Notre Dame Journal of Formal Logic
Volume 49, Number 3, 2008

Classifying the Branching Degrees in the Medvedev Lattice of Π^0_1 Classes

Christopher P. Alfeld

Abstract A Π^0_1 class can be defined as the set of infinite paths through a computable tree. For classes P and Q, say that P is Medvedev reducible to Q, $P \leq_M Q$, if there is a computably continuous functional mapping Q into P. Let $\mathcal{L}_M$ be the lattice of degrees formed by Π^0_1 subclasses of 2^ω under the Medvedev reducibility. In "Non-branching degrees in the Medvedev lattice of Π^0_1 classes," I provided a characterization of nonbranching/branching and a classification of the nonbranching degrees. In this paper, I present a similar classification of the branching degrees. In particular, P is separable if there is a clopen set C such that $P \cap C \neq \varnothing \neq P \cap C^c$ and $P \cap C \perp_M P \cap C^c$. By the results in the first paper, separability is an invariant of a Medvedev degree and a degree is branching if and only if it contains a separable member. Further define P to be hyperseparable if, for all such C, $P \cap C \perp_M P \cap C^c$ and totally separable if, for all $X, Y \in P$, $X \perp_T Y$. I will show that totally separable implies hyperseparable implies separable and that the reverse implications do not hold, that is, that these are three distinct types of branching degrees. Along the way I will show some related results and present a combinatorial framework for constructing Π^0_1 classes with priority arguments.

1 Introduction

A Π^0_1 class of 2^ω is a subset of 2^ω which satisfies a certain notion of computability, namely, that there is a tree in $2^{<\omega}$ which is computable and whose paths form the Π^0_1 class. Cenzer and Jockusch [4] provide a good overview of Π^0_1 classes. Applications of Π^0_1 classes in mathematics can be found in [5].

A common view of Π^0_1 classes is as mass problems. That is, view a class P as representing the set of solutions to a problem in computable mathematics. In such a context, a natural question is whether solving one problem allows us to solve another.

Received May 31, 2007; accepted March 21, 2008; printed June 6, 2008
2000 Mathematics Subject Classification: Primary, 03D30
Keywords: Π^0_1 classes, Medvedev lattice, branching degree
 10.1215/00294527-2008-009

That is, if P and Q represent problem, then saying that P solves Q is saying that we can computably turn members of P into members of Q. When this reduction is uniform, this is exactly the Medvedev reduction. Specifically, $P \geq_M Q$, if there exists a computable functional Φ such that, for every $X \in P$, $\Phi(X) \in Q$. A useful property of such functionals is that they are continuous in the normal topology of 2^ω.

This reduction induces, in general, a lattice of subsets of 2^ω and, in particular, a lattice of Π^0_1 subsets of 2^ω. This lattice has recently been studied by Cenzer and Hinman [3], Binns [2], and Simpson [8], [9], and [10]. We will denote the lattice by $\mathcal{L}_M$. It is distributive with minimum and maximum element.

As convention, say that a degree has a certain property if there exists a member of the degree with that property.

This paper is the second half of the research track begun in "Non-Branching Degrees in the Medvedev Lattice of Π^0_1 classes" [1]. There I studied the nonbranching degrees of $\mathcal{L}_M$, that is, degrees which were not the meet (greatest lower bound) of two other degrees. The results are listed in Section 3. To summarize, I defined two properties of classes, *inseparable* and *hyperseparable*, and showed the following interaction, also considering an existing property *homogeneous* [3]:

$$\text{Non-Branching} \Leftrightarrow \text{Inseparable} \Leftarrow \text{Hyperinseparable} \Leftarrow \text{Homogeneous}. \tag{1}$$

Further, I showed that no unstated implications exist, that is, that these are three distinct classes of nonbranching degrees. The reader may find it useful to look over Section 3 at this point, but it is not necessary.

This paper follows a similar program with regard to branching degrees. I define *separable* as the converse of inseparable, define *hyperseparable* in a similar way to hyperinseparable, and also consider the unnamed but previously known condition which I call *totally separable*. I arrive at

$$\text{Branching} \Leftrightarrow \text{Separable} \Leftarrow \text{Hyperseparable} \Leftarrow \text{Totally Separable}. \tag{2}$$

And no unstated implications exist.

A much needed definition will be the following.

Definition 1.1 For a Π^0_1 class P and clopen set C, C is good for P if

$$P \cap C \neq \varnothing \neq P \cap C^c. \tag{3}$$

A class P is separable if there exists a C good for P such that $P \cap C \perp_M P \cap C^c$, that is, if we can split P into two incomparable clopen subclasses. From the results in [1], we immediately have that a degree is separable if and only if it is nonbranching. A class P is hyperseparable if every clopen set C good for P splits P into incomparable clopen subclasses. In [6], Jockusch and Soare showed that there exists a class P such that all members of P are pairwise Turing incomparable. Call such a P totally separable. It is straightforward to show that totally separable implies hyperseparable.

The primary results of this paper are to show the existence of degrees which are separable and not hyperseparable, and degrees which are hyperseparable and not totally separable. Along the way, we will prove some related results.

An additional contribution of this paper is the notion of tree lifes. Tree lifes are a formalization of a combinatorial method for building Π^0_1 classes with priority arguments. The basic method is found in, among others, [1] and [3]. There are no difficult proofs in the discussion of tree lifes but the notions and basic results encapsulate some of the common behavior of such constructions. Tree lifes will be used in Section 9 to show the existence of a hyperseparable and not totally separable degree.

Section 2 will present the necessary definitions and results from the theory of Π^0_1 classes. Section 3 will review the major definitions and results of [1]. Section 4 will present tree lifes. Sections 5, 6, and 7 will present and discuss separable, hyperseparable, and totally separable degrees, respectively. Section 8 will separate separable and hyperseparable and present some structural results about separable and not hyperseparable degrees. Finally, Section 9 will separate hyperseparable and totally separable.

2 Basic Definitions and Theory

See Soare [11] or Rogers [7] for an overview of the concepts and theory of computability theory.

For a string $\sigma \in 2^{<\omega}$, denote its length by $|\sigma|$. Denote the initial substring relation by $\prec$ and concatenation of σ and τ by $\sigma \frown \tau$. Denote the empty string by $\varnothing$, the string of a single 0 by 0 and of a single 1 by 1. Denote truncation by $\sigma \upharpoonright n$. For $X \in 2^\omega$, say $\sigma \prec X$ if $X \upharpoonright |\sigma| = \sigma$; that is, σ is an initial segment of X.

A tree is a subset of $2^{<\omega}$ which is closed downward under $\prec$. Members of a tree will frequently be referred to as nodes. If $\sigma \prec \tau$, then say that τ is a descendant of σ. For a tree $\mathbb{T}$, denote the set of infinite paths through $\mathbb{T}$ by $[\mathbb{T}]$ and the set of extendible nodes, the nodes with descendants of arbitrary length, by $\mathrm{Ext}(\mathbb{T})$. A dead end is a node that is not extendible. For a string σ, define $\sigma \frown \mathbb{T} = \{\sigma \frown \tau : \tau \in \mathbb{T}\}$.

There are several equivalent definitions of a Π^0_1 class. For our purposes, a Π^0_1 class P is a nonempty subset of 2^ω such that there exists a computable tree $\mathbb{P}$ with $P = [\mathbb{P}]$. Denote the tree of initial substrings of members of P by $\mathcal{T}_P$. Note that $[\mathcal{T}_P] = P$ and $\mathrm{Ext}(\mathbb{P}) = \mathcal{T}_P$ for any $\mathbb{P}$ which generates P. Define $\sigma \frown P = [\sigma \frown \mathcal{T}_P]$.

The set 2^ω can be viewed as the Cantor set in $\mathcal{R}$. We will use the resulting subspace topology on 2^ω. Namely, define $I(\sigma) = \{X \in 2^\omega : \sigma \prec X\}$. A clopen subset will be a finite union of such intervals. To simplify, say that $\sigma \in C$ if there exists $X \in C$ with $\sigma \prec X$. Similarly, for a tree $\mathbb{T}$, define $\mathbb{T} \cap C = \{\sigma \in \mathbb{T} : \sigma \in C\}$.

We need a concept of a computable map between classes. There are several (often equivalent) approaches to this. We use the following.

Definition 2.1 A partial computable function $\varphi\colon 2^{<\omega} \to 2^{<\omega}$ is a *tree map* if it satisfies the following properties:

$$\mathrm{dom}(\varphi) \text{ is a tree,} \tag{4}$$

$$\forall \sigma, \tau \in \mathrm{dom}(\varphi)\big(\sigma \preceq \tau \Rightarrow \varphi(\sigma) \preceq \varphi(\tau)\big), \tag{5}$$

$$\forall X \in [\mathrm{dom}(\varphi)] \forall n \exists m \big(|\varphi(X \upharpoonright m)| > n\big). \tag{6}$$

A *computably continuous functional* is a function $\Phi\colon 2^\omega \to 2^\omega$ such that there exists a tree map φ with $\Phi(X) = \bigcup_n \varphi(X \upharpoonright n)$.

To say $\Phi\colon Q \to P$ means that there is a total tree map φ with $\varphi(\mathcal{T}_Q) \subseteq \mathcal{T}_P$.

Define $P \leq_M Q$, said P is Medvedev below Q, if there exists $\Phi\colon Q \to P$. Note that $\leq_M$ induces an equivalence relation, $\equiv_M$. Denote the resulting lattice of equivalent classes by $\mathcal{L}_M$. Denote the bottom degree by $\mathbf{0}$ and the top degree by $\mathbf{1}$. For a class P, denote the degree of P by $\deg(P)$.

The following lemma states that, in the $\leq_M$ case, we can assume Φ, and thus φ, to be total. For a proof, see [3].

Lemma 2.2 *Let P and Q be Π^0_1 classes such that $P \leq_M Q$. Then there exists a total computable functional $\Phi : 2^\omega \to 2^\omega$ such that $\Phi(Q) \subseteq P$.*

An immediate but much used lemma is the following.

Lemma 2.3 *Let Q and P be Π^0_1 classes with $Q \subseteq P$. Then $Q \geq_M P$.*

Proof The identity function serves as a witness. $\square$

Finally, we need to be able to enumerate the Π^0_1 classes.

Lemma 2.4 *If $\mathbb{P}$ is a co-c.e. tree, then there exists a computable tree $\mathbb{Q}$ such that $[\mathbb{P}] = [\mathbb{Q}]$. Furthermore, we can effectively find $\mathbb{Q}$ from $\mathbb{P}$.*

Proof Sketch Let $\{A_s\}_{s \in \omega}$ be an enumeration of $2^{<\omega} \setminus \mathbb{P}$ and $\mathbb{Q} = \{\sigma : \forall \tau \preceq \sigma(\tau \notin A_{|\sigma|})\}$. $\square$

Definition 2.5 Define P_e to be $[\mathbb{T}_e]$ where $\mathbb{T}_e$ is the eth co-c.e. tree.

3 Nonbranching Review

For discussion, proofs, and related results, see [1].

Definition 3.1 A Π^0_1 class P is inseparable if, for all clopen sets C good for P, $P \cap C \leq_M P \cap C^c$ or $P \cap C \geq_M P \cap C^c$.

Theorem 3.2 ([1]) *If P is inseparable and $Q \equiv_M P$, then Q is inseparable.*

Theorem 3.3 ([1]) *A degree $\boldsymbol{a}$ is nonbranching if and only if it is inseparable.*

Definition 3.4 A Π^0_1 class P is hyperinseparable if, for all clopen sets C good for P, $P \cap C \equiv_M P \cap C^c$.

Equivalently, P is hyperinseparable if, for all clopen sets C good for P, $P \cap C \equiv_M P$.

Not every member of a hyperinseparable degree is hyperinseparable. However, every member has a hyperinseparable core. We will see a very similar result later in the study of totally separable branching degrees, namely, Theorem 7.3.

Theorem 3.5 ([1]) *If $\boldsymbol{a}$ is hyperinseparable and $P \in \boldsymbol{a}$, then there exists $Q \subseteq P$ with $Q \equiv_M P$ and Q hyperinseparable.*

The next theorem shows that the concepts of inseparable and hyperinseparable are distinct.

Theorem 3.6 ([1]) *For degrees $\boldsymbol{a}$ and $\boldsymbol{b}$ with $0 <_M \boldsymbol{b}$, there exists a degree $\boldsymbol{c}$ such that $0 <_M \boldsymbol{c} <_M \boldsymbol{b}$ and $\boldsymbol{c}$ is inseparable and not hyperinseparable.*

In [3], Cenzer and Hinman introduce homogeneous degrees and show that they are nonbranching. It is straightforward to show that homogeneous implies hyperinseparable. See [1] for details.

Definition 3.7 ([3], Definition 8) A tree $\mathbb{P}$ is *homogeneous* if

$$\forall \sigma, \tau \in \mathbb{P}\ \forall i \in 2\big[|\sigma| = |\tau| \Rightarrow (\sigma^\frown i \Leftrightarrow \tau^\frown i)\big].$$

A class P is *homogeneous* if $\mathcal{T}_P$ is.

As with hyperinseparable and inseparable, I separated homogeneous and hyperinseparable.

Theorem 3.8 ([1]) *There exists a degree which is hyperinseparable and not homogeneous.*

4 Tree Lifes

There are many ways to construct Π^0_1 classes via a priority argument. This section formalizes a method in which a tree is enumerated along with a total computable function which tightly bounds the length of nodes added at each stage. The combination of enumeration and length function ensures that the tree is computable and thus produces a Π^0_1 class. This technique was seen in [1] and in the literature, for example, in [3]. The formalization described below will be used to prove Theorem 9.2.

Definition 4.1 A finite tree $L \subseteq 2^{<\omega}$ is a strict tree if all dead ends are of the same length (necessarily maximal). The length of L, denoted $l(L)$, is the length of the dead ends. The set of dead ends is denoted $D(L)$.

Definition 4.2 For strict trees L and M, M is a growth of L if $l(M) \geq l(L)$ and

$$\forall \sigma \in M \setminus L \exists \tau \in D(L)[\sigma \succ \tau]. \tag{7}$$

Call a leaf of maximal length a living leaf. Then a growth can be characterized by two conditions: (1) the length cannot decrease, that is, at least one living leaf must survive, and (2) any additional nodes must extend living leaves. Thus, a valid growth may consist of extending living leaves, pruning part of the tree, or a combination of both.

Definition 4.3 A tree life is a sequence of strict trees $\{L_s : s \in \omega\}$ such that for all $s > 0$, L_s is a growth of L_{s-1} and $\lim_s l(L_s) = \infty$. A tree life is computable if there exists a total computable function f such that $f(s) = L_s$.

To simplify notation we will hereafter omit ": $s \in \omega$"; that is, we will simply write $\{L_s\}$.

Define $\lim_s L_s = \{\sigma \in 2^{<\omega} : \exists t \forall n > t[\sigma \in L_n]\}$.

Lemma 4.4 *For any tree life $\{L_s\}$, any s, and any $\sigma \in L_{s+1} \setminus L_s$, $l(L_s) < |\sigma| \leq l(L_{s+1})$.*

Proof Fix $\sigma \in L_{s+1} \setminus L_s$. Then, as L_{s+1} is a growth of L_s, there must be some $\tau \in D(L_s)$ with $\tau \prec \sigma$. Thus $|\sigma| > |\tau| = l(L_s)$. That $|\sigma| \leq l(L_{s+1})$ is immediate. $\square$

Corollary 4.5 *For any tree life $\{L_s\}$, any s, and any $\sigma \in L_s$, if $\sigma \notin L_{s+1}$, then for all $t > s$, $\sigma \notin L_t$.*

Observe that this corollary implies that $\lim_s L_s$ is well defined; it is a d.c.e. set.

Lemma 4.6 *For a tree life $\{L_s\}$, $[\lim_s L_s] = \left[\bigcup_s L_s\right]$.*

Proof The inclusion $\supseteq$ is immediate. For $\subseteq$, fix $X \in \left[\bigcup_s L_s\right]$. Fix n and let $\sigma = X \restriction n$ and s be such that $\sigma \in L_s$. As σ has descendants of arbitrary length, σ must be in every L_t for $t > s$. As n was arbitrary, $X \in [\lim_s L_s]$. $\square$

Lemma 4.7 *For a computable tree life* $\{L_s\}$, $\bigcup_s L_s$ *is computable and* $\lim_s L_s$ *is co-c.e.*

Proof Observe that $l(L_s)$ is a computable function. Fix $\sigma \in \bigcup_s L_s$ and t such that $l(L_t) \geq |\sigma|$. Then $\sigma \in \bigcup_s L_s$ if and only if $\sigma \in L_t$. Thus $\bigcup_s L_s$ is computable. As $\lim_s L_s$ is the difference of a computable set $\left(\bigcup_s L_s\right)$ and a c.e. set (the nodes that leave), it is co-c.e. □

Corollary 4.8 *For a computable tree life* $\{L_s\}$, $[\lim_s L_s]$ *is a* Π^0_1 *class.*

Proof By Lemma 4.7, $\bigcup_s L_s$ is a computable tree. By Lemma 4.6, $[\lim_s L_s] = \left[\bigcup_s L_s\right]$ and thus is a Π^0_1 class. □

Having defined the basic construction and shown that it results in computable trees we now define some growth operations which are effective.

Definition 4.9 For a nonempty strict tree L, the single extension of L, denoted $\mathrm{extend}(L)$, is defined by

$$\mathrm{extend}(L) = L \cup \{\sigma ^\frown i : \sigma \in D(L), i \in 2\}. \tag{8}$$

Define $\mathrm{extend}(\varnothing) = \{\varnothing, 0, 1\}$.

Note that $l(\mathrm{extend}(L)) = l(L) + 1$.

Definition 4.10 For a strict tree L and $\sigma \in L$, the trim of L by σ, denoted $\mathrm{trim}(L, \sigma)$ is defined by

$$\mathrm{trim}(L, \sigma) = \{\tau \in L : \exists \nu \succeq \tau[\nu \perp \sigma]\}. \tag{9}$$

Note that $\mathrm{trim}(L, \sigma)$ is L with σ and all descendants removed. We also remove ancestors of σ which do not lead to other non-σ descendants to ensure that $\mathrm{trim}(L, \sigma)$ is a strict tree. Note that $l(\mathrm{trim}(L, \sigma)) \leq l(L)$.

Lemma 4.11 *For a strict tree* L*,* $\mathrm{extend}(L)$ *is a strict tree and a growth of* L*.*

Proof Fix $\tau \in D(\mathrm{extend}(L))$. By definition, $\tau = \sigma ^\frown i$ for $\sigma \in D(L)$ and $i \in 2$. Then $|\sigma| = l(L)$, $|\tau| = l(L) + 1$. As τ was arbitrary, $\mathrm{extend}(L)$ is a strict tree.

Fix $\tau \in \mathrm{extend}(L) \setminus L$. Then $\tau = \sigma ^\frown i$ for $\sigma \in D(L)$ and $i \in 2$, and σ witnesses that $\mathrm{extend}(L)$ is a growth. □

Lemma 4.12 *For a strict tree* L *and* $\sigma \in L$*,* $\mathrm{trim}(L, \sigma)$ *is a strict tree and either empty or a growth of* L*.*

Proof Let $M = \mathrm{trim}(L, \sigma)$ and assume M has a dead end α with $|\alpha| < l(L)$. As $|\alpha| < l(L)$ there is an immediate successor of α, $\beta \in L$. As $\beta \notin M$, β is comparable with σ but α is not, a contradiction. Thus M is a strict tree.

As trim only removes paths, if M is not empty, then it contains a path of length $l(L)$. Thus M is a growth of L. □

5 Separable Degrees

We define separability as the inverse of inseparability. Separable degrees are those whose members can be split into incomparable clopen subclasses.

Definition 5.1 A Π^0_1 class P is *separable* if there exists a clopen set C good for P such that $P \cap C \perp_M P \cap C^c$.

The primary results move over directly.

Theorem 5.2 *Separability is an invariant of a Medvedev degree; that is, if $P \equiv_M Q$ and P is separable, then Q is separable.*

Proof This theorem is the contrapositive of Theorem 3.2. □

Corollary 5.3 *A degree $\boldsymbol{a}$ is separable if and only if $\boldsymbol{a}$ is branching; that is, there exists $\boldsymbol{b} > \boldsymbol{a}$, $\boldsymbol{c} > \boldsymbol{a}$ with $\boldsymbol{a} = \boldsymbol{b} \wedge \boldsymbol{c}$.*

Proof This is the contrapositive of Theorem 3.3. □

6 Hyperseparable Degrees

Definition 6.1 A Π^0_1 class P is *hyperseparable* if for all clopen sets C good for P, $P \cap C \perp_M P \cap C^c$.

Observe that hyperseparable implies separable. As in the case of hyperinseparability, it is too much to hope that this would be invariant.

Theorem 6.2 *For any Π^0_1 class P, there exists a class Q, with $Q \equiv_M P$ and Q not hyperseparable.*

Proof Take $Q = P \wedge P$ and observe that $C = I(0)$ contradicts hyperseparability. □

Additional results about hyperseparable degrees, in the context of nonseparability, can be found in Section 8.

7 Totally Separable Degrees

As with homogeneous in the nonbranching case, there is a condition in the literature which is stronger than hyperseparable. I was unable to find a name for it, so I refer to it as totally separable.

Definition 7.1 A Π^0_1 class P is *totally separable* if for all $X, Y \in P$, $X \perp_T Y$.

Note that totally separable implies hyperseparable.

Jockusch and Soare proved the existence of a totally separable class.

Theorem 7.2 ([6]) *There exists a totally separable class.*

Totally separable is a very strong condition which enforces a great deal of structure on the other members of the degree. The following is similar to Theorem 3.5; that is, it shows that members of a totally separable degree contain a totally separable core.

Theorem 7.3 *Let P be a totally separable Π^0_1 class and Q a Π^0_1 class with $Q \equiv_M P$. Then there exists a Π^0_1 class $R \subseteq Q$ such that $R \equiv_M Q$ and R is totally separable. Furthermore, if $\Phi : P \to Q$ and $\Psi : Q \to P$ witness $Q \equiv_M P$, then $\Phi : P \to R$ and $\Psi : R \to P$ are bijections.*

Proof Let $R = \Phi(P)$. By Lemma 2.3, $R \geq_M Q$. The function $X \mapsto \Psi(X) \mapsto \Phi(\Psi(X))$ witnesses $Q \geq_M R$. Thus $R \equiv_M Q$.

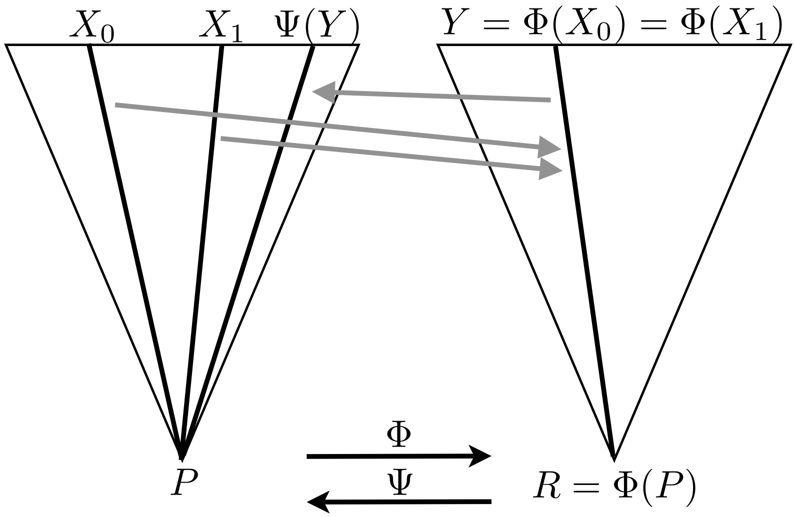

Figure 1 Theorem 7.3: $\Phi : P \to R$ injective.

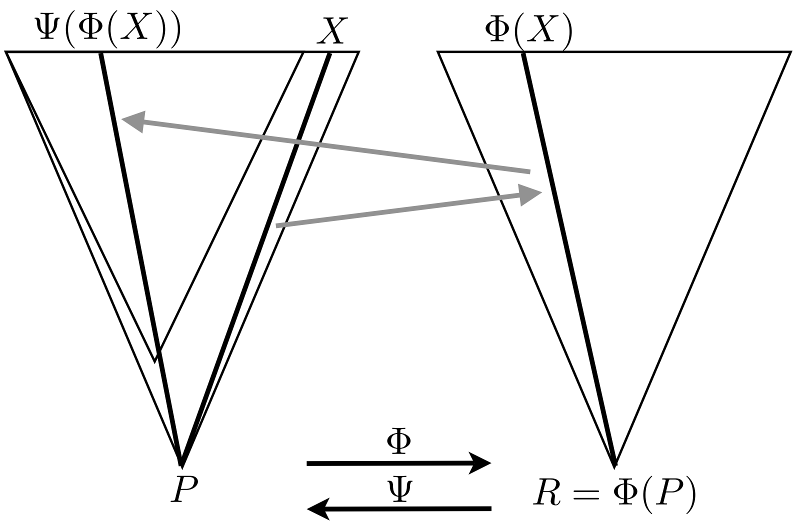

Figure 2 Theorem 7.3: $\Psi : R \to P$ surjective

By definition $\Phi : P \to R$ is surjective. Assume Φ is not injective and fix X_0, X_1 in P with $\Phi(X_0) = \Phi(X_1) = Y$. Assume $\Psi(Y) \neq X_0$ (it must differ from one of X_0 and X_1). But $\Psi(Y) \leq_T X_0$ as $X_0 \mapsto Y \mapsto \Psi(Y)$, a contradiction of P being totally separable. Thus $\Phi : P \to R$ is injective and thus bijective. See Figure 1.

Assume $\Psi : R \to P$ is not surjective. Fix $X \in P \backslash \Psi(R)$. Then $\Psi(\Phi(X)) \in \Psi(R)$ and thus not equal to X, but is Turing reducible from X, a contradiction. Thus $\Psi : R \to P$ is surjective. See Figure 2.

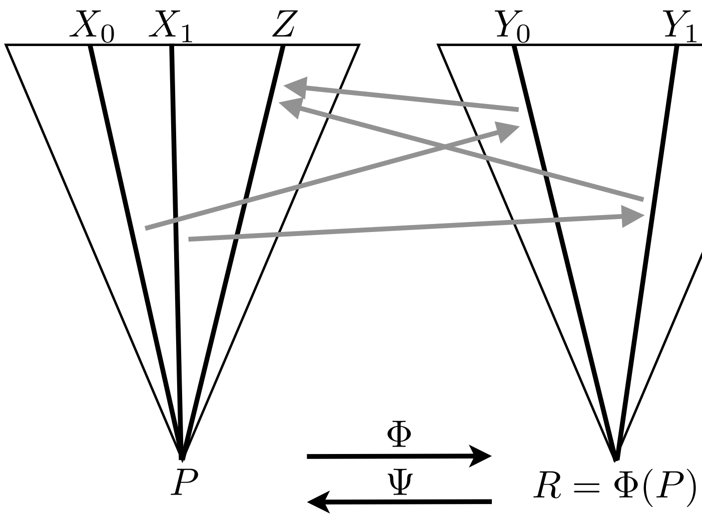

Figure 3 Theorem 7.3: $\Psi : R \to P$ injective

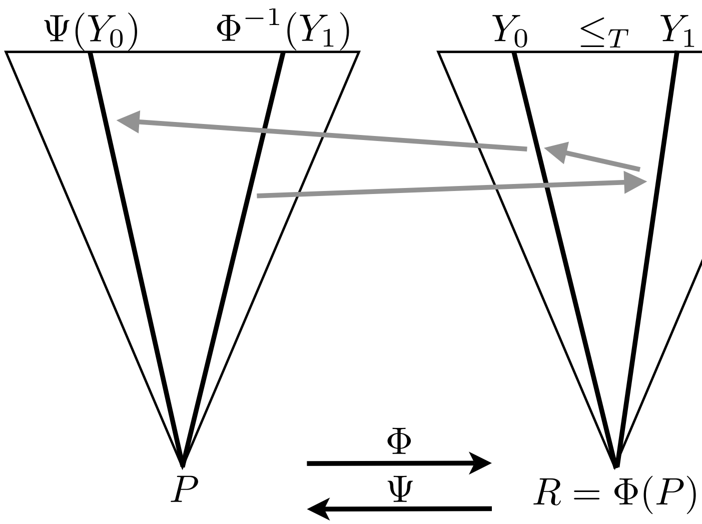

Figure 4 Theorem 7.3: R totally separable.

Assume $\Psi : R \to P$ is not injective. Fix Y_0 and Y_1 in R such that $\Psi(Y_0) = \Psi(Y_1) = Z$. As Φ is bijective there exist X_0, X_1 in P with $X_0 \neq X_1$, $Y_0 = \Phi(X_0)$, and $Y_1 = \Phi(X_1)$. Assume $Z \neq X_0$ (the other case is symmetric). Then $Z \leq_T X_0$ as $X_0 \mapsto Y_0 \mapsto Z$, a contradiction. Thus Ψ is injective and thus bijective. See Figure 3.

Assume R is not totally separable and fix Y_0 and Y_1 in R with $Y_0 \neq Y_1$ and $Y_0 \leq_T Y_1$. Let $X_0 = \Psi(Y_0)$ and $X_1 = \Phi^{-1}(Y_1)$. As Φ^{-1} is a bijection, $X_0 \neq X_1$.

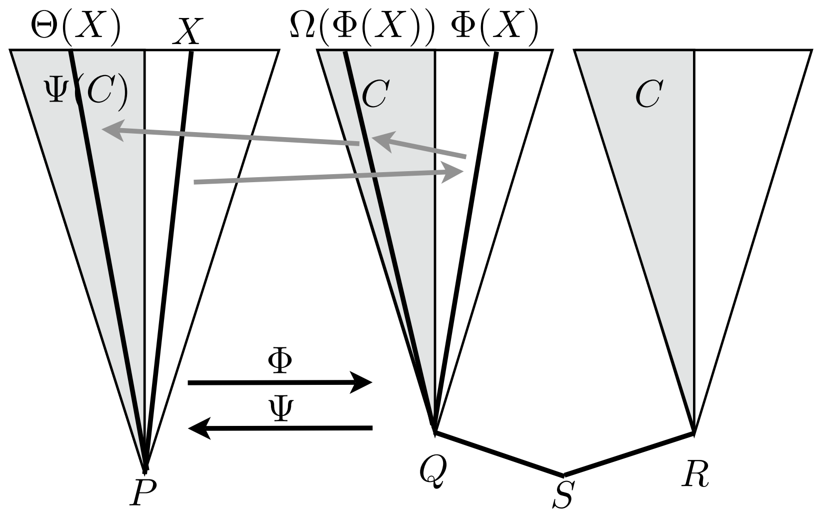

Figure 5 Theorem 8.1: $\Phi(X) \in C^c$

And $X_0 \leq_T Y_0$ via Ψ; $Y_0 \leq Y_1$ by assumption; and $Y_0 \leq X_1$ via Φ. Thus $X_0 \leq_T X_1$, a contradiction. See Figure 4. Thus R is totally separable. □

Corollary 7.4 *If P and Q are totally separable with $P \equiv_M Q$, then P is computably isomorphic to Q.*

Proof Repeat the proof of Theorem 7.3 with P in place of R. □

The following lemma shows that in the situation of Theorem 7.3, P retracts onto its totally separable core.

Lemma 7.5 *If Q and R are such that $R \subseteq Q$, R is totally separable, and $\Phi : Q \to R$ is a computably continuous functional, then $\Phi(X) = X$ for all $X \in R$; that is, Φ is a retraction.*

Proof If $\Phi(X) = Y \neq X$ for some X in R, then $Y \leq_T X$, a contradiction. □

8 Separable and Not Hyperseparable

Theorem 8.1 *If Q and R are hyperinseparable with $Q \perp_M R$, then* $\deg(Q \wedge R)$ *is separable and not hyperseparable.*

Proof Let $S = Q \wedge R$ and $\mathbf{a} = \deg(S)$. Then S is separable. Consider any $C_Q \subset I(0)$ good for $0^\frown Q$ and $C_R \subset I(1)$ good for $1^\frown R$. By hyperinseparability there is a reduction from $0^\frown Q \cap C_Q$ to $0^\frown Q \cap C_Q^c$ and similarly for C_R. Thus there is a reduction to $C = C_Q \cup C_R$ from C^c witnessing that S is not hyperseparable.

Let P be any class with $P \equiv_M S$. Let $\Phi : P \to S$ and $\Psi : S \to P$ witness $P \equiv_M S$. Let C_Q and C_R be clopen sets satisfying $C_Q \subset I(0)$, $C_R \subset I(1)$, C_Q good for $0^\frown Q$, C_R good for $1^\frown R$, and $\Psi(C_R \cup C_Q)$ good for P. The last requirement can be achieved by choosing C_R and C_Q small enough: fix $\sigma \in \mathcal{T}_R$ long

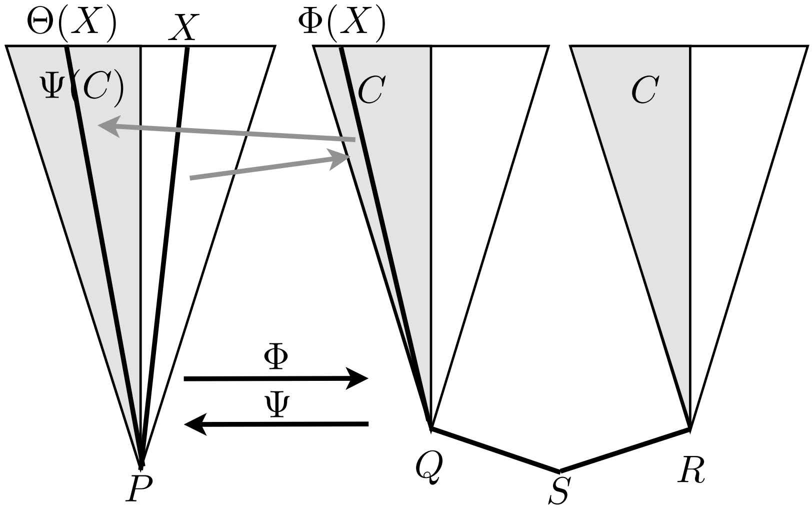

Figure 6 Theorem 8.1: $\Phi(X) \in C$

enough such that there exists $\tau \in \mathcal{T}_S$ with $\varphi(\sigma) \perp \tau$ and let $C_R = I(\sigma)$; similarly for C_Q.

Let $C_S = \Psi(C)$. Fix $\Omega : S \cap C^c \to S \cap C$. Define Θ by

$$\Theta(X) = \begin{cases} \Psi(\Phi(X)) & \text{if } \Phi(X) \in C, \\ \Psi(\Omega(\Phi(X))) & \text{if } \Phi(X) \in C^c. \end{cases} \tag{10}$$

See Figures 5 and 6. Then Θ witnesses $P \cap C_S^c \geq_M P \cap C_S$. So P is not hyperseparable. As P was arbitrary, $\deg(Q \wedge R)$ is not hyperseparable. $\square$

The previous theorem provides a method for constructing separable and not hyperseparable degrees from hyperinseparable degrees. The following theorem of Binns can be used to construct homogeneous (and thus hyperinseparable and thus separable and not hyperseparable) degrees with various structure.

Lemma 8.2 **([2])** *Let A be a c.e. set and P a Π^0_1 class with* $\deg(P) >_M \mathbf{0}$. *Then there exist c.e. sets A^0, A^1 such that*

$$A^0 \cap A^1 = \varnothing, \tag{11}$$

$$A^0 \cup A^1 = A, \tag{12}$$

$$\forall i \in \{0, 1\} \forall f \in P[A^i \not\geq_T f]. \tag{13}$$

The idea is to construct a pair of hyperinseparable (actually homogeneous) degrees whose meet, by Theorem 8.1, is separable and not hyperseparable, but whose join is as high as we want it. We are also able to avoid a cone.

Corollary 8.3 *For any* $\mathbf{b}, \mathbf{c} >_M \mathbf{0}$ *with* $\mathbf{b}$ *homogeneous, there exist* $\mathbf{b}^0$*,* $\mathbf{b}^1$*, and* $\mathbf{a}$ *such that*

$$\mathbf{a} = \mathbf{b}^0 \wedge \mathbf{b}^1, \tag{14}$$

$$\mathbf{a} \textit{ is separable and not hyperseparable}, \tag{15}$$

$$\mathbf{b}^0, \mathbf{b}^1, \mathbf{a} \not\geq \mathbf{c}, \tag{16}$$

$$\mathbf{b}^0 \vee \mathbf{b}^1 \geq \mathbf{b}. \tag{17}$$

Proof Define $\mathcal{S}(A, B) = \{C : A \subseteq C \subseteq B^c\}$. It is known, see [3], that a class P is homogeneous if and only if $P = \mathcal{S}(A, B)$ for c.e. sets A and B. Fix $Q \in \mathbf{c}$. Fix $R = \mathcal{S}(A, B) \in \mathbf{b}$ with A and B c.e. Let $P = Q \wedge R$ and A^0 and A^1 be as in Lemma 8.2. Let $S^0 = \mathcal{S}(A^0, B)$ and $S^1 = \mathcal{S}(A^1, B)$. For $X \oplus Y \in S^0 \vee S^1$ define

$$Z(n) = \begin{cases} 0 & \text{if } X(n) = 1 \text{ or } Y(n) = 1, \\ 1 & \text{else.} \end{cases} \tag{18}$$

Then $Z \in S$, thus $S^0 \vee S^1 \geq_M R$. If $S^1 \geq_M S^0$, then $S^1 \geq_M R \geq_M P$, a contradiction of (13). The case of $S^0 \geq_M S^1$ is symmetric. Thus $S^0 \perp_M S^1$. Let $T = S^0 \wedge S^1$. If $S^0 \geq_M Q$, then $S^0 \geq_M P$, a contradiction. Similarly for S^1 and T. Note that homogeneity implies hyperinseparability. By Theorem 8.1, T is separable and not hyperseparable. Letting $\mathbf{b}^0 = \deg(S^0)$, $\mathbf{b}^1 = \deg(S^1)$, and $\mathbf{a} = \deg(T)$, we arrive at the result. □

Corollary 8.4 *There exists a degree* $\mathbf{a}$ *such that* $\mathbf{a}$ *is separable and not hyperseparable.*

9 Hyperseparable and Not Totally Separable

Finally, we work to separate the notions of hyperseparable and not totally separable. The task is twofold: we must build a hyperseparable degree and avoid being totally separable.

The first task is complicated by the fact that, previously, the only known construction of a hyperseparable degree was to build a totally separable degree. We will use Theorem 7.3 to show that it is sufficient to build a class which is hyperseparable and not totally separable. We then use the methods of Section 4 to build such a class.

Theorem 9.1 *If* P *is hyperseparable and not totally separable, then* $\deg(P)$ *is hyperseparable and not totally separable.*

Proof That $\deg(P)$ is hyperseparable is immediate. Assume $\deg(P)$ is totally separable. Then, by Theorem 7.3, there exists $R \subseteq P$, $R \equiv_M P$, and R totally separable. As P is not totally separable, $R \neq P$. Let C be a clopen set such that $R \subseteq P \cap C \subset P$. Such a C exists as for $X \in P \setminus R$ there is some $\sigma \preceq X$ with $\sigma \notin \mathcal{T}_R$ and $C = I(\sigma)^c$ suffices. Using Lemma 2.3 twice, $P \cap C \leq_M R \equiv_M P \leq_M P \cap C^c$, contradicting that P is hyperseparable. Thus, $\deg(P)$ is not totally separable. □

Theorem 9.2 *There exists a degree which is hyperseparable and not totally separable.*

Proof We will build a computable tree life $\{L_s\}$. By Corollary 4.8, $P = [\lim_s L_s]$ will be a Π^0_1 class. We will build P to be hyperseparable and not totally separable. By Theorem 9.1, $\deg(P)$ will be hyperseparable and not totally separable.

Let $\langle C, \varphi\rangle$ be an enumeration of all pairs of clopen subclasses of 2^ω and partial computable functions. For convenience we often refer to such pairs by their index, e, in the enumeration. We also start the enumeration at $e = 1$. We will blur the distinction between e and $\langle C, \varphi\rangle$. For each $\langle C, \varphi\rangle$ we work to satisfy the requirement,

$$\mathcal{R}_{C,\varphi} : C \text{ good for } P \Rightarrow \exists X \in P \cap C[\Phi(X) \notin P \cap C^c]. \tag{19}$$

To ensure that P is not totally separable we will use a very simple reduction and ensure that paths Turing equivalent through that reduction exist. Namely,

$$\mathcal{S} : \exists X, Y \in P \exists Z \in 2^\omega[X = 0^\frown Z \text{ and } Y = 1^\frown Z]. \tag{20}$$

We have a strategy acting on behalf of each $\mathcal{R}_e$ which will be careful to ensure that $\mathcal{S}$ is satisfied. Strategies are ordered in priority in the order of the enumeration with earlier strategies having higher priority. Each node has a protection level. The function $r_s : 2^{<\omega} \to \omega \cup \{\omega\}$ indicates the protection level; that is, the protection level of σ at stage s is $r_s(\sigma)$. Lower numbers indicate higher protection levels. A strategy may protect a node with its own priority. Each strategy has two states: *wait* and *stop*. Strategies begin in state *wait* and may at some point act and enter state *stop*. Once in state *stop*, a strategy will not act unless injured. When a strategy acts, it injures all lower priority strategies resetting them to state *wait*. The construction thus progresses in typical finite injury fashion. Denote the state of strategy e at stage s by $\text{state}_s(e)$.

In order for all strategies to be able to find witnesses to kill, they must obey a simple rule regarding protection levels. For a node σ protected at level d, strategy e ($e > d$) may only kill $\tau \succeq \sigma$ if $|\tau| \geq |\sigma| + 2(e - d)$. As we shall see in the claims below, this will ensure that every strategy is able to kill a witness if needed.

Let

$$S_s = \{\sigma : 0^\frown\sigma \in D(L_s) \text{ and } 1^\frown\sigma \in D(L_s)\}. \tag{21}$$

To ensure $\mathcal{S}$ is satisfied we require all strategies to preserve $S_s \neq \varnothing$ and only trim the tree at even stages, growing it with single extensions at odd stages. This will ensure that S_s is never empty and every requirement can act if necessary.

Begin with $L_0 = \{\varnothing, 0, 1\}$, $r_0(\sigma) = \infty$ for all σ, and $\text{state}_0(e) = \textit{wait}$ for all e. Assume we have run the construction up to stage s. Thus L_{s-1}, r_{s-1}, and state_{s-1} are all defined. A strategy $e = \langle C, \varphi\rangle$ is eligible to act if $\text{state}_{s-1}(e) = \textit{wait}$ and

$$\begin{aligned}\exists \sigma \in L_{s-1} \cap C[\varphi(\sigma) \in L_{s-1} \cap C^c \\ \text{and } \forall \tau \preceq \varphi(\sigma)[r_{s-1}(\tau) > e \\ \text{or } |\varphi(\sigma)| \geq |\tau| + 2(e - r_{s-1}(\tau))] \\ \text{and } \exists \nu \in S_{s-1}[0^\frown\nu \not\preceq \varphi(\sigma) \text{ and } 1^\frown\nu \not\preceq \varphi(\sigma)]].\end{aligned} \tag{22}$$

If s is odd or if no such e exists, then let $L_s = \text{extend}(L_{s-1})$, $r_s = r_{s-1}$, and $\text{state}_s = \text{state}_{s-1}$. Otherwise, let e be the highest priority (least index) strategy eligible to act. Let $\sigma' \in D(L_s)$ be a descendant of σ (possibly equal to σ). Let

$$L_s = \text{extend}(\text{trim}(L_{s-1}, \varphi(\sigma))), \tag{23}$$

$$r_s(\tau) = \begin{cases} e & \tau = \sigma', \\ r_{s-1}(\tau) & r_{s-1}(\tau) < e, \\ \omega & \text{else}, \end{cases} \tag{24}$$

$$\text{state}_s(n) = \begin{cases} \mathit{stop} & n = e, \\ \mathit{wait} & n > e, \\ \text{state}_{s-1}(n) & \text{else}. \end{cases} \tag{25}$$

Equation (23) describes the evolution of the tree life. Equations (24) and (25) serve to protect σ, stop strategy e, and injure (reset) all lower priority strategies. Observe that $S_s \neq \varnothing$ as $0 ^\frown \nu$ and $1 ^\frown \nu$ were not killed for ν as in (22).

This completes the construction. We now prove that the result has the desired properties.

Claim 9.3 *Fix any d and e with $d < e$. If strategy d is not injured after stage t, then strategy e will be injured less than or equal to 2^{e-d-1} times after stage t.*

Proof Fix d and t and let $I(e)$ denote the maximum number of times e could be injured after stage t. We will show by induction that $I(e) \leq 2^{e-d-1}$.

Consider $e = d + 1$. Then e will be injured only if d acts after stage t. As d is not injured after stage t it will act at most once and thus $I(e) = 1 \leq 2^{e-d-1} = 2^{d+1-d-1} = 2^0 = 1$.

Assume $I(e') \leq 2^{e-d-1}$ for all $d < e' < e$. Any time a strategy below $e - 1$ is injured, $e - 1$ is also injured. Thus $I(e - 1)$ is an accurate count of the number of times e might be injured by strategies $< e - 1$. Each time $e - 1$ is injured, e is injured. In addition, $e - 1$ may act once before being injured again, injuring e as well. Thus $I(e) \leq 2I(e - 1) = 2(2^{e-1-d-1}) = 2^{e-1-d-1+1} = 2^{e-d-1}$. □

Claim 9.4 *Fix any d and e with $d < e$. If strategy d is not injured after stage t, then strategy e will act fewer than 2^{e-d} times after stage t.*

Proof Strategy e can only act once before being injured again. Thus the total number of times it can act is equal to the number of times it is injured plus one. By the previous claim this is less than or equal to $2^{e-d-1} + 1$ which is less than 2^{e-d}. □

Claim 9.5 *For all σ and s such that $r_s(\sigma) = e < \omega$ and strategy e is not injured at or after stage s, $\sigma \in \lim_s L_s$.*

Proof As strategy e is not injured, no strategy of higher priority will kill any ancestor of σ, so our only worry is that lower priority strategies will kill all the children of σ. When σ was protected, it was a leaf node. Thus any strategy which kills ancestors of σ must obey the protection. Namely, for $d > e$, d can only kill $\tau \succeq \sigma$ if $|\tau| \geq |\sigma| + 2(d - e)$.

Let μ be the standard measure on 2^ω; that is, $\mu(I(\tau)) = 2^{-|\tau|}$. For a finite L_s, define $\mu(I(\sigma) \cap L_s)$ to be $\mu(I(\sigma) \cap \bigcup_{\tau \in D(L_s)} I(\tau))$; that is, we assume that L_s will have all possible children. We will show that, for all $t > s$, $\mu(I(\sigma) \cap L_t) > 0$ and, thus, $\sigma \in L_t$.

Fix a stage t and let d be the lowest priority (highest index) strategy to act so far. For each $e < f \leq d$ let N_f be the number of children of σ strategy f has killed and $\{\tau_{i,f}\}$ be the set of these children. Strategy f can kill only a single child when it acts, so by the previous claim, $N_f < 2^{f-e}$. The requirement on the length of τ requires that $I(\tau_{i,f}) \leq 2^{-(|\sigma|+2(d-e))}$.

$$\mu(I(\sigma) \cap L_t) = \mu(I(\sigma)) - \sum_{e<f\leq d} \sum_{i<N_f} \mu(I(\tau_{i,f})) \tag{26}$$

$$\geq 2^{-|\sigma|} - \sum_{e<f\leq d} N_f 2^{-(|\sigma|+2(f-e))} \tag{27}$$

$$> 2^{-|\sigma|} - \sum_{e<f\leq d} 2^{f-e} 2^{-(|\sigma|+2(f-e))} \tag{28}$$

$$= 2^{-|\sigma|} - \sum_{i=1}^{d-e} 2^i 2^{-(|\sigma|+2i)} \tag{29}$$

$$= 2^{-|\sigma|} - \sum_{i=1}^{d-e} 2^{-|\sigma|-i} \tag{30}$$

$$= 2^{-|\sigma|}\left(1 - \sum_{i=1}^{d-e} 2^{-i}\right) \tag{31}$$

$$> 0. \tag{32}$$

□

Claim 9.6 *$\{L_s\}$ is a computable tree life.*

Proof The previous claim shows that, at all stages, L_s is nonempty. By Lemmas 4.11 and 4.12, each L_s is a growth of L_{s-1}. As single extensions and trims are computable, it is a computable tree life. □

Thus, by Corollary 4.8, $P = [\lim_s L_s]$ is a nonempty Π^0_1 class.

Claim 9.7 $\forall n \exists s \forall t > s\big[|S_t| > n\big]$.

Proof Define a *clump* in S_t to be a proper subset $U \subset S_t$ maximal with respect to $U = \{\sigma ^\frown \tau : \tau \in 2^i\}$ for some σ and i. Let c_t be the size of the smallest clump in S_t.

First we show that $c_{t+1} \geq c_t$ for all t. At each stage something may be killed and then every living leaf is extended; that is, each stage is a composition of (possibly) a trim and a single extension. As any living leaf is at least in a clump of itself ($i = 0$), $c_t > 0$. There are four possibilities:

1. Nothing in S_t is killed. Then the clump doubles and $c_{t+1} = 2c_t$.
2. The smallest clump is killed. As it was not everything there is another clump of at least equal size. That clump will double, but it could now be everything in which case it is not a clump but rather two clumps of size equal to the original. So $c_{t+1} \geq c_t$.
3. Everything but the smallest clump is killed. Then the smallest clump will double but as it is now everything it is now two clumps rather than one. So $c_{t+1} = c_t$.

4. Part of a clump is killed. At worst it will kill half the smallest clump. The other half will then double and $c_{t+1} = c_t$.

At odd stages, case (1) occurs, so $c_{t+1} > c_t$ for t odd. Thus c_t is unbounded in t and S_t is unbounded in t. □

Observe that while clumps of arbitrary finite size exist during the construction they may move around. The final Π^0_1 class may be very nonclumpy.

Claim 9.8 *For all e, $\mathcal{R}_e$ is satisfied.*

Proof By a previous claim, let s be sufficiently large such that strategy e is not injured at or after stage s. Let $e = \langle C, \varphi \rangle$. If C is not good for P, then we are done. Assume C is good for P.

Assume there exists $X \in P \cap C$ such that $\Phi(X) \in P \cap C^c$. For stages $t > s$, the set of nodes protected by strategies $d < e$ will stay fixed. Thus there is a stage t and an n such that $\sigma = X \restriction n \in L_t$, $|\varphi(\sigma)| \geq |\tau| + 2(e - r_{t-1}(\tau))$ for all $\tau \preceq \varphi(\sigma)$ with $r_{t-1}(\tau) < e$. Strategy e may still not be able to act because of the requirement to preserve S_t. As higher priority strategies will not act again the strategy will continue to be otherwise eligible to act at later stages. Let Y be such that $X = i^\frown Y$ for some $i \in 2$. If we never act that means that at each stage t, $S_t = \{Y \restriction l(L_t)\}$, contradicting the previous claim that $|S_t|$ is unbounded. □

Claim 9.9 *$\mathcal{S}$ is satisfied.*

Proof By the definition of S_s and that L_s is a tree life, if a string σ leaves S_s, that is, $\sigma \in S_s \setminus S_{s+1}$, then no descendant of it can ever enter S_s later. Thus, using an above claim, $[\lim_s S_s]$ exists and is nonempty. Then, for any Z in $[\lim_s S_s]$, $X = 0^\frown Z$ and $Y = 1^\frown Z$ serve as witnesses that $\mathcal{S}$ is satisfied. □

Thus P is hyperseparable as for any C good for P and any Φ, $\mathcal{R}_{\langle C,\varphi\rangle}$ shows that Φ is not a witness of $P \cap C \geq_M P \cap C^c$. As $\mathcal{S}$ is satisfied, P contains a pair of comparable paths, namely, $0^\frown X$ and $1^\frown X$ for some X. □

References

[1] Alfeld, C. P., "Non-branching degrees in the Medvedev lattice of Π^0_1 classes," *The Journal of Symbolic Logic*, vol. 72 (2007), pp. 81–97. Zbl 1122.03043. MR 2298472. 228, 229, 230, 231

[2] Binns, S., "A splitting theorem for the Medvedev and Muchnik lattices," *Mathematical Logic Quarterly*, vol. 49 (2003), pp. 327–35. Zbl 1022.03021. MR 1987431. 228, 237

[3] Cenzer, D., and P. G. Hinman, "Density of the Medvedev lattice of Π^0_1 classes," *Archive for Mathematical Logic*, vol. 42 (2003), pp. 583–600. Zbl 1037.03040. MR 2001061. 228, 230, 231, 238

[4] Cenzer, D., and C. G. Jockusch, Jr., "Π^0_1 classes—Structure and applications," pp. 39–59 in *Computability Theory and Its Applications (Boulder, CO, 1999)*, vol. 257 of *Contemporary Mathematics*, American Mathematics Society, Providence, 2000. Zbl 0962.03040. MR 1770733. 227

[5] Cenzer, D., and J. B. Remmel, "Π^0_1 classes in mathematics," pp. 623–821 in *Handbook of Recursive Mathematics, Vol. 2*, vol. 139 of *Studies in Logic and the Foundations of Mathematics*, North-Holland, Amsterdam, 1998. Zbl 0941.03044. MR 1673586. 227

[6] Jockusch, C. G., Jr., and R. I. Soare, "Π^0_1 classes and degrees of theories," *Transactions of the American Mathematical Society*, vol. 173 (1972), pp. 33–56. Zbl 0262.02041. MR 0316227. 228, 233

[7] Rogers, H., Jr., *Theory of recursive functions and effective computability*, The MIT Press, Cambridge, 1987. MR 886890. 229

[8] Simpson, S. G., "Mass problems and randomness," *The Bulletin of Symbolic Logic*, vol. 11 (2005), pp. 1–27. Zbl 1090.03015. MR 2125147. 228

[9] Simpson, S. G., "Π^0_1 sets and models of WKL_0," pp. 352–78 in *Reverse Mathematics 2001*, edited by S. G. Simpson, vol. 21 of *Lecture Notes in Logic*, A. K. Peters, Ltd., Wellesley, 2005. Zbl 1106.03051. MR 2186912. 228

[10] Simpson, S. G., "An extension of the recursively enumerable Turing degrees," *Journal of London Mathematical Society, Second Series*, vol. 75 (2007), pp. 287–97. Zbl 1119.03037. MR 2340228. 228

[11] Soare, R. I., *Recursively Enumerable Sets and Degrees*, Perspectives in Mathematical Logic, Springer-Verlag, Berlin, 1987. Zbl 0667.03030. MR 882921. 229

Acknowledgments

I thank my advisor, Steffen Lempp, for his guidance and conversations. I also thank the anonymous referees for their detailed feedback.

Department of Mathematics
University of Wisconsin
Madison WI 53706
calfeld@mac.com