omega(\forall n' \geq m' : \sigma \circ n' \notin T) \rightarrow m \leq m'$ you have a Π^0_1 relation, which turns to be a well-defined function: $\forall \sigma \in \omega^* \exists! m \in \omega : R(\sigma, m)$, because T is bounded, so there is always an upper bound, and the second condition rules out all but the lowest one. Call this function $\tilde{g}$. By Post's Theorem, $\tilde{g}$ is computable in $\mathbf{0}'$. □

Combining Corollary 4.1.2 with Lemma 4.1.6, we get the following result.

Corollary 4.1.7. *Every finitely branching computable tree has the same degree spectrum as some tree in* 2^* *that is computable in* $\mathbf{0}'$*.*

Finally, by combining Theorem 4.1.5 with Lemma 4.1.6, we get the following.

Corollary 4.1.8. *Every finitely branching* Π^0_1 *class has the same degree spectrum as some computably finitely branching* Π^0_1 *class.*

That means, that while there are properly more computably finitely branching Π^0_1 classes than bounded Π^0_1 classes, as pointed out in Lemma 3.2.1, they do not yield any new degree spectra.

In a nutshell, Corollaries 3.1.7 and 4.1.8 tell us, that it is sufficient to study bounded and computably bounded Π^0_1 class, as long as we are only interested in degree spectra of infinite paths through trees bounded in any sense defined in Chapter 3. And by Corollary 4.1.2 and Lemma 4.1.6, this is equivalent to studying infinite paths through trees in 2^* that are computable in $\mathbf{0}'$ or computable, respectively.

Therefore, many results about degree spectra for computably bounded Π^0_1 classes can be transformed into similar results about degree spectra of bounded Π^0_1 class by relativization. See pp. 691-692 of [JLR91] and Theorem 2.7 of [CR98] for examples.

In [Odi92], the author points out that the study of unbounded Π^0_1 class is radically different, but can be developed analogously in many aspects, when replacing the notion of computable sets with that of hyperarithmetic sets and considering hyperdegrees instead of Turing degrees. The treatment of unbounded Π^0_1 classes is promised to be done in the yet-to-be-published third volume of the series.

4.2 Basis theorems

Basis theorems tell us that any nonempty Π^0_1 class of a certain class of Π^0_1 classes contains at least one member of some class of functions.

Definition 4.2.1. A class of functions $\mathcal{B}$ is a *basis* for (a class $\mathcal{C}$ of) Π^0_1 classes if every nonempty Π^0_1 class (of $\mathcal{C}$) contains an element of $\mathcal{B}$. Analogously; a class of degrees is a basis if so is the union of all of its degrees. If a class of functions or degrees is not a basis, we call it a *nonbasis*.

Observe that $\{f\}$ is a Π^0_1 class for every computable function f. Therefore, every basis for the class of all Π^0_1 classes has to be a superset of the class of computable functions. But the class of computable functions itself is not a basis for the class of all Π^0_1 classes, because there are Π^0_1 classes without computable members, i.e. special Π^0_1 classes. We devote Section 4.2.7 to these classes.

Note furthermore that whenever we prove a basis result for computably bounded Π^0_1 classes, these also apply for Π^0_1 classes containing a computably bounded function, because theses classes have nonempty computably bounded Π^0_1 subclasses.

4.2.1 Kreisel Basis Theorem

For our first basis result, we introduce the notion of extendible nodes.

Definition 4.2.2. A node σ of a tree T is *extendible* if $I(\sigma) \cap [T] \neq \emptyset$; i.e., there is an infinite path through T with σ as initial segment. $Ext(T) := \{\sigma | I(\sigma) \cap [T] \neq \emptyset\}$ is the set of extendible nodes of T.

Theorem 4.2.1. *Let T be a computable tree. Then the following holds.*

a) $Ext(T)$ is Σ^1_1.

b) If T is bounded, then $Ext(T)$ is Π^0_2.

c) If T is computably bounded, then $Ext(T)$ is Π^0_1.

The results in b) and c) are sharp in the arithmetical hierarchy.

Proof. In the general case, the formula just reflects our definition of $Ext(T)$:
$\sigma \in Ext(T) \Leftrightarrow (\exists x)[\sigma \prec x \wedge (\forall n)\ x \restriction n \in T]$

Now, let T be a bounded computable tree.
$\sigma \in Ext(T) \Leftrightarrow (\forall n)(\exists \tau \in \omega^*)[|\tau| = n \wedge \sigma \circ \tau \in T]$
The necessary direction follows by choosing $x \in [T]$ such that $\sigma \prec x$, which exists, because $\sigma \in Ext(T)$. Then $\tau := x \restriction n$ meets the requirement in the formula for any given $n \in \mathbb{N}$. The reverse direction easily follows from Kőnig's Lemma for the tree $\{\rho \mid \tau \in \omega^* \wedge \rho \prec \tau \wedge \sigma \circ \tau \in T\}$. It is indeed a tree, by construction. It is also infinite by assumption and bounded, because T is bounded. So there is an infinite path through this tree, which hence extends σ in T.

Note that this direction does not hold already for computable trees that are computably bounded everywhere but the first level. A formal definition of such notions of *almost boundedness* can be found in [Cen10]. Consider for example
$T_1 := \{\lambda\} \cup \bigcup_{n\in\omega} \{(n) \circ 0^i \mid 0 \leq i \leq n\}$
For this tree, λ is a node that meets the above Π^0_2 condition. But if λ were an extendible node, $[T_1]$ would be nonempty, which is not the case.

Further, this result for bounded trees cannot be improved in terms of the arithmetical hierarchy, because there is a bounded computable tree whose set of extendible nodes is Π^0_2-complete.

$T_2 := 0^* \cup 0^* \circ (1) \cup$
$\{0^e \circ (1) \circ (t_1, \ldots, t_n) \mid \forall 1 \leq x \leq n\ \varphi_e(x-1)[t_x] \downarrow \wedge \varphi_e(x-1)[t_x - 1] \uparrow\}$
Then $Ext(T_2) = 0^* \cup \{0^e \circ (1) \mid e \in \text{TOTAL}\} \cup \{0^e \circ (1) \circ (t_1, \ldots, t_n) \mid \forall 1 \leq x \leq n\ \varphi_e(x)[t_x] \downarrow \wedge \varphi_e(x)[t_x - 1] \uparrow \wedge e \in \text{TOTAL}\}$,
where TOTAL is the index set of the total computable functions.

So, $Ext(T_2) \leq_m$ TOTAL via $f : \omega \to \omega, \sigma \mapsto \begin{cases} e_0 & , \sigma \in 0^* \\ e & , \exists \tau \in \omega^* : \sigma = 0^e \circ (1) \circ \tau \\ e_1 & , \text{otherwise}, \end{cases}$

where $e_0 \in$ TOTAL is the index of some total computable function and $e_1 \notin$ TOTAL is not. Further, TOTAL is Π^0_2-complete and m-reducible to $Ext(T_2)$. Therefore $Ext(T_2)$ is Π^0_2-complete.
$e \in \text{TOTAL} \Leftrightarrow \forall x \exists t \; \varphi_e(x)[t] \downarrow$, so TOTAL is Π^0_2.
Let R be a computable ternary relation and $A = \{n \in \omega | \forall x \exists y \, R(n, x, y)\}$ the corresponding Π^0_2 set. Then $A \leq_m$ TOTAL via $f : \omega \to \omega$, where f computes for every n an index of a computable function that, given x, searches for such a y via dove-tailing, that $R(n, x, y)$ holds and terminates iff it finds such a number y.
Finally, for the reduction TOTAL $\leq_m Ext(T_2)$ consider $g : \omega \to \omega$, $g(e) = 0^e \circ (1)$.

For a computable tree T, bounded by some computable function g, Kőnig's Lemma implies
$\sigma \in Ext(T) \Leftrightarrow$
$(\forall n)(\exists \tau \in \omega^n)(\sigma \prec \tau \wedge \tau \in T) \Leftrightarrow (\forall n)(\exists \tau \in \prod_{i \in \mathbb{N}} \{0, \ldots, x_{\max}(i)\})(\sigma \prec \tau \wedge \tau \in T))$,
where $x_{\max}(0) := g(\lambda)$ and $x_{\max}(i + 1) := \max\{g(\tau) \mid |\tau| = i \wedge \tau \leq_{lex} x_{\max} \restriction i\}$.
The part behind the universal quantifier of the latter formula represents a predicate computable in n and σ. So, $Ext(T)$ is Π^0_1.

Again, this result for computably bounded trees cannot be improved in terms of the arithmetical hierarchy, because there is a computably bounded computable tree whose set of extendible nodes is Π^0_1-complete.

$T_3 := 0^* \cup \{0^e \circ (1) \circ 0^i | \varphi_e(x)[i] \uparrow \forall 0 \leq x \leq i\}$
Then $Ext(T_3) = 0^* \cup \{0^e \circ (1) \circ 0^i | i \in \mathbb{N} \wedge e \in \text{EMPTY}\}$,
where EMPTY is the index set of the nowhere defined function. EMPTY is Π^0_1-complete and m-reducible to $Ext(T_3)$. And therefore $Ext(T_3)$ is Π^0_1-complete.
$e \in \text{EMPTY} \Leftrightarrow \forall x \; \varphi_e(\pi_0(x))[\pi_1(x)] \uparrow$, so EMPTY is Π^0_1.
Let R be a computable binary relation and $A = \{n \in \omega | \forall x \, R(n, x)\}$ the corresponding Π^0_1 set. Then $A \leq_m$ EMPTY via $f : \omega \to \omega$, where f computes, for every n, an index of the computable function that terminates with the output 0 if $R(n, x)$ does not hold, and never terminates, otherwise.

Finally, for the reduction EMPTY $\leq_m$ $Ext(T_3)$ consider $g : \omega \to \omega$, $g(e) = 0^e \circ (1)$. □

From Theorems 4.2.2 and 4.2.4 we can derive, that the result a) cannot be improved in terms of the analytical hierarchy.

Theorem 4.2.2 (Kleene)**.** *For every tree T with nonempty set of infinite paths $[T]$, $[T]$ contains a member computable in $Ext(T)$.*

Proof. Define such an infinite path through T inductively by letting $x(0) := \mu k((k) \in Ext(T))$ and $x(n+1) := \mu k((x(0), \dots, x(n), k) \in Ext(T))$, for each $n \in \omega$. □

Corollary 4.2.3. *Let P be any nonempty Π^0_1 class.*

a) P has a member computable from some Σ^1_1 set (Kleene Basis Theorem)*.*

b) If P is bounded, then P has a member of Σ^0_2 degree, which hence is computable from $\mathbf{0}''$.

c) If P is computably bounded, then P has a member of c.e. degree (C.E. Basis Theorem, [JS72a])*, which hence is computable from $\mathbf{0}'$* (Kreisel Basis Theorem)*.*

d) If $P = [T]$ for a computable tree T such that $T = Ext(T)$, then P has a computable member.

Proof. The results a) and d) are immediate when combining the Theorems 4.2.1 and 4.2.2. As for part b) and c), Theorems 4.2.1 and 4.2.2 combined with Post's Theorem convey only the weaker result, i.e. computability from $\mathbf{0}'$ and $\mathbf{0}''$, respectively, that are implied in the above formulation. More precisely, any function that is computable from some Π^0_n set is also computable from its complement in $\mathbb{N}$, which is Σ^0_n, and Post's Theorem tells us that $\emptyset^{(n)}$ is Σ^0_n-complete.

But looking closely at the particular path constructed in the proof of Theorem 4.2.2, we will see that it also meets the stronger requirement, i.e., it is not only computable from a Σ^0_2 or Σ^0_1 set, respectively, but shares its degree with it.

Let $P = [T]$ like in the proof of Theorem 4.2.2. The infinite path in question is the leftmost infinite path x through T. Consider the set B of non-extendible nodes of T, that are lexicografic predecessors of the initial segment of x of same length.

$$B := \{\sigma \mid \exists m\, \forall \tau \in \mathbb{N}^*\, \forall \rho \in \mathbb{N}^*\; \sigma \in T \wedge (\tau \in T \wedge |\tau| = |\sigma| \wedge \tau \leq_{\text{lex}} \sigma \rightarrow \tau \circ \rho \notin T)\}$$

All strings in B are obviously not extendible and in particular, B consists of exactly those non-extendible stings, that are left of the leftmost infinite path x through T, because B is closed under lexicografic predecessors. The number m can in fact be chosen uniformly, because there are only finitely many strings τ that precede σ lexicografically. Further, B is Σ^0_2 by definition. If T is computably bounded, both universal quantifiers can be replaced by bounded ones; cp. according argument in the proof for Theorem 4.2.1. So, then B is Σ^0_1, or in other words, computably enumerable.

For the initial segments of x the following holds:
$\sigma \prec x$ iff $\sigma \in T \wedge \forall \tau \in T(|\tau| = |\sigma| \wedge \tau <_{\text{lex}} \sigma \rightarrow \tau \in B)$
Since the lexicografic predecessors of σ are only finitely many and can effectively be exhausted, the right-hand side represents a predicate computable in B. Therefore, $x \leq_T B$.

Conversely, one finds that
$\sigma \in B$ iff $\sigma \in T \wedge \sigma <_{\text{lex}} x \upharpoonright |\sigma|$
This is obviously decidable from x and therefore, $B \leq_T x$. So, $x \equiv_T B$. □

We will revisit the observations made in this theorem in Section 4.2.6.

To show that the previous basis theorems are not improvable in regard to the arithmetical and analytical hierarchies, Cenzer and Remmel present in [CR98] some particular Π^0_1 classes proving the following theorem, whose proof we omit. Part a) is due to Kleene.

Theorem 4.2.4. *a) There is a nonempty Π^0_1 class with no Δ^1_1 (hyperarithmetic) member.*

b) There is a nonempty bounded Π^0_1 class with no member computable from $\mathbf{0}'$ ([JLR91]).

c) *There is a nonempty computably bounded* Π^0_1 *class with no computable member, i.e. there is a nonempty special computably bounded* Π^0_1 *class.*

Note that the recursively inseparable sets from Chapter 3 suffice to show statement c). Note further, that by the upcomping Theorem 4.2.24, none of these examples can have an isolated member. And by Theorem 4.2.23, they can hence not be countable Π^0_1 classes.

4.2.2 The Shoenfield construction

Shoenfield improved the Kreisel Basis Theorem from $\leq_T$ to $<_T$ in [Sho60] to obtain what is now called the Kreisel-Shoenfield Basis Theorem. Though, in [JS72b] Jockusch and Soare proved the Low Basis Theorem (4.2.13), which is properly stronger than that theorem. Accordingly, Shoenfield's construction does not seem to have received much attention since then. In this section, we first provide a lemma that implements a generalized version of Shoenfield's construction. Then we derive several interesting results that the author of this thesis did not find in the present literature.

Recall that by π_0 and π_1 we denote the projection functions to the bijective binary coding function $\tau\colon \mathbb{N}^2 \to \mathbb{N}$.

Lemma 4.2.5. *Let* $\mathcal{C}$ *be a class of* Π^0_1 *classes closed against* Π^0_1 *subclasses, such that* $\{f|(f \circ \pi_0) \in P \wedge \forall x\ \pi_1(f(x)) \leq \pi_0(f(x)) + 1\} \in \mathcal{C}$ *for all* $P \in \mathcal{C}$.
If $\{\mathbf{d_i} \mid i \in I\}$ *is a class of mutually incomparable degrees such that the class* $\mathcal{B} \subseteq \{\mathbf{a} \mid \exists i \in I\ \mathbf{a} \leq \mathbf{d_i}\}$ *is a basis for* $\mathcal{C}$*, then already* $\mathcal{B} \setminus \{\mathbf{d_i} \mid i \in I\}$ *is a basis for* $\mathcal{C}$*.*

Proof. The basic idea stems from [Sho60], where the special case of $I = \{0\}$, the Kreisel basis $\mathcal{B} = \{\mathbf{a}|\mathbf{a} \leq \mathbf{d_0}\}$, $\mathbf{d_0} = \mathbf{0}'$, and $\mathcal{C}$ equal to the class of computably bounded Π^0_1 classes is treated.

Define for a function $f : \mathbb{N} \to \mathbb{N}$ the functions f_0, f_1 by $f_i := \pi_i \circ f$, i.e. $f_i(x) := \pi_i(f(x))$ for all $x \in \mathbb{N}, i \in \{0, 1\}$. Then clearly, $f_0 \leq_T f$ and $f_1 \leq_T f$. Define similarly for a string $\sigma \in \mathbb{N}^*$ the strings σ_0, σ_1 by $\sigma_i(x) := \pi_i(\sigma(x))$ for all $0 \leq x < |\sigma|, i \in \{0, 1\}$.

Now let $\emptyset \neq P = [T] \in \mathcal{C}$, with T a computable tree.

Define $\tilde{T} := \{\sigma | \sigma_0 \in T \wedge \sigma_1(n) \leq \sigma_0(n) + 1 \ \forall 0 \leq n < |\sigma|\}$. This is obviously a computable tree and by assumption $[\tilde{T}] \in \mathcal{C}$.

Define further $S :=$
$\{\sigma | \sigma_0 \in T \wedge \forall 0 \leq e < |\sigma| \ (\sigma_1(e) \leq \sigma_0(e) + 1 \wedge \varphi_e^{\sigma_0}(e)[|\sigma|] \downarrow \rightarrow \sigma_1(e) \neq \varphi_e^{\sigma_0}(e))\}$.
This set is obviously computable. It is also a tree, since $\varphi_e^{\sigma_0}(e)|[\sigma|]$ terminates for extensions of a finite oracle if it terminates for the oracle itself, and the result of a converging computation stays the same when extending an oracle.

Since $[S] \subseteq [\tilde{T}]$ and $\mathcal{C}$ is closed against Π^0_1 subclasses, it holds that $[S] \in \mathcal{C}$. Since $P = [T]$ is nonempty, there is $p \in P$. But then also $[S]$ is nonempty, since $\hat{p} \in [S]$, defined by $\hat{p}(n) := \tau(p(n), q(n))$, with $q(n) := \begin{cases} 0, & \varphi_e^p(e) \uparrow \\ 1 - (div(\varphi_e^p(e), 2)), & \varphi_e^p(e) \downarrow \end{cases}$.

For all $i \in I$, let d_i be some function of degree $\mathbf{d_i}$. Now let $f \in [S]$ and $k \in I$ such that $f \leq_T d_k$. That is possible, because $[S]$ is nonempty and $\mathcal{B} \subseteq \{\mathbf{a} \mid \exists i \in I \ \mathbf{a} \leq \mathbf{d_i}\}$ is a basis for $\mathcal{C}$. It holds that $f_1 \not\leq_T f_0$ because S is constructed in a way to make sure that f_1 is different from any total function reducible to f_0, diagonalizing by the reduction index. Then, by transitivity, $f_0 \leq_T d_k$ and $f_1 \leq_T d_k$. Since $f_1 \not\leq_T f_0$, it follows that $f_0 \not\equiv_T d_k$. Suppose, $f_0 \equiv_T d_j$ for some $j \in I, j \neq k$. Then $d_j \equiv_T f_0 <_T d_k$, which contradicts the assumption that d_j and d_k be Turing incomparable. So $f_0 \not\equiv_T d_i$ for all $i \in I$. But f_0 is indeed a member of $P = [T]$ by construction of S. □

The condition that $\mathcal{C}$ be closed against Π^0_1 subclasses together with the other condition is sufficient, but not necessary, because in the proof we only require one specific subclass to be in $\mathcal{C}$. Yet, the lemma is general enough to enclose the following important cases.

Definition 4.2.3. Define the following names for classes of Π^0_1 classes. We define $\mathcal{P}$ as the class of all Π^0_1 classes,

$\mathcal{P}_{\mathrm{b}}$ as the class of bounded Π^0_1 class,
$\mathcal{P}_{\mathrm{cb}}$ as the class of computably bounded Π^0_1 classes,
$\mathcal{P}_{\mathrm{s}}$ as the class of special Π^0_1 classes,
$\mathcal{P}_{\mathrm{b,s}}$ as the class of special bounded Π^0_1 classes and
$\mathcal{P}_{\mathrm{cb,s}}$ as the class of special computably bounded Π^0_1 classes.

Theorem 4.2.6. *Let $\{\mathbf{d_i} \mid i \in I\}$ be a class of mutually incomparable degrees and $\mathcal{C} \in \{\mathcal{P}, \mathcal{P}_{\mathrm{b}}, \mathcal{P}_{\mathrm{cb}}, \mathcal{P}_{\mathrm{s}}, \mathcal{P}_{\mathrm{b,s}}, \mathcal{P}_{\mathrm{cb,s}}\}$. Then the following statements are true:*

a) If $\mathcal{B} \subseteq \{\mathbf{a} \mid \exists i \in I\ \mathbf{a} \leq \mathbf{d_i}\}$ is a basis for $\mathcal{C}$, then so is $\mathcal{B} \setminus \{\mathbf{d_i} \mid i \in I\}$.

b) The class $\mathcal{D} = \{\mathbf{d_i} \mid i \in I\}$ is not a basis for $\mathcal{C}$.

Proof. The classes $\mathcal{P}, \mathcal{P}_{\mathrm{b}}, \mathcal{P}_{\mathrm{cb}}, \mathcal{P}_{\mathrm{s}}$ are obviously closed against Π^0_1 subclasses. Then also any intersection is.

If P is a Π^0_1 class bounded by a function g, then $\tilde{P} := \{f|(f \circ \pi_0) \in P \land \forall x\ \pi_1(f(x)) \leq \pi_0(f(x)) + 1\}$ is bounded by the g-computable function $\tilde{g}$ defined by $\tilde{g}(\sigma) := div(g(\sigma)^2 + 2 \cdot g(\sigma)(g(\sigma)+1) + (g(\sigma)+1)^2 + g(\sigma) + 3(g(\sigma)+1), 2) + 1 = div(4 \cdot (g(\sigma))^2 + 7 \cdot g(\sigma) + 4, 2) + 1$. This works because the tree $\tilde{T}$ is defined in such a way that the relationship between nodes in $\tilde{T}$ and T reflect the relationship between functions $f \in [\tilde{T}] = \tilde{P}$ and $f \circ \pi_0 \in [T] = P$.

Now let P be a special Π^0_1 class and $h \in \tilde{P}$. Then h computes $h \circ \pi_0$, which is noncomputable, because it is a member of P. So $\tilde{P}$ is also a special Π^0_1 class.

That means, that $\tilde{P}$ inherits any of the three properties *boundedness*, *computably boundedness*, *having no computable member* that P might have. So if P is a member of $\mathcal{C}$, then $\tilde{P}$ is, as well.

So any basis $\mathcal{B}$ for the class $\mathcal{C}$ satisfies all conditions for Lemma 4.2.5 and the claim a) holds.

Suppose by way of contradiction, b) would not hold, i.e. $\mathcal{D}$ was a basis for $\mathcal{C}$. Then by a), also $\emptyset$ would be a basis for $\mathcal{C}$, which is a contradiction. Hence, b) holds as well. $\square$

Corollary 4.2.7 (Kreisel-Shoenfield Basis Theorem, [Sho60])**.** $\{\mathbf{a}|\mathbf{a} < \mathbf{0}'\}$ *is a basis for the computably bounded* Π^0_1 *classes.*

Proof. Let $\mathcal{C} = \mathcal{P}_{\text{cb}}$. Then $\mathcal{B} := \{\mathbf{a}|\mathbf{a} \leq \mathbf{0}'\}$ is a basis for $\mathcal{C}$, according to the Kreisel Basis Theorem. Applying Theorem 4.2.6.a gives us the desired result. □

Note that the following result cannot be derived from the Kreisel-Shoenfield Basis Theorem by mere relativization because of the existence of high sets.

Corollary 4.2.8. $\{\mathbf{a}|\mathbf{a} < \mathbf{0}''\}$ *is a basis for the bounded* Π^0_1 *classes.*

Proof. Let $\mathcal{C} = \mathcal{P}_{\text{b}}$. Then $\mathcal{B} := \{\mathbf{a}|\mathbf{a} \leq \mathbf{0}''\}$ is a basis for $\mathcal{C}$, according to Corollary 4.2.3.b. We conclude the proof by applying Theorem 4.2.6.a. □

However, the upcoming Low Basis Theorem (Theorem 4.2.13) is a properly stronger result than the Kreisel-Shoenfield Basis Theorem, and by relativization also implies this corollary.

Remark. In these two corollaries, the bases we are starting with are the downward closure of one singleton class of degrees, while Lemma 4.2.5 works for any subclass of the downward closure of a class of incomparable degrees. In [DDS10], *basis* is defined in such a way that it is always a downward closed basis in the sense of our definition. The authors do not use the term in the actual statement of their basis theorems, however. Also, the more general definition we use seems to be the standard one.

Theorem 4.2.9. *Let* $\mathcal{C} \in \{\mathcal{P}, \mathcal{P}_{\text{b}}, \mathcal{P}_{\text{cb}}, \mathcal{P}_{\text{s}}, \mathcal{P}_{\text{b,s}}, \mathcal{P}_{\text{cb,s}}\}$*. For every basis* $\mathcal{B}$ *of* $\mathcal{C}$ *and every* $n \in \mathbb{N}$*, there are degrees* $\mathbf{b}_0, \ldots, \mathbf{b}_n \in \mathcal{B}$ *satisfying* $\mathbf{b}_0 < \cdots < \mathbf{b}_n$*.*

Proof. Assume that $\mathcal{B}$ is a basis for $\mathcal{C}$ and $n \in \mathbb{N}$ a number such that there are no degrees $\mathbf{b}_0, \ldots, \mathbf{b}_n \in \mathcal{B}$ with $\mathbf{b}_0 < \cdots < \mathbf{b}_n$. We will show that this assumption will lead to a contradiction.

First of all, we observe that then there is, for all $\mathbf{b} \in \mathcal{B}$, a degree $\mathbf{c} \in \mathcal{B}$ maximal in $\mathcal{B}$ such that $\mathbf{b} \leq \mathbf{c}$. For if there were a degree $\mathbf{b} \in \mathcal{B}$ such that every degree $\mathbf{c} \in \mathcal{B}, \mathbf{b} \leq \mathbf{c}$ would not be maximal, one could find for such a degree $\mathbf{c}$ a degree

$\tilde{\mathbf{c}} \in \mathcal{B}$ properly above $\mathbf{c}$, which could not be maximal itself, because it then would also be above $\mathbf{b}$. So, by induction, there would be an infinite sequence of properly increasing degrees, which then contained chains of arbitrary length. But that would contradict the assumption.

For every class of degrees $\mathcal{A} \subseteq \mathcal{B}$ and every degree $\mathbf{d} \in \mathcal{A}$ define the rank $\mathrm{rk}_{\mathcal{A}}(\mathbf{d}) := \max\{k \in \mathbb{N} \mid \exists \mathbf{d}_1, \ldots, \mathbf{d}_k \in \mathcal{A}\ \mathbf{d} < \mathbf{d}_1 < \cdots < \mathbf{d}_k\}$ of $\mathbf{d}$ in $\mathcal{A}$. Note that this rank is well-defined, according to our previous assumption. More precisely, $\mathrm{rk}_{\mathcal{A}}(\mathbf{d}) \leq n$ for all $\mathcal{A} \subseteq \mathcal{B}$ with $\mathbf{d} \in \mathcal{A}$, for the number n from the assumption. Define then, for every class of degrees $\mathcal{A} \subseteq \mathcal{B}$ the subclass of all maximal in $\mathcal{A}$ degrees $M(\mathcal{A}) := \{\mathbf{d} \in \mathcal{A} \mid \mathrm{rk}_{\mathcal{A}}(\mathbf{d}) = 0\}$ and the subclass of degrees with a positive rank $N(\mathcal{A}) := \mathcal{A} \setminus M(\mathcal{A})$. Then observe that for every $\mathcal{A} \subseteq \mathcal{B}$ the following holds.

$$\mathcal{A} \subseteq \{\mathbf{a} \in \mathcal{A} \mid \exists \mathbf{m} \in M(\mathcal{A})\ \mathbf{a} \leq \mathbf{m}\} \tag{4.1}$$

Of course, $M(\mathcal{A}) \subseteq \{\mathbf{a} \in \mathcal{A} \mid \exists \mathbf{m} \in M(\mathcal{A})\ \mathbf{a} \leq \mathbf{m}\}$ holds vacuously. For a degree $\mathbf{d}$ with $\mathrm{rk}_{\mathcal{A}}(\mathbf{d}) = k > 0$, there are degrees $\mathbf{d}_1, \ldots, \mathbf{d}_k \in \mathcal{A}$ such that $\mathbf{d} < \mathbf{d}_1 < \cdots < \mathbf{d}_k$. Then $\mathbf{d}_k$ must be an element of $M(\mathcal{A})$, because otherwise there were still more degrees to prolong the chain starting with $\mathbf{d}$. So also $N(\mathcal{A}) \subseteq \{\mathbf{a} \in \mathcal{A} \mid \exists \mathbf{m} \in M(\mathcal{A})\ \mathbf{a} \leq \mathbf{m}\}$ and hence (4.1) holds.

Furthermore, the degrees in $M(\mathcal{A})$ are mutually incomparable. For if they were not, there were two distinct degrees $\mathbf{d}, \tilde{\mathbf{d}} \in M(\mathcal{A})$ such that $\mathbf{d} \leq \tilde{\mathbf{d}}$. But then $\mathrm{rk}_{\mathcal{A}}(\mathbf{d}) \geq 1$, which would be a contradiction.

Let $\mathcal{A} \subseteq \mathcal{B}$ such that $N(\mathcal{A}) \neq \emptyset$. Then the following holds.

$$\max\{\mathrm{rk}_{\mathcal{A}}(\mathbf{d}) \mid \mathbf{d} \in \mathcal{A}\} \geq \max\{\mathrm{rk}_{N(\mathcal{A})}(\mathbf{d}) \mid \mathbf{d} \in N(\mathcal{A})\} + 1 \tag{4.2}$$

In fact, even equality holds in (4.2), but that is not required for our proof. Let $\mathbf{d} \in N(\mathcal{A})$ such that $\mathrm{rk}_{A}(\mathbf{d}) = \max\{\mathrm{rk}_{N(A)}(\mathbf{a}) \mid \mathbf{a} \in \mathcal{N}(\mathcal{A})\} = k$ and let $\mathbf{d}_1, \ldots, \mathbf{d}_k \in N(\mathcal{A})$ be the degrees of a chain witnessing it: $\mathbf{d} < \mathbf{d}_1 < \cdots < \mathbf{d}_k$. Since $\mathbf{d}_k \in N(\mathcal{A})$, there is some $\mathbf{d}_{k+1} \in \mathcal{A}$ such that $\mathbf{d}_k < \mathbf{d}_{k+1}$. Then this extended chain witnesses that $\mathrm{rk}_{\mathcal{A}}(\mathbf{d}) \geq k + 1$.

So combining all of the observations, we see that for all $\mathcal{A} \subseteq \mathcal{B}$ statement (4.1) holds, while $M(\mathcal{A})$ is a class of mutually incomparable degrees. So if $\mathcal{A}$ is a basis for $\mathcal{C}$, we can apply Theorem 4.2.6.a that tells us, that $N(\mathcal{A})$ also is a basis for $\mathcal{C}$. But we have also seen, that $\max\{\mathrm{rk}_{\mathcal{A}}(\mathbf{d}) \mid \mathbf{d} \in \mathcal{A}\} \leq n$ for every $\mathcal{A} \subseteq \mathcal{B}$ and this value strictly decreases when we apply the operation N, as seen in (4.2). So particularly, since $\mathcal{B}$ is a basis for $\mathcal{C}$ and $\mathcal{B} \subseteq \mathcal{B}$ there is $k \in \mathbb{N}, k \leq n$ such that $\max\{\mathrm{rk}_{N^{[k]}(\mathcal{B})}(\mathbf{d}) \mid \mathbf{d} \in N^{[k]}(\mathcal{B})\} = 0$ and $N^{[k]}(\mathcal{B})$ is also a basis for $\mathcal{C}$. But then $N^{[k]}(\mathcal{B}) = M(N^{[k]}(\mathcal{B}))$, which implies that the degrees in $N^{[k]}(\mathcal{B})$ are mutually incomparable, as observed. But then, $N^{[k]}(\mathcal{B})$ cannot be a basis according to Theorem 4.2.6.b, which is a contradiction. So, reversely, the original claim holds. □

Corollary 4.2.10. *Let $\mathcal{C} \in \{\mathcal{P}, \mathcal{P}_{\mathrm{b}}, \mathcal{P}_{\mathrm{cb}}, \mathcal{P}_{\mathrm{s}}, \mathcal{P}_{\mathrm{b,s}}, \mathcal{P}_{\mathrm{cb,s}}\}$. Then there is no finite basis for $\mathcal{C}$.*

Proof. Obviously, the length of any chain in a finite class of degrees $\mathcal{B}$ is bounded by $|\mathcal{B}|$. Hence, no finite class of degrees $\mathcal{B}$ is able to satisfy the condition for bases of a class $\mathcal{C} \in \{\mathcal{P}, \mathcal{P}_{\mathrm{b}}, \mathcal{P}_{\mathrm{cb}}, \mathcal{P}_{\mathrm{s}}, \mathcal{P}_{\mathrm{b,s}}, \mathcal{P}_{\mathrm{cb,s}}\}$ expressed in Theorem 4.2.9. □

In Chapter 5, we will show other interesting and even new results that are implied by Theorem 4.2.6.

4.2.3 Forcing with Π^0_1 classes

Next, we want to show two classical results of Jockusch and Soare ([JS72b]) - the Low Basis Theorem and the Hyperimmune-free Basis Theorem. Their proofs can be put in the context of a general construction framework known as *forcing with Π^0_1 classes* or *Jockusch-Soare forcing*. This technique will also be used in the proof of the Minimal Pair Basis Theorem. See [DDS10] for a very abstract and modular approach to this kind of forcing and a range of derived basis theorems.

In this forcing-technique one makes use of the fact that by the upcoming Compactness Theorem all of the Π^0_1 classes of a descending sequence of nonempty Π^0_1 classes contain a common member. That provides the instrument of *forcing* any such common element to meet infinitely many requirements, by defining the classes of the sequence to each ensure some requirement for its members while being nonempty. So any member common to all of those Π^0_1 classes will meet all requirements.

To establish the required Compactness Theorem, we first prove a version for general topological spaces.

Theorem 4.2.11. *Let P be a compact space, and $(C_i)_{i\in\mathbb{N}}$ a sequence of closed subsets of P such that $C_i \supseteq C_{i+1}$ and $C_i \neq \emptyset$ for all $i \in \mathbb{N}$. Then $\bigcap\limits_{i=0}^{\infty} C_i \neq \emptyset$.*

Proof. First, define $U_i := P \setminus C_i = \overline{C_i}$ the complements of the C_i. Theses sets are open in P, because the C_i are closed. Then $U_i \subseteq U_{i+1}$ and $U_i \neq P$ for all $i \in \mathbb{N}$. Now assume that $\bigcup\limits_{i=0}^{\infty} U_i = P$. Then the family $(U_i)_{i\in\mathbb{N}}$ is an open cover of P and since P is a compact space, there is a finite subcover. That is, there is $n \in \mathbb{N}$ such that $\bigcup\limits_{i=0}^{n} U_i = P$. But since $U_i \subseteq U_{i+1}$ for all $i \in \mathbb{N}$, it already holds that $U_n = P$. But then $C_n = \overline{U_n} = \emptyset$, which contradicts the assumption. So $\bigcup\limits_{i=0}^{\infty} U_i \neq P$, or conversely, $\bigcap\limits_{i=0}^{\infty} C_i \neq \emptyset$. □

Theorem 4.2.12 (Compactness Theorem, [DDS10])**.** *Let $(T_e)_{e\in\mathbb{N}}$ be a sequence of infinite trees such that $T_e \supseteq T_{e+1}$ and T_0 is bounded. Then $\bigcap_{e\in\mathbb{N}}[T_e] \neq \emptyset$.*

Proof. For a sequence of trees $(T_e)_{e\in\mathbb{N}}$ scuh that $T_e \supseteq T_{e+1}$, we have $[T_e] \supseteq [T_{e+1}]$. Furthermore, because T_0 is bounded, T_e is bounded for every $e \in \mathbb{N}$. And since T_e is infinite for every $e \in \mathbb{N}$, $[T_e] \neq \emptyset$ according to Kőnig's Lemma. Also, every $[T_e]$ is a closed set, since Theorem 2.3.1 states that the closed sets in $\mathbb{N}^\mathbb{N}$ are exactly the sets of infinite paths through trees in $\mathbb{N}^*$.

So to apply Theorem 4.2.11, we need yet to prove that $[T_0]$ is a compact subspace of $\mathbb{N}^\mathbb{N}$. This can easily be done by applying Tychonoff's theorem. We need only to observe that if T_0 is bounded by a function $g : \mathbb{N}^* \to \mathbb{N}$, then $[T_0] \subseteq \prod\limits_{i\in\mathbb{N}}\{0, \dots, x_{\max}(i)\}$, where $x_{\max}(0) := g(\lambda)$ and $x_{\max}(i+1) := \max\{g(\tau) \mid |\tau| = i \wedge \tau \leq_{lex} x_{\max} \upharpoonright i\}$. Tychonoff's theorem states that an arbitrary product of compact topological spaces is compact. And $\{0, \ldots, n\}$ with the discrete topology is obviously compact for all $n \in \mathbb{N}$. Therefore, $\prod_{i\in\mathbb{N}}\{0, \ldots, x_{\max}(i)\}$ is compact. And since $[T_0]$ is a closed subset of $\prod_{i\in\mathbb{N}}\{0, \ldots, x_{\max}(i)\}$, it is also compact.

But since we do not prove Tychonoff's theorem in this thesis, we provide another, more direct, proof for the compactness of $[T_0]$. It uses the notion of *sequentially compactness* which is equivalent to compactness for metric spaces. Since we have defined the metric $d : \mathbb{N}^\mathbb{N} \times \mathbb{N}^\mathbb{N} \to \mathbb{R}$ for Baire space, that induces the topology we are using, we only need to show that every sequence $(x_n)_{n\in\mathbb{N}} \in [T_0]$ has a convergent subsequence. For the rest of the proof, the construction in a way resembles a construction of nested intervals.

Let $(x_n)_{n\in\mathbb{N}} \in [T_0]$ and define the following descending sequence of subtrees, not to be confused with the original one.
$S_0 := T_0, S_{i+1} := S_i \cap I(x_{f(i)} \restriction i)$,
where f is itself inductively defined by an induction codependent with that of the sequence of trees.
Define $f(0) = 0$, therefore $x_{f(0)} \in [S_0]$. For $i > 0$, $f(i)$ shall be the least number above $f(i-1)$ such that $x_{f(i)} \in [S_i]$ and that there are infinitely many $j \in \mathbb{N}$ satisfying $x_j \restriction i = x_{f(i)} \restriction i$. Note that, such a number exists, by applying the pigeon-hole principle to the finitely many extensions of $(x_{f(i-1)} \restriction i-1)$ of length i in S_0 and the infinitely many infinite extensions of $(x_{f(i-1)} \restriction i-1)$ in S_{i-1} that exist among the $(x_i)_{i\in\mathbb{N}}$ by induction.

Then $(x_{f(i)})_{i\in\mathbb{N}}$ is a subsequence of $(x_i)_{i\in\mathbb{N}}$, because $f(i)$ is strictly increasing. Define the function $x(i) := x_{f(i)}(i)$ for all $i \in \mathbb{N}$. Since this infinite string x shares arbitrarily long initial segments with some $x_{f(i)} \in [S_0]$, all of its initial segments are in S_0 and therefore $x \in [S_0]$. By construction, $d(x, x_{f(i)}) \leq 2^{-1}$. So we have found a convergent subsequence of $(x_i)_{i\in\mathbb{N}}$. □

This abstract setting lets one easily try and combine different kinds of requirements as to achieve more and, especially stronger, basis results. For a more formal description and a variety of applications, consult [DDS10]. There, some of the forcing ideas are rolled out into separate *forcing modules*, which may be combined. However, not all such results may be combined. This will be shown in Corollary 4.2.22, where, amongst others, the intersection of the Low basis and the Hyperimmune-free basis, both obtained by forcing, results in a nonbasis.

In our proofs, we will point out how to construct the descending sequence of trees and what the corresponding requirements are that we are forcing; or in other words, the invariant properties of the respective infinite paths.

4.2.4 Low Basis Theorem

We now implement the technique of forcing with Π^0_1 classes to accomplish the most important basis theorem, the Low Basis Theorem. After the statement and proof of the theorem, we will disuss its relation to the Kreisel-Shoenfield Basis Theorem.

Definition 4.2.4. A set A is called *low* if $A \leq_T \emptyset'$, and hence $A \equiv_T \emptyset'$. Analogously, a degree $\mathbf{a}$ is called *low* if $\mathbf{a\prime} = \mathbf{0}'$.

Theorem 4.2.13 (Low Basis Theorem, [JS72b])**.** *Every nonempty computably bounded* Π^0_1 *class contains a low member.*

Proof. Let $P = [T]$ be a computably bounded Π^0_1 class, with $T = T_0$ an infinite computable tree, bounded by the computable function b.

First, define $U_e := \{\sigma \in \mathbb{N}^* | \varphi_e^\sigma(e)[|\sigma|] \uparrow\}$. The set U_e is a computable tree. It is obviously computable, and the reason for it being a tree is that if the computation uses any information of a string σ, and diverges, then it will also diverge for any prefix of σ long enough to contain the numbers the computation was asking for. For any shorter prefix, the oracle request will fail anyway and the respective computation will diverge.

Now define

$$T_{e+1} := \begin{cases} T_e, & T_e \cap U_e \text{ is finite,} \\ T_e \cap U_e, & T_e \cap U_e \text{ is infinite} \end{cases}$$

This definition ensures, for every e, that T_e is an infinite tree and that $T_{e+1} \subseteq T_e$. According to the Compactness Theorem (4.2.12), $[\bigcap_{e\in\mathbb{N}} T_e]$ is nonempty, so choose some $f \in [\bigcap_{e\in\mathbb{N}} T_e]$.

The forced property of $[T_{e+1}]$ is that either $\varphi_e^g(e)\uparrow$ for all $g \in [T_{e+1}]$, or $\varphi_e^g(e)\downarrow$ for all $g \in [T_{e+1}]$.

We now determine the complexitiy of the jump of f. Just as for any function, $\emptyset' \leq_T f'$. For this function f, however, we will now prove that also $f' \leq_T \emptyset'$ holds.

Observe that $\varphi_e^f(e)\downarrow$ if and only if $T_e \cap U_e$ is finite: If $T_e \cap U_e$ is finite, then for all sufficiently large finite oracles in T_e the respective relative computation converges, and then obviously also for $f \in [\bigcap_{i\in\mathbb{N}} T_i] \subseteq [T_e] = [T_{e+1}]$. And if $T_e \cap U_e$ is infinite, then $f \in [\bigcap_{i\in\mathbb{N}} T_i] \subseteq [T_e + 1] \subseteq [U_e]$. And because the computation diverges for all finite subsequences of f as oracle, and there can only be finitely many requests to the oracle in the case of convergence, it diverges also with f as an oracle: $\varphi_e^f(e)\uparrow$. So it is sufficient to determine the complexity of deciding whether $T_e \cap U_e$ is finite, because $f' = \{e \mid \varphi_e^f(e)\downarrow\} = \{e \mid T_e \cap U_e \text{ is finite}\}$.

Furthermore, U_e is obviously uniformly in e computable and T_e is uniformly in e computable in $\emptyset'$. This can be shown by induction. An index i_0 for $T = T_0$ can be hard-wired into the algorithm. Then, compute from indices of U_e and T_e an index i_{e+1} for T_e or for $T_e \cap U_e$, respectively. The decision which one T_{e+1} is, depends on whether or not $T_e \cap U_e$ is finite, by definition. To decide this, ask the halting problem whether the following sentence is true:

$$\exists n\, \forall \sigma\, (\varphi_{i_e}(\sigma) = 1 \wedge |\sigma| = n \rightarrow \varphi_e^\sigma(e)[\sigma]\downarrow) \tag{4.3}$$

Note that $\varphi_{i_e}(\sigma) = 1$ implies that $\sigma(i) < b(\sigma \upharpoonright i)$ for all $0 \leq i < |\sigma|$, because i_e is an index of the tree T_e bounded by b. Since b is computable, the formula following the existential quantifier represents a computable predicate. The halting problem $\emptyset'$ can answer this, since it is Σ^0_1-complete. And this sentence indeed is true if and only if $T_e \cap U_e$ is finite. Because if the sentence is true, there are only finitely many such oracles $\sigma \in T_e$ of a length bounded by some given n, such that the respective relative computation diverges. And if $T_e \cap U_e$ is finite, obviously every relative computation with sufficiently large oracles converge.

This result can finally be reused to show that $\{e | T_e \cap U_e \text{ is finite}\} \leq_T \emptyset'$. For this purpose, compute an index i_e of the set T_e from $\emptyset'$ like above, then once again ask $\emptyset'$, whether (4.3) holds. □

Remark. As mentioned before, the Low Basis Theorem is properly stronger than the Kreisel-Shoenfield Basis Theorem, and hence the Kreisel Basis Theorem, because for any low set A, $\emptyset' \equiv_T A' \geq_T A$. Nevertheless, the Kreisel Basis Theorem is worth proving, for its historical significance on the one hand, and because the used construction also yields results for other Π^0_1 class than computably bounded ones as well as the C.E. Basis theorem. And the Kreisel-Shoenfield Basis Theorem is worth proving because of the beauty of its proof and because its generalization leads to Theorem 4.2.6. This theorem improves the characterization of PA degrees. That then leads to a new result about PA degrees and an easier proof for an existing theorem. See Chapter 5 for these results.

Corollary 4.2.14. *There are chains of low degrees of arbitrary finite length.*

Proof. This follows immediately from the Low Basis Theorem and Theorem 4.2.9. □

It was not pointed out in [JS72b], but the constructed path is in fact truth-table-reducible to the halting problem. This observation is due to Marcus Schaefer, according to [DH10].

Definition 4.2.5. A set A is called *superlow*, if $A \leq_{tt} \emptyset'$, and hence $A \equiv_{tt} \emptyset'$.

Theorem 4.2.15 (Superlow Basis Theorem)**.** *Every nonempty computably bounded Π^0_1 class contains a superlow member.*

Proof. Let T_e, U_e be computable trees for every e and $f \in [\bigcap_{e \in \mathbb{N}} T_e]$ as defined in the proof for the Low Basis Theorem. Recall, that $f' = \{e \mid \varphi^f_e(e) \downarrow\} = \{e \mid T_e \cap U_e \text{ is finite}\}$ and that the finiteness of $T_e \cap U_e$ can be decided by a request to the halting problem, that involves the knowledge of an index of T_e. It is possible to uniformly compute an index for $U_e = \{\sigma \in \mathbb{N}^* | \varphi^\sigma_e(e)[|\sigma|] \uparrow\}$. And by definition of

T_{e+1}, it is therefore possible to uniformly compute an index of T_{e+1} from an index of T_e by only one request to the halting problem. Inductively, one can uniformly in e compute an index of T_e with e many requests to the halting problem.

So to ask the final question that decides whether $T_e \cap U_e$ is finite, and thereby whether $e \in f'$, one first has to ask e many consecutive other questions. And moreover, the actual form of each question depends on every preceding question, since it always involves an index of some tree T_i. More precisely, the first question can be computed in advance. The second question may have two different forms, depending on the answer to the first question. So, by induction, the ith question of the first e many questions required to compute an index for T_e, may have 2^{i-1} different actual forms, depending on the preceding answers. So the outlined algorithm may compute any of as much as up to 2^e many possible indices for T_e, depending on the oracle. Not knowing in advance, which index one will be the true one, one has to prepare to pose the final question, establishing the reduction, in 2^e different ways.

But it is not sufficient to know how many questions might be asked. For a truth-table-reduction we need to uniformly in n compute a definite finite set of questions to the oracle $\emptyset'$ that are sufficient to decide the membership of n in the set f' we want to reduce to the oracle, and also a Boolean expression telling what the decision is, depending on the answers. But that can clearly be done, as well. The first request asks, in some encoded form, whether $T_0 \cap U_0$ is finite. According to the answer, an index of T_1 is computed, on which the actual form of the next question depends. At any stage, however, either $T_{i+1} = T_i$ or $T_{i+1} = T_i \cap U_i$. By induction, $T_{i+1} = T_0 \bigcap_{\sigma_{i+1}(j)=0} U_j$ for every $0 \leq i < e$, where $\sigma_{i+1} \in 2^{i+1}$ denotes the string of preceding answers in chronological order. So, by exhausting the 2^e many strings corresponding to possible ways of answering the e many first questions, one can compute a truth table reflection the 2^{e+1} many courses of questions and answers. The according propositional statement has to be true if and only if the answers correspond to any such computation that the according last question is answered negatively.

We can also estimate the size of the truth table. In every computation, there are exactly $e + 1$ many requests to the oracle. And the ith request can have up to 2^{i-1} many actual forms, for all $1 \leq i \leq e$. So there would be up to $2^{e+1} - 1$ many propositional variables. Their number can be properly lower, if $U_i \supseteq T_i$ for some $0 \leq i < e$. □

There is even a universal low set, in the sense that in any computably bounded Π^0_1 class one can find a function computable from that low set. This follows in a straightforward way from the findings in Chapter 5.

Theorem 4.2.16 (Second Low Basis Theorem, [DDS10])**.** *There is a low set A such that every nonempty computably bounded Π^0_1 class has a member f such that $f \leq_T A$.*

In fact, by Corollary 4.10 of [DDS10], there are countably many such sets that are mutually Turing incomparable.

In the original paper [JS72b], where the Low Basis Theorem was first stated, the authors point out, that the proof can easily be modified so that the jump of the constructed path is in any given degree above $\mathbf{0}'$. Of course this does no longer work for all computably bounded Π^0_1 classes, since there are some containing only computable members. It is sufficient, however, to restrict the scope of the statement to special Π^0_1 classes. We omit the proof.

Theorem 4.2.17. *For any nonempty computably bounded special Π^0_1 class P and any degree $\mathbf{a} \geq \mathbf{0}'$, there is a member of P of degree $\mathbf{b}$ such that $\mathbf{b}' = \mathbf{b} \cup \mathbf{0}' = \mathbf{a}$.*

4.2.5 Hyperimmune-free Basis Theorem

For the next classical Basis Theorem by Jockusch and Soare, we need to define *hyperimmune-free* degrees. As the name suggests, the original definitions relates to a notion of *hyperimmunity.* A hyperimmune-free degree is originally defined as a degree containing no hyperimmune sets. But there is a property of hyperimmune-free degrees that can be shown to be equivalent to that definition. And since the notion of degrees with that property is better accessible and the definition is required nowhere else in this thesis, we take the liberty to instead define hyperimmune-free degrees that way. Some authors (e.g. [DDS10]) use the term *computably dominated*, which is closer to our definition.

Definition 4.2.6. A degree **a** is *hyperimmune-free* if for every function f and set A with $f \leq_T A \in \mathbf{a}$, there is a computable function g with $f(x) \leq g(x)$ for all $x \in \mathbb{N}$.

Note, that while any member x of a computably bounded Π^0_1 class is of course computably bounded, not every function computable from x must be bounded by the same or any other computable function. The following theorem, however, exactly states that there is always at least one such member.

Theorem 4.2.18 (Hyperimmue-free Basis Theorem, [JS72b])**.** *Every nonempty computably bounded Π^0_1 class contains an element of hyperimmune-free degree.*

Proof. Let $P = [T]$ be a computably bounded Π^0_1 class with T an infinite computable tree bounded by the computable function b.

As in the proof for the Low Basis Theorem, we define a sequence of trees we would like to perform intersections with.

$$U^x_e := \{\sigma \in \mathbb{N}^* | \varphi^\sigma_e(x)[|\sigma|] \uparrow\}$$
$$T_{e+1} := \begin{cases} T_e, & T_e \cap U^x_e \text{ is finite for every } x \\ T_e \cap U^{x_e}_e, & T_e \cap U^x_e \text{ is infinite for some } x \text{, and } x_e \text{ is the least such} \end{cases}$$

And U^x_e is a tree, just as in the proof of the Low Basis Theorem. Also, $T_{e+1} \subseteq T_e$ for every e, T_0 is bounded and the definition ensures that T_e is an infinite tree for every e. Again, by the Compactness Theorem, $\bigcap_{e\in\mathbb{N}}[T_e] \neq \emptyset$. So, we choose $f \in \bigcap_{e\in\mathbb{N}}[T_e] \subseteq [T]$.

The forced property of $[T_{e+1}]$ is that either φ^g_e is total for all $g \in [T_{e+1}]$, or φ^g_e is not total for all $g \in [T_{e+1}]$.

We will now show that every total function that is Turing reducible to f can be majorized by a computable function. A fortiori, this shows that every function in the degree of f can be majorized that way. Therefore, the degree of f is hyperimmune-free.

Let $g \leq_T f$ be a total function. So there is a number $e \in \mathbb{N}$, such that $g \equiv \varphi_e^f$. Suppose that $T_e \cap U_e^x$ was infinite for some x. Then $f \in [\bigcap_{i\in\mathbb{N}} T_i] = \bigcap_{i\in\mathbb{N}}[T_i] \subseteq [T_{e+1}] = [T_e \cap U_e^{x_e}]$ and so $\varphi_e^f(x)\uparrow$, similar to the observation in the previous proof. So g would not be total, contradicting our assumption. Hence, $T_e \cap U_e^x$ is finite for all x. Define the function $k(x) := \min\{l \in \mathbb{N} | |\sigma| = l \wedge \sigma \in T_e \to \varphi_e^\sigma(x)\downarrow\}$. It is well-defined, because $T_e \cap U_e^x$ is finite, and it is also computable, because T_e is computable and bounded by the computable function b.

Then the following function is computable.
$h_e(x) := \max\{\varphi_e^\sigma(x) | \sigma \in T_e \wedge |\sigma| = k(x)\}$
Since $\varphi_e^f(x) = \varphi_e^\sigma(x)$ for some σ with $|\sigma| = k(x)$, it holds that h_e indeed majorizes $g \equiv \varphi_e^f$. □

Corollary 4.2.19. *There are chains of hyperimmune-free degrees of arbitrary finite length.*

Proof. This follows immediately from the Hyperimmune-free Basis Theorem and Theorem 4.2.9. □

Remark. It is shown in [DDS10], that the constructed path is low$_2$. Though, it is impossible to choose a low path with the same property. Neither can we always find a member of a given computably bounded Π_1^0 class that satisfies both statements of any other pair of bases of the previous basis theorems. This will be stated more exactly in the following Corollary 4.2.22. We first state some results we require for the proof of the corollary.

Theorem 4.2.20 ([JS72a])**.** *There is a nonempty computably bounded* Π_1^0 *class* P *such that* $\mathbf{0}'$ *is the only c.e. degree of a member of* P*.*

Proof. Jockusch and Soare showed in [JS72a], that there is a nonempty computably bounded Π_1^0 class P such that if $\mathbf{a}$ is the degree of any member of P and $\mathbf{c}$ is a c.e. degree and $\mathbf{a} \leq$, then $\mathbf{c} = \mathbf{c}'$. According to the C.E. basis theorem, P has got a member of c.e. degree. So that degree must then be $\mathbf{0}'$. □

Remark. We will find in the final chapter that the Π^0_1 class of complete extensions of Peano arithmetic has the property of both the statement of this theorem and the stronger property used in the proof.

Interestingly, this theorem has a degree-theoretic ramification, as was pointed out in [JS72a]. By the Low Basis theorem the mentioned Π^0_1 class P has got a member of low degree. Consequently, there is a low degree such that the only c.e. degree above it is $\mathbf{0}'$.

The final ingredient to Corollary 4.2.22 is the following lemma, that we present without proof.

Lemma 4.2.21 ([DDS10])**.** *No degree* $\mathbf{0} < \mathbf{d} \leq \mathbf{0}'$ *is hyperimmune-free.*

No we combine these observations to a corollary stating the incompatibility of the three most important basis theorems. Some of the implications can be found scattered across the literature.

Corollary 4.2.22. *The intersection of any two of the Kreisel-Shoenfield basis, the C.E. basis and the Hyperimmune-free basis is not a basis for the computably bounded* Π^0_1 *classes. The same holds when replacing the Kreisel-Shoenfield basis with the Low basis.*

Proof. According to Theorem 4.2.4.c, $\{\mathbf{0}\}$ is not a basis for the computably bounded Π^0_1 classes. And according to Theorem 4.2.20, neither is $\{\mathbf{d} \mid \mathbf{d} < \mathbf{0}' \wedge \mathbf{d} \text{ is a c.e. degree}\}$.

Because of Lemma 4.2.21, the intersection of the Kreisel-Shoenfield basis with the Hyperimmune-free basis equals $\{\mathbf{0}\}$. Analogously, the intersection of the C.E. basis with the Hyperimmune-free basis equals $\{\mathbf{0}\}$. The intersection of the C.E. basis with the Kreisel-Shoenfield basis equals $\{\mathbf{d} \mid \mathbf{d} < \mathbf{0}' \wedge \mathbf{d} \text{ is a c.e. degree}\}$.

Since the Low basis is a subclass of the Kreisel-Shoenfield basis, the according statement holds for any intersection with the Low basis. □

4.2.6 Countable Π^0_1 classes, isolated paths

Definition 4.2.7. An *isolated path* x of a class of functions $P \subseteq \omega^\omega$ is an isolated point in the sense of the topology of ω^ω. This means, that there is an open set in ω^ω containing only x.

In contrast to generic members of a Π^0_1 class, isolated members are particularly simple to describe. That is why their complexity is lower in general. In contrast, it is not necessary for a member of somewhat low complexity to be isolated, as the tree 2^* illustrates. It has members of any degree while none of its members is isolated.

Theorem 4.2.23. *Let P be a nonempty countable Π^0_1 class. Then P has an isolated path. If P is finite, all of its members are isolated. If P is countably infinite, the set of its isolated paths is infinite.*
In short: For all Π^0_1 classes P, if $|P| \leq \aleph_0$, then $|P| = |\{x \in P \mid x \text{ isolated in } P\}|$.

Proof. At least the first statement can be proved in mere topological terms, as it is true for every closed so-called Polish space. That is a topologic space with a countable dense subset that is metrizable with a complete metric. That the metric d on Baire space is complete, can be shown with a similar technique as implemented in the proof of the Compactness Theorem (4.2.12). And the set of almost everywhere constant functions is a countable dense subset, as one easily checks.

But in this context, we give a specific proof for the Baire space ω^ω, as it may be more insightful to the considered structures.

If P is empty, $|P| = |\{x \in P | x \text{ isolated in } P\}|$ holds vacuously. If P is finite, then there cannot be any $x \in P$ and $(x_i)_{i \in \mathbb{N}} \in P$ so that $x \restriction i \equiv x_i \restriction i, x \neq x_i$ for all i. So there is a neighbourhood for each $x \in P$ that contains none but the point x itself, i.e. all membes of P are isolated.

Now let P be a nonempty Π^0_1 class and T a computable tree, such that $P = [T]$. Assume for a proof by contraposition that $P = [T]$ has no isolated member. We show that P then must be uncountable. We do that by identifying a subtree S of $Ext(T)$ such that 2^* is embeddable in it, with respect to the relation $\precneqq$ of being proper predecessor. This done bottom-up, i.e. we look for some string to map λ to and then we consider two incompatible extensions σ_0 and σ_1 to map 0 and 1 to. And then, inductively for every σ, we map the two strings extending σ to two strings on two (even later) incompatible extensions. While the idea is simple, the formal construction and the proof of its correctness turn out to be cumbersome.

The embedding shows that every nonempty countable Π^0_1 class has at least one isolated member. At the end of this proof, we show that this argument serves also to show that the set of isolated points of a countably infinite Π^0_1 class P must indeed be infinite. Let us now turn to the construction.

Since no member of P is isolated in P, there is a function $f : P \times \omega \to P$, such that $f(x,n) \neq x$ and $x \upharpoonright n \equiv f(x,n) \upharpoonright n$ for all $x \in P, n \in \mathbb{N}$. Define furthermore the function $s : \mathbb{N}^{\mathbb{N}} \times \mathbb{N}^{\mathbb{N}} \to \mathbb{N} \cup \{\infty\}, (x,y) \mapsto s(x,y) := \min\{n \in \mathbb{N} | x(n) \neq y(n)\}$, that tells us, what the first number is, two functions x and y disagree about. For the sake of totality, s is defined to be ∞ on the diagonal.

Let $x_\lambda \in P \neq \emptyset$. Define $x_0 := x_\lambda$ and $x_1 := f(x_\lambda, 0)$. Assume now, x_σ has already been defined for all σ of length shorter or equal to $n \in \mathbb{N}$. Then inductively define $x_{\sigma\circ 0} := x_\sigma$ and $x_{\sigma\circ 1} := f(x_\sigma, s(x_{(\sigma\upharpoonright|\sigma|-1)\circ 0}, x_{(\sigma\upharpoonright|\sigma|-1)\circ 1}))$. Finally define $b : 2^* \times \mathbb{N} \setminus \{0\} \to \mathbb{N} \cup \{\infty\}, (\sigma, i) \mapsto b(\sigma, i) := s(x_{((\sigma\circ 0^\omega)\upharpoonright i-1)\circ 0}, x_{((\sigma\circ 0^\omega)\upharpoonright i-1)\circ 1})$. Then we can more compactly describe the relation between the infinite paths assigned to successive finite strings: $x_{\sigma\circ 0} = x_\sigma$ and $x_{\sigma\circ 1} = f(x_\sigma, b(\sigma, |\sigma|))$.

This means that the two incomparable strings $\sigma \circ 0$ and $\sigma \circ 1$ are assigned two infinite paths $x_{\sigma\circ 0}$ and $x_{\sigma\circ 1}$ that disagree for the first time at a number larger than that of the previous least disagreement, i.e. that of the paths for $(\sigma \upharpoonright |\sigma| - 1) \circ 0$ and $(\sigma \upharpoonright |\sigma| - 1) \circ 1$. More formally, this means that the x_σ are defined in such a way that $b(\sigma, i)$ is a strictly monotonic increasing function in the second argument: According to the definition, $b(\sigma, i+1) = s(x_{(\sigma\circ 0^\omega)\upharpoonright i\circ 0}, x_{(\sigma\circ 0^\omega)\upharpoonright i\circ 1})$, and $x_{(\sigma\circ 0^\omega)\upharpoonright i\circ 0} = x_{(\sigma\circ 0^\omega)\upharpoonright i}$. Now either $x_{(\sigma\circ 0^\omega)\upharpoonright i} = x_{(\sigma\circ 0^\omega)\upharpoonright(i-1)\circ 0}$ or $x_{(\sigma\circ 0^\omega)\upharpoonright i} = x_{(\sigma\circ 0^\omega)\upharpoonright(i-1)\circ 1}$. Either way, because $x_{(\sigma\circ 0^\omega)\upharpoonright i\circ 1} = f(x_{(\sigma\circ 0^\omega)\upharpoonright i}, s(x_{(\sigma\circ 0^\omega)\upharpoonright(i-1)\circ 0}, x_{(\sigma\circ 0^\omega)\upharpoonright(i-1)\circ 1}))$, it holds that

$b(\sigma, i+1) = s(x_{(\sigma\circ 0^\omega)\upharpoonright i\circ 0}, x_{(\sigma\circ 0^\omega)\upharpoonright i\circ 1}) > s(x_{(\sigma\circ 0^\omega)\upharpoonright (i-1)\circ 0}, x_{(\sigma\circ 0^\omega)\upharpoonright (i-1)\circ 1}) = b(\sigma, i)$ for all $i \geq 1$.

Now define the embedding of 2^* into $Ext(T)$.
$E : 2^* \to Ext(T), \sigma \mapsto E(\sigma) := x_\sigma \upharpoonright b(\sigma, |\sigma| + 1)$. This map is well-defined, because every $E(\sigma)$ is an initial segment of some x_σ, which is an infinite path through T. The fact that E is injective follows from the following property.

The embedding E respects the relation $\precneqq$ of proper predecessors, i.e. $\tau \precneqq \sigma \Leftrightarrow E(\tau) \precneqq E(\sigma)$. Let $\tau \precneqq \sigma$. Then $|E(\sigma)| = b(\sigma, |\sigma|+1) > b(\sigma, |\tau|+1) = b(\tau, |\tau|+1) = |E(\tau)|$. So it is sufficient to show, that $E(\tau) \prec x_\sigma$.

x_σ is defined via x_τ in the inductive definition. If $\tau \circ 0^k = \sigma$ for some k, then $x_\sigma = x_\tau$, and so $E(\tau) \prec x_\sigma$.

So let us assume that $\sigma \notin \tau \circ 0^*$. There must be a least $0 < j \leq |\sigma| - |\tau|$ such that $\sigma(|\tau| - 1 + j) = 1$. This means that $x_{\sigma\upharpoonright|\tau|-1+j} = x_\tau$. Therefore, $|E(\tau)| = b(\tau, |\tau|) = b(\tau \circ 0^j, |\tau|) < b(\tau \circ 0^j, |\tau| + j) = s(x_{\tau\circ 0^j}, x_{\tau\circ 0^j\circ 1}) = s(x_\tau, x_\sigma)$, hence $E(\tau) \prec x_\sigma$.

Now show $E(\tau) \precneqq E(\sigma) \Rightarrow \tau \precneqq \sigma$ by contraposition. The implication $\tau = \sigma \Rightarrow E(\tau) = E(\sigma)$ holds trivially. And $\sigma \precneqq \tau \Rightarrow E(\tau) \not\prec E(\sigma)$, because $|E(\sigma)| < |E(\tau)|$, like above. So the only case left is $\tau|\sigma$. So assume that there are strings $\rho, \hat{\tau}, \hat{\sigma}$ such that, without loss of generality, $\tau = \rho \circ 0 \circ \hat{\tau}, \sigma = \rho \circ 1 \circ \hat{\sigma}, \hat{\tau} \neq \lambda, \hat{\sigma} \neq \lambda$. Then $s(x_\tau, x_\sigma) = b(\rho, |\rho| + 1) = |E(\rho)|$. But $|E(\tau)| > |E(\rho)| < |E(\sigma)|$, because ρ is a proper predecessor of both τ and σ. Therefore, $E(\sigma)|E(\sigma)$, especially $E(\tau) \precneqq E(\sigma)$.

Consequently, E is an embedding of $(2^*, \precneqq)$ into $(Ext(T), \precneqq)$. This induces an injective map of $[2^*]$ into $[Ext(T)] = [T]$. Therefore, $|[2^*]| = 2^{\aleph_0} \leq [T]$. By contraposition, this shows that every nonempty countable Π^0_1 class has an isolated member.

To show that every countably infinite Π^0_1 class has already infinitely many isolated members, one can reuse the above construction. Let P be a countably infinite Π^0_1 class and assume that there are only finitely many isolated members. Then there must be $x \in P$ that is not isolated. So there exist $(y_i)_{i\in\mathbb{N}} \in P$ such that $x \upharpoonright i \equiv y_i \upharpoonright i, x \neq y_i$ for every $i \in \mathbb{N}$. Let $n \in \mathbb{N}$ be sufficiently large such that

$s(x,z) < s(x,y_i)$ for all isolated $z \in P$ and all $i > n$. Define $\sigma := x \upharpoonright s(x, y_{n+1})$. Then consider the Π^0_1 class $\hat{P} := P \cap I(\sigma)$. By construction, this class $\hat{P}$ has no isolated members and by the above part of the proof, is uncountable. But since $\hat{P} \subseteq P$, the class P must be uncountable as well, which contradicts the assumption. □

Cenzer and Remmel, in [CR98], attribute the following theorem to Kreisel ([Kre59]). It strengthens the Kreisel-Shoenfield Basis Theorem to Corollary 4.2.25 in the case of countable Π^0_1 classes.

Theorem 4.2.24 ([Kre59])**.** *Let P be a Π^0_1 class and $x \in P$ an isolated member.*

a) x is hyperarithmetic.

b) If P is bounded, then x is computable from $\mathbf{0}'$.

c) If P is computably bounded, then x is computable.

Proof. Let T be a computable tree such that $[T] = P$. If P is bounded or computably bounded, let T be bounded or computably bounded, respectively.

Since x is isolated, there is a $n \in \mathbb{N}$ such that x is the only element of P extending $x \upharpoonright n$. Define the computable tree $\tilde{T} := T \cap \mathit{Pref}(I(x \upharpoonright n))$ consisting of all strings in T comparable to $x \upharpoonright n$. Then $[\tilde{T}] = \{x\}$. Observe, that if T is bounded, that S is bounded by the same function. Therefore, we can assume, without loss of generality, that T has only one member x.

But then clearly $\mathit{Ext}(T) = \{x \upharpoonright i \mid i \in \mathbb{N}\}$ and hence $x \equiv_T \mathit{Ext}(T)$. We recall now the observations made in Theorem 4.2.1 and strengthen them, which is possible, because x is now not only the leftmost member of P, but the also the rightmost, and hence there is only one extendible node of any given length.

By Theorem 4.2.1, $\mathit{Ext}(T)$ is Σ^1_1. Also, the following statement holds.
$\sigma \in \mathit{Ext}(T) \Leftrightarrow (\forall \tau \in \omega^*)((|\tau| = |\sigma| \wedge \tau \neq \sigma) \rightarrow \tau \notin \mathit{Ext}(T))$
Because the quantifier is of first order, and $\mathit{Ext}(T)$ is Σ^1_1, the above formula is Π^1_1. Therefore $\mathit{Ext}(T)$ is both Σ^1_1 and Π^1_1, it is Δ^1_1 and hence hyperarithmetic.

Now suppose T is bounded. Then Theorem 4.2.1 states, that $Ext(T)$ is Π^0_2. Suppose, there were a $m \in \mathbb{N}$ so that for all $k \in \mathbb{N}$ there were a $\tau \in T$ of length $|\tau| > k$ with $\tau \upharpoonright m \neq x \upharpoonright m$. Then these τ, together with all of their prefixes, would form an infinite bounded tree S. Then from Kőnig's Lemma we know, that there is an infinite path through S. As S is a subtree of T, that path must equal x, which is a contradiction, because none of the strings in S extend $x \upharpoonright m$. So, the reverse statement holds and thus the initial segments of x can be characterized as follows.
$\sigma \in Ext(T) \Leftrightarrow (\exists k \in \mathbb{N})(\forall \tau \in \omega^*)((\tau \in T \wedge |\tau| > k) \rightarrow \sigma \prec \tau)$
So $Ext(T)$ is also Σ^0_2 and hence Δ^0_2. By Post's Theorem, $Ext(T)$ is computable in $\mathbf{0}'$.

Now suppose further, that T is computably bounded. Then, by Theorem 4.2.1, $Ext(T)$ is Π^0_1. Also, it holds that
$\sigma \in Ext(T) \Leftrightarrow (\forall \tau \in \omega^*)((f \text{ bounds } \tau \wedge \tau \neq \sigma) \rightarrow \tau \notin Ext(T))$.
Because the complement of $Ext(T)$ relative to ω^* is Σ^0_1 and the universal quantifier in the above formula is computably bounded, we can derive that $Ext(T)$ is also Σ^0_1 and hence $\Delta^0_1 = \Delta^0_0$, i.e. computable. □

Corollary 4.2.25. *Let P be a countable Π^0_1 class.*

a) P has $|P|$ many hyperarithmetic members.

b) If P is bounded, then P has $|P|$ many members computable from $\mathbf{0}'$.

c) If P is computably bounded, then P has $|P|$ many computable members.

Proof. Combine Theorems 4.2.23 and 4.2.24. Note that Lemma 4.2.5 is not applicable, because $\{f|(f \circ \pi_0) \in P \wedge \forall x\ \pi_1(f(x)) \leq \pi_0(f(x)) + 1\} \in \mathcal{C}$ is not countable. Also it would contradict c). □

One can generalize the notion of isolated points by the Cantor-Bendixson rank. For a Π^0_1 class P, the rank of x in P, is the least ordinal α so that $x \in D^\alpha(P) \setminus D^{\alpha-1}(P)$, where D denotes the removal of the isolated points from a set. Hence isolated points can be characterized as elements of rank 0. For results achieved by this approach, see [Cen10] and [CR98].

4.2.7 Special Π^0_1 classes, Minimal Pair Basis Theorem

In this section we examine, what can be said about the degree spectrum of a Π^0_1 class based on the fact that it has no computable member. Note that no statement we derive here for special Π^0_1 classes is exclusive for these, since one can always add the computable path 0^ω to a Π^0_1 class without changing the fact that it is a Π^0_1 class or even any boundedness property.

Definition 4.2.8. A Π^0_1 class is called *special* if it contains no computable members.

Corollary 4.2.26. *Every special computably bounded Π^0_1 class P is uncountable. More precisely, $|P| = 2^{\aleph_0}$.*

Proof. By Theorem 4.2.23, a countable Π^0_1 class P contains an isolated member. And if P is computably bounded, that isolated member is already computable, by Theorem 4.2.24. So no countable computably bounded Π^0_1 class is special. Or conversely, every special computably bounded Π^0_1 class P is uncountable. Without making use of the Continuum Hypothesis, the proof of Theorem 4.2.23 shows that for such class P, in fact, $|P| = 2^{\aleph_0}$. □

Jockusch and Soare ([JS72a]) show some theoreoms, that are interesting in this context. We omit some of the proofs.

Theorem 4.2.27 ([JS72a])**.** *For any special Π^0_1 class P there is a c.e. degree $\mathbf{c} > \mathbf{0}$ such that P has no member of degree below or equal to $\mathbf{c}$.*

By the C.E. basis theorem, any nonempty computably bounded Π^0_1 class P has a member of c.e. degree $\mathbf{b}$. If P has no computable member, however, the previous theorem implies, that there is another c.e. degree $\mathbf{c}$ so that there is no member of P of any c.e. degree $\mathbf{d}$ below or equal to $\mathbf{c}$. So by the famous result of Sacks, stating that the partial order of c.e. degrees is dense, every special Π^0_1 class does not only have no computable member, but lacks the members of infinitely many c.e. degrees.

Further, Jockusch and Soare show the following.

Theorem 4.2.28 ([JS72a]). *For every c.e. degree* **c** *there is a computably bounded* Π^0_1 *class* P *such that the c.e. degrees of members of* P *are exactly the c.e. degrees* **c** *and above.*

The case $\mathbf{c} = \mathbf{0}$ is trivial. When choosing $\mathbf{c} > \mathbf{0}$, the according Π^0_1 class is hence special. So for these particular special Π^0_1 classes, there are not only no members of c.e. degrees below **c**, but also none of c.e. degrees incomparable to **c**, while all of the other c.e. degrees are represented by some member.

Theorem 4.2.27 implies, that there is no Π^0_1 class with a degree spectrum consisting exactly of the nonzero c.e. degrees. However, if one allows additional computable paths, the according statement holds.

Theorem 4.2.29. *There is a bounded* Π^0_1 *class* P *so that the degrees of members of* P *are precisely the c.e. degrees.*

Proof. The c.e. sets can be characterized as the domains of partial computable functions.

$T_e := \{\sigma \in \omega^* \mid \forall 0 \leq i < |\sigma| \; \varphi_e(i)[\max\{\sigma(i), |\sigma|\}] \downarrow \rightarrow (\sigma(i) > 0 \wedge \varphi_e(i)[\sigma(i)] \downarrow \wedge \varphi_e(i)[\sigma(i) - 1] \uparrow)\}$

The defining formula holds for the values at any position of the string. The only varying aspect is the parameter $|\sigma|$. But for increasing $|\sigma|$, the condition for the implication changes only from false to true, not vice versa, so that T_e is closed under intial segments.

There is the following infinite path through T_e:

$$x_e(n) := \begin{cases} 0, & \varphi_e(n) \uparrow, \\ t_n, & \varphi_e(n) \downarrow, \end{cases}$$

where t_n is the lowest number of steps, after which the computation of $\varphi_e(n)$ converges.

There is no other infinite path through T_e, because strings τ with $|\tau| \geq n + 1$ and $\tau(n) \neq x_e(n)$ will not be extended to strings longer than $t_n - 1$, if $\varphi_e(n) \downarrow$, or even only $|\tau|$, if $\varphi_e(n) \uparrow$.

Further, $W_e \leq_T x_e$, where W_e denotes the domain of φ_e, because $x_e(n) > 0$ iff $n \in W_e$. Also, $x_e \leq_T W_e$, because one can uniformly compute t_e in the case $\varphi_e(n)\downarrow$, which is equivalent to $n \in W_e$, while $x_e(n) = 0$ iff $n \in W_e$. So, $x_e \equiv_T W_e$.

Now simply consider the composite tree
$T := 0^* \cup 0^* \circ 1 \cup \bigcup_{e\in\mathbb{N}} 0^e \circ 1 \circ T_e$
Clearly, there is an infinite path $\tilde{x}_e := 0^e \circ 1 \circ x_e$ through T for every $e \in \mathbb{N}$. Since x_e and $\tilde{x}_e$ only differ in their initial segment, $x_e \equiv_T \tilde{x}_e$. The only infinite path through T apart from the $\tilde{x}_e$ is the computable path 0^ω. □

Note that by the Hyperimmune-free Basis Theorem combined with Lemma 4.2.21 the construced Π^0_1 class cannot be computably bounded.

Here is another interesting result from [JS72b]. We omit the proof.

Theorem 4.2.30. *Let* $\mathbf{a} > \mathbf{0}$ *be any noncomputable degree. There is a nonempty special computably bounded* Π^0_1 *class containing no member of any degree* $\mathbf{d} \geq \mathbf{a}$.

That means, every basis of any of the classes $\mathcal{P}, \mathcal{P}_{\mathrm{b}}, \mathcal{P}_{\mathrm{cb}}, \mathcal{P}_{\mathrm{s}}, \mathcal{P}_{\mathrm{b,s}}, \mathcal{P}_{\mathrm{cb,s}}$ must contain, for every degree $\mathbf{a} > \mathbf{0}$, a degree $\mathbf{b}$ such that either $\mathbf{b} < \mathbf{a}$ or $\mathbf{b}|\mathbf{a}$.

Definition 4.2.9. Let $\mathbf{a}, \mathbf{b} > \mathbf{0}$ be two nonzero degrees. We call $\mathbf{a}, \mathbf{b}$ a *minimal pair* if $\mathbf{0}$ is the only degree below or equal to both of them:
$\forall \mathbf{c}(\mathbf{c} \leq \mathbf{a} \wedge \mathbf{c} \leq \mathbf{b} \Rightarrow \mathbf{c} = \mathbf{0})$. Analogously, two noncomputable functions are called a *minimal pair* if the only functions computable from both of them are already computable.

The proof of the previous theorem uses the construction of a minimal pair of Π^0_1 classes done by Jockusch and Soare in a different paper. A minimal pair of Π^0_1 classes is a pair of Π^0_1 classes $S, \tilde{S}$, such that any pair of functions $f \in S, \tilde{f} \in \tilde{S}$ form a minimal pair. Since their particular $S, \tilde{S}$ are classes of separating sets of pairs of c.e. sets, the construction generalizes the result of Yates and Lachlan, who independently proved the existence of a minimal pair of c.e. sets.

The following *Minimal Pair Basis Theorem* also deals with minimal pairs of degrees. The proof of the theorem is adapted from [DJ09] and uses *forcing with*

Π^0_1 classes (see Section 4.2.3). Note that we here use the more common definition of *minimal pair*, which requires both degrees to be noncomputable, contrary to the cited paper.

There is an equivalent, but differently stated version of this theorem in [DDS10], however, the proof given there is flawed, as was confirmed by one of the authors in response to a counter-example ([Dzh12]).

Theorem 4.2.31 (Minimal Pair Basis Theorem, [JS72b])**.** *Let B be some set of degree $\mathbf{b} > \mathbf{0}$. Every nonempty special computably bounded Π^0_1 class contains a member of such degree $\mathbf{a}$ that $\mathbf{a}, \mathbf{b}$ form a minimal pair.*

Proof. We proof the theorem by forcing with Π^0_1 classes, introduced in Section 4.2.3. So let P be a computably bounded Π^0_1 class and $T = T_0$ a computably bounded computable tree such that $P = [T]$.

Define $U_n := \{\sigma \in T | \varphi^\sigma_e(n)[|\sigma|] \uparrow\}$,

$$T_{e+1} := \begin{cases} T_e \cap U_{n_0}, & T_e \cap U_n \text{ is infinite for some } n; n_0 \text{ is the least such} \\ T_e, & T_e \cap U_n \text{ is finite and for all } n \in \mathbb{N}, \sigma, \tau \in Ext(T_e), \\ & \text{ if } \varphi^B_e(n) \downarrow \wedge \varphi^\sigma_e(n) \downarrow \wedge \varphi^\tau_e(n) \downarrow, \text{ then } \varphi^\sigma_e(n) = \varphi^\tau_e(n) \\ (T_e \cap \sigma \circ \omega^*) & T_e \cap U_n \text{ is finite for all } n \in \mathbb{N} \text{ and} \\ \cup \{\sigma \restriction i | i < |\sigma|\}, & \varphi^B_e(n_0) \downarrow \neq \varphi^\sigma_e(n_0) \downarrow \neq \varphi^\tau_e(n_0) \downarrow \text{ for some } n_0 \in \mathbb{N} \\ & \text{and } \sigma, \tau \in Ext(T_e); \langle n_0, \langle \sigma \rangle, \langle \tau \rangle \rangle \text{ minimal} \end{cases}$$

U_n is a tree, similarly to the trees U_e, U^x_e in the Low and Hyperimmunefree Basis Theorem, respectively. T_{e+1} is well defined, because the cases are mutually exclusive and some case applies for every e. T_{e+1} is obviously a tree in all three cases. It is also infinite, which is obvious in the first case and holds by induction in the second. In the third case, this holds because $\sigma \in Ext(T_e)$ and T_{e+1} consists of all nodes of T_e compatible with σ.

By the Compactness Theorem (4.2.12), there is a function $f \in [\bigcap_{e \in \mathbb{N}} T_e] = \bigcap_{e \in \mathbb{N}} [T_e] \subseteq [T]$. So let us now check whether it meets the requirements of the theorem. Consider an index $e \in \mathbb{N}$. We require φ^f_e total $\wedge\, \varphi^B_e$ total $\Rightarrow (\varphi^f_e \equiv \varphi^B_e \Rightarrow \varphi^f_e$ computable).

In the first case of the definition of T_{e+1}, the tree consists only of nodes $\sigma \in T_e$ such that $\varphi_e^\sigma(n_0)[|\sigma|] \uparrow$. Since $f \in \bigcap_{i \in \mathbb{N}}[T_i] \subseteq [T_{e+1}]$, it also holds that $\varphi_e^f(n_0) \uparrow$, i.e. φ_e^f is not total.

We can assume that $\varphi_e^B(n) \downarrow$ and $\varphi_e^\sigma(n) \downarrow$ for all $n \in \mathbb{N}, \sigma \in Ext(T)$, in the second case, because otherwise neither φ_e^B nor φ_e^f would be total and the requirement would be met trivially. For the same reason, we can assume that $\varphi_e^B(n) = \varphi_e^f(n)$ for all $n \in \mathbb{N}$.

But then φ_e^f is computable; here is how: Because U_n is finite for every n, there is a least stage l such that $\varphi_e^\sigma(n)[|\sigma|] \downarrow$ for all $\sigma \in (T_e \cap \omega^l)$. And because $T_e = T_{e+1}$ is computably bounded, one can search for that stage effectively. The results of those computations might still differ - although not for the extendible nodes - according to the assumption. But the result of $\varphi_e^\rho(n) \downarrow$ for a string $\rho \in T_e$ stays the same for any extension of ρ. This means that on every subsequent stage all the results of computations with extendible nodes as oracle reoccur, while every different result eventually disappears. Therefore, one can even effectively search for the least stage, such that all $\varphi_e^\sigma(n)[|\sigma|] \downarrow$ agree on some $m \in \mathbb{N}$. Then $\varphi_e^f(n) = m$, since $f \in \bigcap_{i \in \mathbb{N}}[T_i] \subseteq [T_e]$ and every initial segment of f is in $Ext(T_e)$.

In the third case, T_{e+1} is defined in such a way that $\varphi_e^\rho(n_0) \downarrow \neq \varphi_e^B(n_0)$ for all $\rho \in T_e$, and consequently $\varphi_e^B(n_0) \neq \varphi_e^f(n_0)$, since $f \in \bigcap_{i \in \mathbb{N}}[T_i] \subseteq [T_e]$. Therefore, $\varphi_e^B \not\equiv \varphi_e^f$.

So, for every $e \in \mathbb{N}$ such that φ_e^B and φ_e^f are total and identical, this function is computable. Or, in other words, $g \leq_T f \wedge g \leq_T B \Rightarrow g$ computable, i.e. the degrees of f and B form a minimal pair. □

Note, that the minimal degrees do not form a basis for the computably bounded Π^0_1 classes. This is implied by Corollary 5.2.10.

In [JS72b], Jockusch and Soare prove an intricate theorem, which we want to present here without proof.

Theorem 4.2.32 ([JS72b]). *For any nonempty special computably bounded Π^0_1 class P and any sequence of noncomputable degrees $(\mathbf{a_i})_{i \in \mathbb{N}}$, P has $2^{\aleph_0}$ many* *members, that are mutually Turing incomparable, such that the degree of any of these members is incomparable to each degree* $\mathbf{a_i}$.

From this result, it is easy to derive the Minimal Pair Basis Theorem. The authors of [JS72b] give a corollary in that paper which is a bit weaker, since they only consider a given function that is itself member of the Π^0_1 class. But their proof works for any other noncomputable function as well. That is why we attribute it to [JS72b]. Here is the idea. Given a nonempty special computably bounded Π^0_1 class P and an noncomputable function f, define $(\mathbf{a_i})_{i\in\mathbb{N}}$ as the sequence of degrees of noncomputable functions that are computable from f. You can gain this by eliminating the computable sets from the f-computable functions given by the standard enumeration of the according Turing reductions. Then the above theorem states the existence of $2^{\aleph_0}$ many members of P such that their respective degree forms a minimal pair together with the degree of f.

Another result in [JS72b] shows, that in addition to the above theorem, there are particular Π^0_1 classes with the property that any two members of such a Π^0_1 class are already mutually incomparable.

Theorem 4.2.33 ([JS72b]). *There is a computably bounded* Π^0_1 *class* P *such that any two members of* P *are mutually Turing incomparable. In particular,* P *is special and hence* $|P| = 2^{\aleph_0}$.

5 Application to logical theories

In this last chapter we want to return to the problem that originally motivated the study of degree spectra of Π^0_1 classes. That is, what degrees do the complete extensions of a given consistent effectively axiomatizable first-order theory have? Of course, by Gödel's Incompleteness Theorem, one cannot hope to find a complete extension that is of computable degree, in general. Lindenbaum's Theorem, however, states that there is some complete extension at all.

At first, we want to fix some definitions. Although we will not go into logical theories too far, we need to outline some technical terms. However, a basic familiarity with first-order predicate calculus is helpful for the understanding of some parts of this chapter. The exact knowledge of the statement of the axioms of Peano arithmetic are obsolete here.

For our purposes, a *language* $\mathcal{L}$ will always be a effective first-order language. That is, the index sets to all families of non-logical symbols, i.e. relation, function and constant symbols, are initial segments of $\mathbb{N}$, and the two functions specifying the arities of the relation and function symbols, respectively, are partial computable. When we speak of computability theoretic properties of sets of $\mathcal{L}$-sentences in the following, we do this by identifying them with the according set of codes for sentences of a fixed Gödel numbering. As underlying syntactical deduction system we choose some adequate one such as the Shoenfield calculus, so that we can define consistency as we do in the following.

Definition 5.0.10. The set of (first-order) sentences of $\mathcal{L}$ is denoted by $Sent(\mathcal{L})$. For $\Sigma \subseteq Sent(\mathcal{L})$, the set $Con(\Sigma)$, the *consequences* of Σ, is the closure of Σ under the syntactic deduction $\vdash$. The set $Ref(\Sigma)$ consists of the negations of the consequences of Σ and is called the set of *refutations* of Σ. A set $\Gamma \subseteq Sent(\mathcal{L})$ is a *(first-order) logical ($\mathcal{L}$-)theory* if $\Gamma = Con(\Gamma)$. A set Σ of sentences is called *a set of axioms for* Γ if $\Gamma = Con(\Sigma)$. A theory Γ is *computably axiomatizable* if there is a computable set of axioms for Γ. It is said to be *consistent* if for no $\mathcal{L}$-sentence

ψ both $\Gamma \vdash \phi$ and $\Gamma \vdash \neg\phi$. It is called *complete* if for each $\mathcal{L}$-sentence ϕ either $\Gamma \vdash \phi$ or $\Gamma \vdash \neg\phi$. An *extension* of a theory Γ is a theory Δ such that $\Gamma \subseteq \Delta$.

Note that we defined completeness as maximal consistency.

5.1 Π^0_1 classes as sets of extensions of theories

In [Sho60], Shoenfield supplements Lindenbaum's Theorem by showing that, for each consistent computably axiomatizable first-order theory, its complete extensions form a nonempty Π^0_1 class, without denoting it that way. The term *Π^0_1 class* seems to appear no earlier than in the works of Jockusch and Soare in the 1970s. The $\mathcal{L}$-theories in the statement of the theorem are to be understood as identified with the charactistic functions of the set of Gödel numbers of their members, as suggested before.

Theorem 5.1.1. *([Sho60]) For any computably axiomatizable logical theory* Γ *(of* $\mathcal{L}$*), both the class of consistent extensions of* Γ *and the class of complete extensions of* Γ *are computably bounded* Π^0_1 *classes. These are nonempty if and only if* Γ *is consistent.*

Remark. In [Sho60], Shoenfield mentions that a logical theory is c.e. iff it is computably axiomatizable. So the above theorem can be restated accordingly. Also, the statement can be specified for computable theories, in such a way that the Π^0_1 class P of the consistent (complete) extensions of a computable theory is *decidable*. That is, in that case a computable tree T representing P by $P = [T]$ can be chosen such that $T = Ext(T)$.

Definition 5.1.1. A theory Γ is called *essentially undecidable* if there is no computable complete extension of Γ. A theory Γ is said to be *(recursively) separable* if it is separable from its refutations by a computable set, in the sense of Definition 3.2.2. Otherwise it is called *(recursively) inseparable.*

Every recursively inseparable theory is obviously essentially undecidable. The converse statement does not hold, which was proved by Ehrenfeucht ([Ehr61]).

Rosser improved Gödel's Incompleteness Theorem, that originally required the stronger condition of so-called ω-consistency, to a theorem that then implied that Peano arithmetic is essentially undecidable. According to [CR98], he in fact showed that Peano arithmetic is even recursively inseparable.

Theorem 5.1.1 implies, that all basis theorems for $\mathcal{P}_{\mathrm{cb}}$ apply to the class of consistent (complete) extensions of a computably axiomatizable consistent theory. For example, every such theory has a complete extension that is low, by the Low Basis Theorem. By the additional specification in the remark after the theorem, every decidable consistent computably axiomatizable theory has a complete decidable extension according to Corollary 4.2.3.d. And all basis theorems for $\mathcal{P}_{\mathrm{cb,s}}$ apply to consistent (complete) extensions of a consistent computably axiomatizable theory that is essentially undecidable, for instance Peano arithmetic. That is, by the Minimal Pair Basis Theorem, Peano arithmetic has two complete extensions such that any function computable from both of them is already computable.

We will now state the theorem that establishes the converse of the previous theorem.

Theorem 5.1.2. *([Ehr61],[JS72b]) Every computably bounded* Π^0_1 *class is equal to the class of complete extensions of some computably axiomatizable theory.*

By this result, also the existential theorems for Π^0_1 classes hold. That is, we could derive from the existence of special computably bounded Π^0_1 classes the existence of essentially undecidable computably axiomatizable consistent theories. More precisely, by Theorem 4.2.30, there is for every degree $\mathbf{a} > \mathbf{0}$ a consistent computably axiomatizable theory having no complete extensions of any degree $\mathbf{d} \geq \mathbf{a}$.

5.2 PA degrees, Scott Basis Theorem

We now turn to the special case of recursively inseparable theories. As we have seen, Peano arithmetic is one example of such a recursively inseparable theory. In [JS72b], others are pointed out: Zermelo-Fraenkel set theory and even the finitely axiomatized Robinson arithmetic, which is a subtheory of Peano arithmetic. But we may take Peano arithmetic as a generic representative, because Jockusch and Soare showed the following. We omit the proof.

Theorem 5.2.1. *([JS72b]) Let* Γ *and* $\tilde{\Gamma}$ *be two consistent computably axiomatizable recursively inseparable theories and let* P *and* $\tilde{P}$ *be the respective classes of complete extensions of* Γ *and* $\tilde{\Gamma}$*. Then* P *and* $\tilde{P}$ *have the same degree spectrum.*

We now give this uniquely defined degree spectrum a name.

Definition 5.2.1. We define $\mathcal{D}_{\mathrm{PA}}$ as the class of Turing degrees of complete extensions of Peano arithmetic. A degree $\mathbf{a}$ is called *PA degree* if $\mathbf{a} \in \mathcal{D}_{\mathrm{PA}}$.

We state, without giving a proof, some theorems that we require for the proof of the Theorem 5.2.5 that characterizes the PA degrees.

Theorem 5.2.2. *(Scott Basis Theorem, [Sco62]) If* Γ *is a consistent extension of Peano arithmetic of degree* $\mathbf{g}$*, the class* $\{\mathbf{a}|\mathbf{a} \leq \mathbf{g}\}$ *is a basis for* $\mathcal{P}_{\mathrm{cb}}$*.*

Theorem 5.2.3. *(Solovay, [unpublished])* $\mathcal{D}_{\mathrm{PA}}$ *is closed upwards.*

For a proof, see [Odi92] or [DH10]. This gives a positive answer to the according question in [JS72b].

Corollary 5.2.4. $\mathcal{D}_{\mathrm{PA}}$ *is an upper semilattice, but not a lattice.*

Proof. By Theorem 5.2.3, $\mathcal{D}_{\mathrm{PA}}$ is closed against the least upper bound for a pair of degrees in the upper semilattice of all Turing degrees, which then must also be the least upper bound among the PA degrees. But by applying the Minimal Pair Basis Theorem to the computably bounded Π^0_1 class of complete extensions of Peano arithmetic and any given PA degree $\mathbf{a}$, we get a PA degree $\mathbf{b}$ such that the only lower bound in the partial order of degrees is $\mathbf{0}$, wich is not PA, because Peano arithmetic is essentially undecidable. □

Recall the definition of the class of separating sets for a pair of c.e. sets.

Definition 5.2.2. We define some particular class of separating sets.
$K_0 := \{e \mid \varphi_e(e) = 0\}$
$K_1 := \{e \mid \varphi_e(e) = 1\}$
$U := S(K_0, K_1)$

Obviously, K_0 and K_1 are disjoint c.e. sets, hence U is a computably bounded Π^0_1 class.

We now state some very interesting characterizations of the PA degrees. Item iv) is added for the sake of completeness. For how that statement is to be understood, see [JS72a]. We prove the most important implications.

Theorem 5.2.5. *(Characterization of the PA degrees) Let* $\mathbf{a}$ *be any degree. Then the following are equivalent.*

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

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

iii) $\mathbf{a}$ *is the degree of a complete extension of Peano arithmetic.*

iv) $\mathbf{a}$ *is the degree of a countable non-standard model of Peano arithmetic.*

v) $\{\mathbf{d} | \mathbf{d} \leq \mathbf{a}\}$ *is a basis for* $\mathcal{P}_{\text{cb}}$.

vi) $\{\mathbf{d} | \mathbf{d} < \mathbf{a}\}$ *is a basis for* $\mathcal{P}_{\text{cb}}$.

vii) $\mathbf{a}$ *computes a separating set for a recursively inseparable pair of c.e. sets.*

viii) $\mathbf{a}$ *is in the degree spectrum of the* Π^0_1 *class* U.

Proof. By definition, i) and iii) are equivalent. Clearly, iii) implies ii). By the Scott Basis Theorem, ii) implies v). The Shoenfield construction in Theorem 4.2.6 improves v) to vi). Because $\mathcal{D}_{\text{PA}}$ is the spectrum of a Π^0_1 class in $\mathcal{P}_{\text{cb}}$ and $\mathcal{D}_{\text{PA}}$ is upward closed, vi) implies i). That establishes the equivalence of statements i), ii, iii, v) and vi). For the exact meaning of iv) and its equivalence to v), see [JS72a]. The implication of viii) by vii) is trivial. Because every consistent extension of Peano arithmetic is a separating set for the inseparable consequences and refutations of Peano arithmetic, ii) implies vii). A proof for the two remaining implications can be found in [DH10]. □

Remark. Further characterization can be found in [DDS10] and [DH10]. A Π^0_1 class U with the property of equality of v) and viii) is called *universal* Π^0_1 class by some authors ([DH10], [CJ99]).

The equality of vi) to vii) and viii) provides a mere recursion theoretical approach to the basis theorems in vi).

Corollary 5.2.6.

a) $\mathbf{0} \notin \mathcal{D}_{\mathrm{PA}}$

b) There is a low PA degree.

c) $\mathbf{0}' \in \mathcal{D}_{\mathrm{PA}}$

d) No incomplete c.e. degree is PA. ([ST60])

e) If $\mathbf{d}$ *is a degree such that* $\mathbf{0}' < \mathbf{d}$*, then* $\mathbf{d}$ *is PA. ([ST60])*

f) There is a class of mutually incomparable PA degrees of cardinality $2^{\aleph_0}$*.*

Proof. The first item is implied by the (improved) Gödel's Incompleteness Theorem. That there is a low PA degree follows from the Low Basis Theorem and that $\mathcal{D}_{\mathrm{PA}}$ is the degree spectrum of a computably bounded Π^0_1 class. That implies the weaker result that there are PA degrees below $\mathbf{0}'$, announced in [ST60]. Item d) follows from Theorem 4.2.27 in combination with item v) of the characterization of PA degres. Then item c) must hold, because of the C.E. Basis Theorem. The statement in e) follows from item c) combined with Theorem 5.2.3. The last item follows from 4.2.32. It implies that there is a PA degree that is imcomparable to $\mathbf{0}'$, wich was announced in [ST60]. □

In [KL10], the authors prove that $\mathcal{D}_{\mathrm{PA}}$ is minimal in the lattice of degree spectra of computably bounded Π^0_1 classes ordered by inclusion. This can be restated in our terms as follows. We omit the proof.

Theorem 5.2.7 ([KL10])**.** *Every computably bounded* Π^0_1 *class* P *is empty or has a member of degree that is not PA or it already has a member of every PA degree.*

Shoenfield's construction has given us item vi) of the characterization of PA degrees, which enforces item v). While the according amplification of the Kreisel Basis Theorem to the Kreisel-Shoenfield Basis Theorem may be regarded as somewhat insignificant ever since the Low Basis Theorem was proved, Shoenfield's construction has greater impact in the context of PA degrees.

Corollary 5.2.8. *For every PA degree* $\mathbf{p}$ *there are PA degrees* $(\mathbf{p}_n)_{n\in\mathbf{N}}$ *such that* $\mathbf{p}_0 < \mathbf{p}$ *and* $\mathbf{p}_{i+1} < \mathbf{p}_i$ *for all* $i \in \mathbb{N}$*.*

Proof. Let $\mathbf{p}$ be a PA degree. By Corollary 5.2.5.vi), $\{\mathbf{d}|\mathbf{d} < \mathbf{a}\}$ is a basis for the computably bounded Π^0_1 classes. Since the class of consistent extensions of Peano arithmetic is itself a computably bounded Π^0_1 class, there must hence be another PA degree properly below $\mathbf{p}$. The claim follows by induction. □

This statement seems to be lacking in the present literature.

Together with Theorem 5.2.3 and Corollary 5.2.6.f, this gives a wide range of degrees that are PA. But in [JS72b], the authors employ a measure for 2^ω and show, that the measure of the union of all PA degrees is equal to 0. They also find that this class is meagre in the topological sense. Note that we can consider the PA degrees classes of sets.

Compare Corollary 5.2.8 to the following result, achieved by Jockusch and Soare by much more intricate means. We omit the proof.

Theorem 5.2.9 ([JS72b])**.** *If* $\mathbf{a}$ *is a PA degree, then any countable partially ordered set is embeddable in the upper semilattice of degrees below* $\mathbf{a}$

This is much stronger on the one hand, as there is no constraint on the partial order one wishes to embed below a given PA degree. But on the other hand, the degrees involved in representing that partial order might not be chosen to be PA. So, one can derive from Jockusch and Soare's theorem the following.

Corollary 5.2.10 ([ST60]). *There is no minimal degree that is PA.*

But one can not derive the following, stronger corollary from Theorem 5.2.9. It is a result the author of this thesis was unable to find in the present literature.

Corollary 5.2.11. *There is no minimal degree in the partial order of PA degrees.*

Proof. This is immediate from Corollary 5.2.8. □

Bibliography

[Cen10] Douglas Cenzer, *Effectively Closed Sets, draft*, `http://www.math.ufl.edu/~cenzer/research_html/book2.pdf`, May 2010.

[CJ99] Douglas Cenzer and Carl Jockusch, *Π^0_1 classes - structure and applications*, Computability theory and its applications, Proceedings of a 1999 AMS-IMS-SIAM Joint Summer Research Conference (Boulder, CO) (Peter Cholak, ed.), American Mathematical Society, 1999, pp. 39–59.

[CR98] Douglas Cenzer and Jeffrey Remmel, *Handbook of Recursive Mathematics*, vol. 2, ch. 13: Π^0_1 classes in mathematics, pp. 623–821, Elsevier, Amsterdam, New York, 1998.

[CR03] ______, *Index sets for ω-languages*, Mathematical Logic Quarterly **49** (2003), no. 1, 22–33.

[DDS10] David Diamondstone, Damir Dzhafarov, and Robert Soare, *Π^0_1 classes, Peano arithmetic, randomness, computable domination*, Notre Dame Journal of Formal Logic **51** (2010), no. 1, 127–159.

[DH10] Rod G. Downey and Denis R. Hirschfeldt, *Algorithmic Randomness and Complexity*, Theory and Applications of Computability, ch. 2.19: Π^0_1 and Σ^0_1 classes, 2.21: PA degrees, pp. 72–82, 84–87, Springer, New York ; Heidelberg, 2010.

[DJ09] Damir Dzhafarov and Carl Jockusch, *Ramsey's theorem and cone avoidance*, The Journal of Symbolic Logic **74** (2009), no. 2, 557–578.

[Dzh12] Damir Dzhafarov, *private communication*, February 2012.

[Ehr61] A. Ehrenfeucht, *Separable theories*, Bulletin de l'Académie Polonaise des Sciences **9** (1961), no. 1, 17–19.

[JLR91] Carl Jockusch, Andrew Lewis, and Jeffrey Remmel, Π^0_1 *classes and Rado's selection principle*, The Journal of Symbolic Logic **56** (1991), no. 2, 684–693.

[JS72a] Carl Jockusch and Robert Soare, *Degrees of members of* Π^0_1 *classes*, Pacific Journal of Mathematics **40** (1972), no. 3, 605–616.

[JS72b] ———, Π^0_1 *classes and degrees of theories*, Transactions of the American Mathematical Society **173** (1972), 33–56.

[KL10] Thomas Kent and Andrew Lewis, *On the degree spectrum of a* Π^0_1 *class*, Transactions of the American Mathematical Society **362** (2010), no. 10, 5283–5319.

[Kre59] Georg Kreisel, *Analysis of the Cantor-Bendixson Theorem by means of the analytic hierarchy*, Bulletin de l'Académie Polonaise des Sciences **7** (1959), no. 10, 621–626.

[Odi92] Piergiorgio Odifreddi, *Classical Recursion Theory*, Studies in Logic and the Foundation of Mathematics, vol. 125, North-Holland, Amsterdam, 1992.

[Sco62] Dana Scott, *Algebras of sets binumerable in complete extensions of arithmetic*, Recursive function theory (Providence, RI) (J.C.E. Dekker, ed.), Proceedings of Symposia in Pure Mathematics, vol. V, American Mathematical Society, American Mathematical Society, 1962, pp. 117–121.

[Sho60] Joseph R. Shoenfield, *Degrees of models*, The Journal of Symbolic Logic **25** (1960), no. 3, 233–237.

[ST60] Dana Scott and Stanley Tennenbaum, *On the degrees of complete extensions of arithmetic*, Notices of the American Mathematical Society **7** (1960), 242–243.

Erklärung

Hiermit versichere ich, dass ich meine Arbeit selbstständig unter Anleitung verfasst habe, dass ich keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe, und dass ich alle Stellen, die dem Wortlaut oder dem Sinne nach anderen Werken entlehnt sind, durch die Angabe der Quellen als Entlehnungen kenntlich gemacht habe.

Datum Unterschrift