ON THE DEGREE SPECTRUM OF A Π^0_1 CLASS

THOMAS KENT
ANDREW E.M. LEWIS

Abstract. For any $\mathcal{P} \subseteq 2^\omega$, define $S(\mathcal{P})$, the *degree spectrum* of $\mathcal{P}$, to be the set of all Turing degrees $\mathbf{a}$ such that there exists $A \in \mathcal{P}$ of degree $\mathbf{a}$. We prove a number of basic properties of the structure which is the degree spectra of Π^0_1 classes ordered by inclusion and also study in detail some other phenomena relating to the study of Π^0_1 classes from a degree theoretic point of view, which are brought to light as a result of this analysis.

1. Introduction

The study of the degree theoretic complexity of members of Π^0_1 classes, dating back as far as Kleene [SK], has resulted in a rich and well developed theory. Two fundamental papers in this area are [JS1] and [JS2] by Jockusch and Soare in which they proved, amongst other things, the much cited low basis theorem as well as a number of basic results regarding *special* Π^0_1 classes i.e. non-empty Π^0_1 classes which have no computable members. For a detailed account of work in the area we refer the reader to [DC]. In this paper we shall be concerned with a natural structure regarding the study of Π^0_1 classes from a degree theoretic point of view which, while many known theorems can be seen to relate, has not been studied explicitly in the previous literature.

Unless stated otherwise, by a Π^0_1 class we shall mean a Π^0_1 class of sets of natural numbers. We let $\mathfrak{D}$ denote the set of all Turing degrees. For any $\mathcal{P} \subseteq 2^\omega$ we define $S(\mathcal{P})$, the *degree spectrum* of $\mathcal{P}$, to be the set of all degrees $\mathbf{a}$ such that there exists $A \in \mathcal{P}$ of degree $\mathbf{a}$. We define $\mathfrak{P} = \{S(\mathcal{P}) : \mathcal{P} \text{ is a } \Pi^0_1 \text{ class}\}$ and we consider the elements of $\mathfrak{P}$ to be ordered by inclusion. While this structure has not been studied directly before, it does relate to recent work by Simpson [SS] [SS2], Binns, Cole [SC] and others, who have been interested in the Muchnik degrees of Π^0_1 classes. As pointed out by Simpson in a private correspondence, it is not difficult to see that the latter structure is dually isomorphic to the substructure of $(\mathfrak{P}, <)$ given by the non-empty degree spectra of Π^0_1 classes which are upward closed.

Let us begin, then, by making some basic observations about the structure $(\mathfrak{P}, <)$. The following facts are easily derived.

(i) $(\mathfrak{P}, <)$ has a greatest element $\mathbf{1}_{\mathfrak{P}} = \mathfrak{D}$ and a least element $\mathbf{0}_{\mathfrak{P}} = \emptyset$.
(ii) $(\mathfrak{P}, <)$ is an uppersemilattice.

2000MSC: 03D28. The first author was supported by Marie-Curie Fellowship MIFI-CT-2006-021702. The second author was supported by a Royal Society University Research Fellowship.

(iii) We shall say that $\boldsymbol{\alpha} > \mathbf{0}_{\mathfrak{P}}$ in $\mathfrak{P}$ is *minimal* if there does not exist $\boldsymbol{\beta} \in \mathfrak{P}$ with $\mathbf{0}_{\mathfrak{P}} < \boldsymbol{\beta} < \boldsymbol{\alpha}$. $(\mathfrak{P}, <)$ has at least one minimal element $\boldsymbol{\alpha} = \{\mathbf{0}\}$.

In order to prove (ii) it suffices to observe that if $\mathcal{P}_0$ and $\mathcal{P}_1$ are Π^0_1 classes then $\mathcal{P}_0 \cup \mathcal{P}_1$ is a Π^0_1 class with degree spectrum $S(\mathcal{P}_0) \cup S(\mathcal{P}_1)$.

Some well known examples of degree spectra will prove useful tools in analyzing the structure. Recall that a degree is PA if it contains a set which codes a complete and consistent extension of Peano Arithmetic according to some computable bijection between sentences in the first order language of arithmetic and the natural numbers, and that a degree is (Martin-Löf) random if it contains a random set—for a detailed account of the theory of algorithmic randomness we refer the reader to [DH] and [AN].

Definition 1.1. *We let* $\boldsymbol{p}$ *denote the set of all PA degrees and we let* $\boldsymbol{r}$ *denote the set of all Martin-Löf random degrees.*

It is well known that $\mathbf{p}$ and $\mathbf{r}$ are both degree spectra of Π^0_1 classes. For degree spectra $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, we shall write $\boldsymbol{\alpha} \cap \boldsymbol{\beta}$ in order to denote their intersection and, when $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are both elements of $\mathfrak{P}$, we shall write $\boldsymbol{\alpha} \wedge \boldsymbol{\beta}$ in order to denote their greatest lower bound in $\mathfrak{P}$ when this exists. Stephan has shown [FS] that the degrees which are random and PA are precisely the degrees $\geq \mathbf{0}'$. Since every nonempty Π^0_1 class contains a member of low degree [JS1] this suffices to show that there exist $\boldsymbol{\alpha}, \boldsymbol{\beta} \in \mathfrak{P}$ such that $\boldsymbol{\alpha} \cap \boldsymbol{\beta} \notin \mathfrak{P}$. Note, however, that $\mathbf{r} \wedge \mathbf{p}$ is defined and is equal to $\mathbf{0}_{\mathfrak{P}}$, so that we cannot immediately conclude that $(\mathfrak{P}, <)$ is not a lattice. In section 8 we shall show that, in fact, the structure is a lattice, since the intersection of two degree spectra of Π^0_1 classes is the degree spectrum of a Π^0_1 class if it is the superset of the degree spectrum of a non-empty Π^0_1 class.

Since any special Π^0_1 class is uncountable, it immediately follows that any minimal element of $(\mathfrak{P}, <)$ other than $\{\mathbf{0}\}$ has uncountably (and so continuum) many elements. The following theorem suffices to show that there are no maximal elements amongst the degree spectra of special Π^0_1 classes.

Theorem 1.1 (Jockusch, Soare [JS2]). *If* $\mathcal{P}$ *is a special* Π^0_1 *class then there exists a nonzero c.e. degree* $\boldsymbol{a} \notin S(\mathcal{P})$. *On the other hand, for every degree* $\mathbf{a}$ *with* $\mathbf{0} < \mathbf{a} \leq \mathbf{0}'$ *there exists a special* Π^0_1 *class* $\mathcal{P}$ *with* $\boldsymbol{a} \in S(\mathcal{P})$.

In the following sections we shall prove a number of basic structural properties of $(\mathfrak{P}, <)$, and in doing so we shall come across some phenomena relating to the study of Π^0_1 classes which are worthy of examination in their own right. In section 2 we shall describe notation and recall basic definitions required for what is to follow. In section 3 we shall give some easy examples of elements of the structure. In section 4 we shall prove, as well as some corresponding results for the Borel hierarchy, that there exist degrees $\boldsymbol{a}$ such that no Σ^0_3 class contains a member of degree $\boldsymbol{a}$ unless it contains a member of every degree. One immediate consequence of these proofs is a characterization, for sufficiently low levels of the arithmetical and Borel hierarchies, of those sets which contain a member of every degree in an upper cone. Much of this section will be devoted to a study of the *invisible* degrees—degrees, that is, which do not belong to any member of $\mathfrak{P}$ other than $\mathbf{1}_{\mathfrak{P}}$. The existence of invisible degrees suffices to show that there do not exist $\boldsymbol{\alpha},\boldsymbol{\beta} \in \mathfrak{P}$ such that $\boldsymbol{\alpha} < \mathbf{1}_{\mathfrak{P}}$ and $\boldsymbol{\beta} < \mathbf{1}_{\mathfrak{P}}$ with $\boldsymbol{\alpha} \vee \boldsymbol{\beta} = \mathbf{1}_{\mathfrak{P}}$. In section 5 we shall give a characterization of the minimal elements of the structure and use this to show, for example, that both $\boldsymbol{r}$ and $\boldsymbol{p}$ are minimal. We shall also show that for any $\boldsymbol{\alpha} < \mathbf{1}_{\mathfrak{P}}$ there exists $\boldsymbol{\beta}$ which is minimal and such that $\boldsymbol{\beta} \not\leq \boldsymbol{\alpha}$. In section 6 we shall consider pairs of degrees $\boldsymbol{a}$ and $\boldsymbol{b}$ such that, whenever a Π^0_1 class contains a member of degree $\boldsymbol{a}$, it contains a member of degree $\boldsymbol{b}$. We define the Π^0_1-*mates* of $\boldsymbol{a}$ to be the intersection of all $\boldsymbol{\alpha}$ in $\mathfrak{P}$ which contain $\boldsymbol{a}$ and we show that for any $\boldsymbol{\alpha} < \mathbf{1}_{\mathfrak{P}}$ there exists a degree $\boldsymbol{a} \notin \boldsymbol{\alpha}$ such that the Π^0_1-mates of $\boldsymbol{a}$ are precisely $\boldsymbol{\alpha} \cup \{\boldsymbol{a}\}$. This suffices to show that every element of the structure other than $\mathbf{1}_{\mathfrak{P}}$ has a strong minimal cover. We also show that there exists a non-computable set A which is (Cantor-Bendixson) ranked, such that every $B \leq_T A$ is ranked, and such that no non-computable $B \leq_T A$ is a member of any rank faithful class. In section 7 we prove two anti-basis theorems for Π^0_1 classes. We show first that any Π^0_1 class that contains a member of every low degree contains a member of every degree. We then go on to show that if $\boldsymbol{b}$ is non-GL$_2$ then, for any Π^0_1 class $\mathcal{P}$ which does not contain a member of every degree, there exists some non-zero $\boldsymbol{a} \leq \boldsymbol{b}$ such that $\mathcal{P}$ does not contain any member of non-zero degree below $\boldsymbol{a}$.

2. Basic definitions and notation

Some frequently used variables. We shall use the variables $e, i, j, k, l, n, m, s, t$ for elements of ω; σ, τ for elements of $2^{<\omega}$; ϕ for elements of $\omega^{<\omega}$; f, g for elements of ω^ω; h for (possibly) partial functions $\omega \to \omega$; A, B, C for subsets of ω (which we identify in the standard way with infinite binary strings); $\mathbf{a}, \mathbf{b}$ for elements of $\mathfrak{D}$. We shall use the variables $\mathcal{P}$ and $\mathcal{Q}$ for subsets of 2^ω and we shall use the variable $\mathcal{R}$ for subsets of ω^ω. We shall use the variables $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ to range over the elements of $\mathfrak{P}$. We write $\phi \subseteq h$ in order to denote that ϕ is an initial segment of h. We write $h_0 \oplus h_1$ in order to denote that (possibly partial) function h_2 such that for all n, $h_2(2n) \downarrow = m$ iff $h_0(n) \downarrow = m$ and $h_2(2n+1) \downarrow = m$ iff $h_1(n) \downarrow = m$.

Turing functionals. We let $\{\Psi_i\}_{i\in\omega}$ be an effective listing of the Turing functionals, and we shall use the variables Ψ and Φ to range over the Turing functionals. We say that a Turing functional Φ is total if $\Phi(A)$ is total for all A. **The following convention is worth singling out for attention**: we assume that, for any σ and any n, $\Psi(\sigma; n)$ is not defined unless σ is of length at least n, this computation converges in at most $|\sigma|$ steps, and $\Psi(\sigma; n')$ is defined for all $n' < n$. Thus at any stage of a computable construction we can decide any value $\Psi(\sigma; n)$.

Strings and sets of strings. For any $\phi \in \omega^{<\omega}$ and any $\phi' \in \omega^{<\omega}$ ($f \in \omega^\omega$) we let $\phi \star \phi'$ ($\phi \star f$) denote the concatenation of ϕ and ϕ' (ϕ and f). For $i \in \omega$ we may sometimes write i in order to denote the string ϕ of length 1 such that $\phi(0) = i$. We may also write 01, for example, in order to denote the string τ of length 2 such that $\tau(0) = 0$ and $\tau(1) = 1$. We shall use the variables Λ, Υ, T and U to range over subsets of $2^{<\omega}$. We say that Λ is downward closed if, whenever $\tau \in \Lambda$, all initial segments of τ are also in this set. Generally we shall use the variables Λ, Υ to range over downward closed subsets of $2^{<\omega}$ and we shall use the variables T and U to range over subsets of $2^{<\omega}$ which may not be downward closed. We say that τ and τ' are Ψ-splitting if $\Psi(\tau)$ and $\Psi(\tau')$ are incompatible. We say T is Ψ-splitting if every pair of incompatible strings in T is Ψ-splitting. We shall use the variable Π to range over subsets of $\omega^{<\omega}$. We define $[\Pi] = \{f : \exists^{\infty}\phi(\phi \subset f \wedge \phi \in \Pi)\}$. A leaf of Π is a string $\phi \in \Pi$ such that no proper extension of this string is in Π. If $\phi \in \Pi$ then the level of ϕ in Π is defined to be the number of proper initial segments of ϕ in Π. If $\phi, \phi' \in \Pi$, $\phi \subset \phi'$ and there does not exist ϕ'' with $\phi \subset \phi'' \subset \phi'$ then we say that ϕ' is a successor of ϕ in Π and that ϕ is the immediate predecessor of ϕ' in Π. We say that non-empty Π is perfect if every element has two incompatible extensions in the set, and that Π is 2-branching if it has a single element of level 0 and every element has precisely two successors. Π is finitely 2-branching if Π is finite, it has a single element of level 0, all leaves are of the same level, and every element which is not a leaf has precisely two successors. We say that Π is of level n if Π is finite and all leaves of Π are of level n in Π. Infinite Π is pointed if every element of $[\Pi]$ computes Π. We let λ denote the string of length 0.

Listing the Π^0_1 classes. We let $\{\Lambda_i\}_{i\in\omega}$ be a uniformly computable sequence of downward closed subsets of $2^{<\omega}$ such that, for any Π^0_1 class $\mathcal{P}$, there exists an infinite number of i with $\mathcal{P} = [\Lambda_i]$.

Cantor-Bendixson rank. We consider the Cantor topology on 2^{ω}. A subset of 2^{ω} is F_{σ} if it is a countable union of closed sets, and is G_{δ} if it is a countable intersection of open sets. The Cantor-Bendixson derivative of $\mathcal{P}$ is the set of non-isolated points of $\mathcal{P}$ according to the Cantor topology and is denoted $D(\mathcal{P})$. The iterated derivative $D^{\alpha}(\mathcal{P})$ is defined for all ordinals α by transfinite recursion:

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

A set A has rank α relative to $\mathcal{P}$ if α is the least ordinal such that $A \in D^{\alpha}(\mathcal{P})$ and $A \notin D^{\alpha+1}(\mathcal{P})$. The (Cantor-Bendixson) rank of any set (should it have a rank) is its least rank relative to any Π^0_1 class.

Some standard theorems and facts. We shall use the standard Lebesgue measure on Cantor space. It's a theorem of Kucera [AK] that every Π^0_1 class of positive measure contains a member of every random degree and it is also easily observed that any Π^0_1 class containing a random set is of positive measure. The low basis theorem [JS1] states that every non-empty Π^0_1 class contains a member of low degree. Recall that a degree $\boldsymbol{a}$ is hyperimmune-free iff every f of degree below $\boldsymbol{a}$ is majorized by a computable function i.e. there exists a computable function g such that $g(n) \geq f(n)$ for all n. The hyperimmune-free basis theorem [JS1] states that every non-empty Π^0_1 class contains a member of hyperimmune-free degree. If A is of hyperimmune-free degree and $B \leq_T A$ then there exists a total Turing functional Φ such that $\Phi(A) = B$ i.e. $B \leq_{tt} A$.

3. Examples

In order to get a better picture of what $(\mathfrak{P}, <)$ looks like, it seems a good idea to describe some easy examples of elements of the structure. Since every non-zero degree below $\mathbf{0}'$ is hyperimmune and the hyperimmune degrees are upward closed, it follows immediately from the low basis theorem and the hyperimmune-free basis theorem that no non-trivial upper cone can be the degree spectrum of a Π^0_1 class. The situation becomes more interesting, however, if we consider the union of upper cones with $\{\mathbf{0}\}$. For every computable ordinal α and every degree $\boldsymbol{a}$ with $\mathbf{0}^{(\alpha)} \leq \boldsymbol{a} \leq \mathbf{0}^{(\alpha+1)}$, the sets $\{\mathbf{0}\} \cup \{\boldsymbol{b} : \boldsymbol{b} \geq \boldsymbol{a}\}$ and $\{\mathbf{0}, \boldsymbol{a}\}$ are each the degree spectrum of a Π^0_1 class. This follows (as pointed out by Jockusch in a private correspondence) since Jockusch and McLaughlin [JM] have shown that any such degree contains some function which is a Π^0_1 singleton i.e. which is the only element of some Π^0_1 class of functions. From this it is easily seen that the cone above $\boldsymbol{a}$ is the degree spectrum of a Π^0_1 class of functions. Then it follows from the proof of Theorem 1 of [JS2] that if $\mathcal{R}$ is a Π^0_1 class in ω^ω, there is a Π^0_1 class $\mathcal{P} \subseteq 2^\omega$ such that $S(\mathcal{P}) = S(\mathcal{R}) \cup \{\mathbf{0}\}$. By taking a suitable increasing sequence of degrees, it follows that $(\mathfrak{P}, <)$ is not wellfounded. Combined with the hyperimmune-free basis theorem it also follows that there exists an uncountable $\boldsymbol{\alpha}$ such that all $\boldsymbol{\beta} \leq \boldsymbol{\alpha}$ contain $\mathbf{0}$. For every $\boldsymbol{a}$ which contains a set of rank 1, $\{\mathbf{0}, \boldsymbol{a}\}$ is the degree spectrum of a Π^0_1 class. Upon realizing this, it is natural to ask whether for every such degree $\boldsymbol{a}$, $\{\mathbf{0}\} \cup \{\mathbf{b} : \mathbf{b} \geq \mathbf{a}\}$ is the degree spectrum of a Π^0_1 class. Later, in section 4, we shall be able to give a negative solution.

Recall that a degree is fixed point free iff it contains a DNC (diagonally non-computable) function, where a function f is defined to be DNC if $f(n) \neq \Psi_n(n)$ for all n.

Definition 3.1. *We let* $\mathbf{f}$ *denote the set of all fixed point free degrees.*

Simpson [SS] [SS2] has shown that $\mathbf{f}$ is an element of $\mathfrak{P}$. Another very natural example of an element of the structure is the set of c.e. degrees.

Theorem 3.1. *The set of all c.e. degrees is the degree spectrum of a* Π^0_1 *class.*

Proof. We let W_i denote the i^{th} c.e. set according to some fixed effective listing. It is not difficult to see that there exists a computable function f such that, for any $i \in \omega$, $S([\Lambda_{f(i)}]) = \{\mathbf{0}, \boldsymbol{a}_i\}$ where $\boldsymbol{a}_i$ is the degree of W_i. Now let Λ be the computable set of strings which contains a copy of $\Lambda_{f(i)}$ above each string $0^i \star 1$, where 0^i is the sequence of i many zeros. □

Corollary 3.1. *There exists* $\boldsymbol{\alpha} < \mathbf{1}_{\mathfrak{P}}$ *such that, for every* $\boldsymbol{\beta} > \mathbf{0}_{\mathfrak{P}}$, $\boldsymbol{\alpha} \cap \boldsymbol{\beta} \neq \emptyset$.

Proof. Every element of $\mathfrak{P}$ other than $\mathbf{0}_{\mathfrak{P}}$ contains a c.e. degree (the degree of the leftmost element of any Π^0_1 class is the same as the degree of the set of finite binary strings strictly to its left, which is clearly a computably enumerable set). □

Note, however, that for any countable $\boldsymbol{\alpha} \in \mathfrak{P}$ and any $\boldsymbol{\beta}$ which is the degree spectrum of a special Π^0_1 class, $\boldsymbol{\alpha} \wedge \boldsymbol{\beta} = \mathbf{0}_{\mathfrak{P}}$ (since any countable Π^0_1 class has a computable element), so that the only $\boldsymbol{\alpha} < \mathbf{1}_{\mathfrak{P}}$ for which it is not obviously the case that there exists $\boldsymbol{\beta} > \mathbf{0}_{\mathfrak{P}}$ with $\boldsymbol{\alpha} \wedge \boldsymbol{\beta} = \mathbf{0}_{\mathfrak{P}}$ are those uncountable $\boldsymbol{\alpha}$ which contain $\mathbf{0}$. The following theorem suffices to show that the equivalent of Corollary 3.1 does not hold for the degree spectra of special Π^0_1 classes. Although the result has already been proved by Cole and Simpson [CS], we include a proof here which serves as a good introduction to techniques which will be used later in the paper.

Theorem 3.2 (Cole and Simpson [CS])**.** *For any special Π^0_1 class $\mathcal{P}_0$ there exists a special Π^0_1 class $\mathcal{P}_1$ such that no member of $\mathcal{P}_1$ computes any member of $\mathcal{P}_0$.*

Proof. Suppose given downward closed and computable Λ such that $\mathcal{P}_0 = [\Lambda]$. We define an approximation to a 2-branching T such that $\mathcal{P}_1 = [T]$ satisfies the statement of the theorem. Those τ in T of level $2i+1$ will be defined so as to satisfy requirement:

Θ_i: If $A \in \mathcal{P}_1$ then $A \neq \Psi_i(\emptyset)$.

Those τ in T of level $2i+2$ will be defined so as to satisfy requirement:

Ξ_i: If $A \in \mathcal{P}_1$ and $\Psi_i(A)$ is total then $\Psi_i(A) \notin \mathcal{P}_0$.

Stage 0. Enumerate λ into T.

Stage $s > 0$. Consider all the strings in T to be ordered first according to their level in T and then lexicographically. Find the least string $\tau \in T$ (if any) such that either:

(1) τ is of level $2i+1$ and $\tau \subset \Psi_i(\emptyset)[s]$. In this case let τ_0 be the immediate predecessor of τ in T and let τ_1 be a leaf of T extending τ_0 and incompatible with τ. Remove all strings properly extending τ_0 from T and then enumerate in two incompatible extensions of τ_1.

(2) τ is of level $2i+2$, $\Psi_i(\tau)$ is compatible with some string in Λ of length s and there exists a leaf τ' of T extending τ such that $\Psi_i(\tau')$ properly extends $\Psi_i(\tau)$. In this case remove all strings extending (and including) τ from T, other than τ'.

Once these instructions are completed, choose two incompatible strings extending each leaf of T, and enumerate these strings into T.

It is clear that $\mathcal{P}_1$ is a Π^0_1 class. In order to see that our approximation to T converges, suppose that for some least i there exists a sequence $\{\tau_j\}_{j\geq 0}$ of strings such that each τ_j is a string of level $2i+2$ in T at some stage of the construction and $\tau_j \subset \tau_{j+1}$ for all j. Let $A = \bigcup_j \tau_j$. Then $\Psi_i(A)$ is computable and is in $\mathcal{P}_0$, which gives the required contradiction. □

4. Invisible degrees

Before going on to consider the issue of which $\boldsymbol{\alpha} \in \mathfrak{P}$ can be cupped to $\mathbf{1}_{\mathfrak{P}}$, we present in this section a number of theorems concerning lower levels of the arithmetical and Borel hierarchies, which will give us valuable information regarding the structure $(\mathfrak{P}, <)$ when applied to the countable sequence of Π^0_1 classes. We prove Theorems 4.1 and 4.2 in a way that may initially seem more complicated than is necessary, in order that we can later use these proofs to deduce certain corollaries.

Theorem 4.1. *For any countable sequence of F_σ sets, $\{\mathcal{Q}_k\}_{k\in\omega}$ say, there exists a degree* $\mathbf{a}$ *such that, for any $k \in \omega$, if $\boldsymbol{a} \in S(\mathcal{Q}_k)$ then $S(\mathcal{Q}_k) = \mathfrak{D}$. In fact $\boldsymbol{a}$ can be chosen to be hyperimmune-free and minimal.*

Proof. We will in fact show that the conclusion holds for a countable sequence of closed sets. That this is sufficient to imply the full theorem is immediate since any F_σ set is the union of a countable sequence of closed sets. More specifically, let $\{\mathcal{Q}_k\}_{k\in\omega}$ be a sequence of F_σ sets, with $\mathcal{Q}_k = \bigcup_{j\in\omega} \mathcal{P}_{k,j}$ with $\mathcal{P}_{k,j}$ closed for each $k, j \in \omega$, and let $\mathbf{a}$ be such that if $\boldsymbol{a} \in S(\mathcal{P}_{k,j})$ then $S(\mathcal{P}_{k,j}) = \mathfrak{D}$. If $\boldsymbol{a} \in S(\mathcal{Q}_k)$ then there exists a j such that $\boldsymbol{a} \in S(\mathcal{P}_{k,j})$, giving $S(\mathcal{Q}_k) = \mathfrak{D}$ as required. So suppose we are given a sequence $\{\mathcal{P}_k\}_{k\in\omega}$ of closed sets.

In order to show that any $\mathcal{P}$ contains a member of every degree it suffices to show that $[T] \subset \mathcal{P}$ for some 2-branching computable T. In order to see that this suffices, suppose that $\mathcal{P}$ contains all paths through T of this kind. Then we can define, given any set B, a set $C \in [T]$ such that $C = \bigcup_s \sigma_s$ and is of the same degree as B. Define σ_0 to be the string of level 0 in T. Given σ_s, define σ_{s+1} to be the leftmost successor of σ_s in T if $B(s) = 0$, and define σ_{s+1} to be the rightmost successor otherwise. Note also, for future reference, that the same argument suffices to show that any $\mathcal{P}$ containing $[T]$ for some perfect pointed $\boldsymbol{b}$-computable T, contains a member of every degree above $\boldsymbol{b}$.

For every k, let $\mathcal{P}_k = [\Upsilon_k]$ for some downward closed $\Upsilon_k \subseteq 2^{<\omega}$. We shall use a simple forcing argument in order to construct $A = \bigcup_s \sigma_s$ which is of hyperimmune-free minimal degree. Initially we define $\sigma_0 = \lambda$ and we define T_0 to be the identity tree. At every stage s, given computable 2-branching T_s and σ_s which is the string of level 0 in T_s, we define $\sigma_{s+1} \supset \sigma_s$ and $T_{s+1} \subset T_s$. At stage s we act according to the first of the following situations which applies. Let $s = \langle i, j, k\rangle$.

(1) There exists $\sigma \in T_s$ such that no two strings extending σ in T_s are Ψ_i-splitting. In this case we define σ_{s+1} to be the first such σ properly extending σ_s and we define T_{s+1} to be the set of strings in T_s extending σ. In so doing we have ensured that $\Psi_i(A)$ is either computable or partial.

(2) Since the previous case does not apply we may let T^0_s be a computable, 2-branching and Ψ_i-splitting subset of T_s containing σ_s (we shall eventually define T_{s+1} to be a subset of T^0_s). There exists $\sigma \in T^0_s$ such that either $\Psi_j(\Psi_i(\sigma))$ is incompatible with σ or else for no string $\tau \supset \sigma$ in T^0_s is it the case that $\Psi_j(\Psi_i(\tau))$ properly extends $\Psi_j(\Psi_i(\sigma))$. In this case we define σ_{s+1} to be the first such σ properly extending σ_s and we define T_{s+1} to be the set of strings in T^0_s extending σ. In so doing we have ensured it is not the case that $\Psi_j(\Psi_i(A)) = A$.

(3) Since the previous case does not apply we may let T^1_s be a computable and 2-branching subset of T^0_s containing σ_s such that, whenever $\tau, \tau' \in T^1_s$ and $\tau' \supset \tau$, we have that $\Psi_j(\Psi_i(\tau'))$ properly extends $\Psi_j(\Psi_i(\tau))$. There exists $\sigma \in T^1_s$ such that $\Psi_i(\sigma)$ is not in Υ_k. In this case we define σ_{s+1} to be the first such σ properly extending σ_s

and we define T_{s+1} to be the set of strings in T^1_s extending σ. In so doing we have ensured that $\Psi_i(A) \notin \mathcal{P}_k$.

(4) Since none of the previous cases apply we have that, for all $B \in [T^1_s]$, $\Psi_i(B) \in \mathcal{P}_k$ and $\Psi_j(\Psi_i(B)) = B$. Since T^1_s is 2-branching and computable it follows that $\mathcal{P}_s$ contains a member of every Turing degree. In this case we define σ_{s+1} to be some proper extension of σ_s in T^1_s and we define T_{s+1} to be the set of strings in T^1_s which extend σ_{s+1}.

The verification that A is of minimal degree is standard (that A is non-computable follows from Posner's lemma), see for example [BC]. In order to see that A is of hyperimmune-free degree, observe that whenever $\Psi_i(A)$ is total and non-computable, A lies on a computable 2-branching Ψ_i-splitting tree T. Thus, for any n, $\Psi_i(A;n)$ is amongst the values $\Psi_i(\tau;n)$ such that τ is of level $n+1$ in T.

□

If we restrict ourselves to the arithmetical hierarchy, then we can do a little better:

Theorem 4.2. *There exists a hyperimmune-free minimal degree* $\boldsymbol{a}$ *below* $\mathbf{0}''$*, such that no* Σ^0_3 *class contains a member of degree* $\boldsymbol{a}$ *unless it contains a member of every degree.*

Proof. For the same reasons that it sufficed to prove Theorem 4.1 for countable sequences of closed sets, it suffices to prove this result for Π^0_2 classes. If U is a set of finite binary strings then let $I(U)$ be the set of all A which extend some member of U. Let $\mathcal{P}_k$ be the k^{th} Π^0_2 class according to the standard indexing of such classes, and for each k let $\mathcal{P}_k = \bigcap_l I(U_{k,l})$ where $\{U_{k,l}\}_{l\in\omega}$ is a uniformly c.e. sequence of sets of strings.

The construction works in exactly the same way as that of Theorem 4.1 except that now case (3) is as follows:

(3) Since the previous case does not apply we may let T^1_s be a computable and 2-branching subset of T^0_s containing σ_s such that, whenever $\tau, \tau' \in T^1_s$ and $\tau' \supset \tau$, we have that $\Psi_j(\Psi_i(\tau'))$ properly extends $\Psi_j(\Psi_i(\tau))$. Now let T^2_s be the subset of T^1_s enumerated as follows:

Step $t = 0$. Enumerate the string of level 0 in T^1_s into T^2_s.

Step $t > 0$. For each leaf τ of the present value T^2_s, if τ is of level l then search for two incompatible extensions τ_0, τ_1 in T^1_s such that $\Psi_i(\tau_0)$ and $\Psi_i(\tau_1)$ extend strings in every $U_{k,l'}$ such that $l' \leq l$. If two such extensions are found then enumerate them into T^2_s.

It is easily seen that if T^2_s is 2-branching then it is computable. If not then there exists σ which is a leaf of T^2_s. In this case we define σ_{s+1} to be an element of T^1_s properly extending the first such σ and we define T_{s+1} to be the set of strings in T^1_s extending σ. In so doing we have ensured that $\Psi_i(A) \notin \mathcal{P}_k$.

In case (4) we must now replace T^1_s with T^2_s. □

An analysis of the proofs of Theorems 4.1 and 4.2 gives a characterization, for sufficiently low levels of the arithmetical and Borel hierarchies, of those sets which contain a member of every degree in an upper cone:

Theorem 4.3. *An F_σ set $\mathcal{Q}$ contains a member of every degree above $\boldsymbol{b}$ iff there exists some $\boldsymbol{b}$-computable perfect and pointed T with $[T] \subseteq \mathcal{Q}$.*

Proof. Let $\mathcal{Q} = \bigcup_k \mathcal{P}_k$ where each $\mathcal{P}_k$ is closed. Given B of degree $\boldsymbol{b}$, consider running the construction of Theorem 4.1, but beginning with T_0 as the set of strings of the form $\tau_0 \oplus \tau_1$ such that $\tau_0 \subset B$—so that all the T_s are now computable in B. Let the set A constructed be of degree $\boldsymbol{a}$. If $\mathcal{Q}$ contains a member of every degree above $\boldsymbol{b}$ then, in particular, some $\mathcal{P}_k$ must contain a member of degree $\boldsymbol{a}$. Therefore, for some $s = \langle i, j, k \rangle$ it must be that case (4) applies. Then $T = \{\Psi_i(\sigma) : \ \sigma \in T_{s+1}\}$ is B-computable perfect and pointed, and $[T] \subseteq \mathcal{Q}$. □

Theorem 4.4. *A Σ^0_3 class $\mathcal{P}$ contains a member of every degree above $\boldsymbol{b}$ iff there exists some $\boldsymbol{b}$-computable perfect and pointed T with $[T] \subseteq \mathcal{P}$.*

Proof. Almost exactly the same as the proof of Theorem 4.3. □

It follows easily from the fact that the class of all non-computable sets is Π^0_3 and G_δ, that Theorems 4.1,4.2, 4.3 and 4.4 are the best we can do—Theorems 4.1 and 4.3 do not hold for G_δ sets and Theorems 4.2 and 4.4 do not hold for Π^0_3 classes.

Definition 4.1. *We say that a degree* $\mathbf{a}$ *is* invisible *if any Π^0_1 class which contains a member of degree* $\mathbf{a}$ *contains a member of every degree.*

The existence of invisible degrees immediately suffices to give the following corollary.

Corollary 4.1. *There do not exist $\boldsymbol{\alpha} < \mathbf{1}_{\mathfrak{P}}$ and $\boldsymbol{\beta} < \mathbf{1}_{\mathfrak{P}}$ with $\boldsymbol{\alpha} \vee \boldsymbol{\beta} = \mathbf{1}_{\mathfrak{P}}$.*

Proof. If neither of $\boldsymbol{\alpha}$ or $\boldsymbol{\beta}$ contain any invisible degrees then neither does their union. □

A simple analysis tells us something about the distribution of the invisible degrees. Since $\mathbf{f} \in \mathfrak{P}$ no fixed point free degree is invisible. We say a degree is *completely unranked* if it does not contain a ranked set. Since no degree containing a ranked set is invisible—any ranked set is a member of a countable Π^0_1 class—the invisible degrees are a subclass of the completely unranked degrees and are all incomparable with $\mathbf{0}'$. Since there are only countably many ranked sets, the completely unranked degrees are of measure 1. Since the fixed point free degrees are of measure 1, the invisible degrees are of measure 0 and so are a proper subclass of the completely unranked degrees. It was shown in [AL] that any hyperimmune-free degree which is not fixed point free has a strong minimal cover (and where $\boldsymbol{b}$ is a strong minimal cover for $\boldsymbol{a}$ if the degrees strictly below $\boldsymbol{b}$ are precisely the degrees below and including $\boldsymbol{a}$). Thus all invisible degrees which are hyperimmune-free have a strong minimal cover.

Recall that $T \subseteq 2^{<\omega}$ is *dense* if for every τ there exists $\sigma \supseteq \tau$ in T, and that A is *weakly 2-generic* if for every dense set of strings T which is c.e. in $\mathbf{0}'$ there exists $\sigma \subset A$ which is an element of T. A degree is weakly 2-generic if it contains a weakly 2-generic set.

Theorem 4.5 (With Jockusch)**.** *Every weakly 2-generic degree is invisible.*

Proof. For every i, j, τ we try to define a string $\sigma(i, j, \tau)$. In order to do so, let T be a Ψ_i-splitting set of strings (enumerated in some uniform way), which has τ as the single element of level 0, such that each element which is not a leaf has precisely two successors, such that for any σ which is a leaf of T there does not exist a Ψ_i-splitting set of strings above σ, and such that at each stage of the enumeration of T we only enumerate in strings which properly extend leaves of the set of strings already enumerated into T. Let the strings in T be ordered according to their level and then from left to right. If there exists a least string σ in T such that either σ is a leaf of T, or else $\Psi_i(\sigma) \notin \Lambda_j$ then define $\sigma(i, j, \tau)$ to be that string. If there exists no such string then $\sigma(i, j, \tau)$ is undefined.

Now if $[\Lambda_j]$ does not contain a member of every degree, then for each i, the set of strings $T_i = \{\sigma(i, j, \tau) : \tau \in 2^{<\omega}\}$ is dense and c.e. in $\mathbf{0}'$. If A is weakly 2-generic then, for each i, there exists some $\sigma \subset A$ which is in T_i, so that either $\Psi_i(A)$ is partial or computable, or else is not an element of $[\Lambda_j]$. The result follows since no weakly 2-generic is computable. □

The reader may have noticed that the proof of Theorem 4.5 actually suffices to show that every weakly 2-generic degree is strongly invisible:

Definition 4.2. *We say that a degree $\boldsymbol{a} > \mathbf{0}$ is* strongly invisible *if any Π^0_1 class which contains a member of non-zero degree below $\boldsymbol{a}$ contains a member of every degree.*

Also, in the proofs of Theorems 4.1 and 4.2 we did not really need to consider ordered triples $\langle i, j, k\rangle$. It would have sufficed to consider ordered pairs $\langle i, k\rangle$, in order to deduce according to this modified proof of Theorem 4.2, for example, that the degree constructed is strongly invisible. Of course the fact that the degree constructed is minimal suffices to show it is strongly invisible anyway. This extra level of complication was used in order that we could immediately apply the proofs of Theorems 4.1 and 4.2 in order to deduce Theorems 4.3 and 4.4 (and also for later applications). Since the constructions of invisible degrees that we have described so far automatically produce strongly invisible degrees, however, it seems natural to ask whether all invisible degrees are strongly invisible. Note that this holds iff the invisible degrees are downward closed avoiding $\mathbf{0}$ i.e. iff any non-zero degree bounded by an invisible degree is invisible. In order to answer this question we need to consider a third category of invisibility:

Definition 4.3. *We say that a degree* $\mathbf{a}$ *is* weakly invisible *if any Π^0_1 class which contains a member of every degree above* $\mathbf{a}$ *contains a member of every degree.*

Theorem 4.6. *There exists a non-zero weakly invisible degree which is not invisible.*

Proof. We fix some effective bijection from ω to the finite subsets of $2^{<\omega}$ and we let $\{\Phi_i\}_{i\in\omega}$ be an effective listing of the Turing functionals Φ such that:

(i) for any σ, n, m if $\Phi(\sigma; n) \downarrow= m$, then m codes some T which is finitely 2-branching and of level n;

(ii) for any σ, m, m' and $n' > n$ if $\Phi(\sigma; n) \downarrow= m$ and $\Phi(\sigma; n') \downarrow= m'$, m codes T and m' codes T', then $T \subset T'$ and all strings in T are of the same level in T as they are in T'.

We construct downward closed computable Υ and A which is the only non-computable element of $[\Upsilon]$, so as to satisfy requirements:

Θ_i: $A \neq \Psi_i(\emptyset)$;

Ξ_i: Let $i = \langle j, k\rangle$. If $\Phi_j(A)$ is total and $[\Phi_j(A)] \subseteq [\Lambda_k]$ then $[\Lambda_k]$ contains a member of every degree.

It follows from Theorem 4.3 that this suffices to prove the result. The construction is most simply described using modules. The role of the module θ_i placed on σ will be to ensure that if $\sigma \subset A$ then Θ_i is satisfied, and the role of the module ξ_i placed on σ will be to ensure that if $\sigma \subset A$ then Ξ_i is satisfied. When any module is passed control at a stage s it will either have outcome 0 or outcome 1 at this stage. If the outcome of the module placed on σ is d ($\in \{0,1\}$) at stage s, it may then pass control its d-successor module which will be placed on a string extending $\sigma \star d$, or else define its d-successor module if it has not already done so. If the module is passed control at an infinite number of stages and has outcome 1, then $\sigma \star 1$ will be an initial segment of A. If the module is passed control at an infinite number of stages but has outcome 1 at only finitely many of these stages, then $\sigma \star 0$ will be an initial segment of A.

At the end of each stage s, if any string σ is a leaf of the present value Υ which has not been declared terminal, then we shall enumerate $\sigma \star 0$ into Υ and if a module is placed on σ then we shall enumerate also the string $\sigma \star 1$ into Υ. We use the variable υ to range over the modules.

The instructions for the module θ_i placed on σ, when passed control at stage s. If $\Psi_i(\emptyset)[s] \supseteq \sigma \star 0$ then the module has outcome 1, and otherwise it has outcome 0. Suppose the outcome at stage s is d. If the d-successor module is already defined, then pass control to it. Otherwise let σ' be the unique leaf of the present value Υ extending $\sigma \star d$, place the module ξ_i on σ', and define this to be the d-successor module for this module (without passing control to it at this stage).

The instructions for the module ξ_i placed on σ. Let υ be the module ξ_i placed on σ. The module uses a parameter τ_υ which is initially defined to be $\sigma\star 0$, and which may be redefined at any given stage at which the module is passed control to be some extension of its previous value. If there is some stage after which this value is never redefined then its final value will be an initial segment of A, and otherwise $\sigma \star 1$ will be an initial segment of A (providing υ is passed control at an infinite number of stages). The basic idea is that the module is prepared to keep A looking computable on a long initial segment (the final value τ_υ should this exist) extending $\sigma \star 0$, so that if it cannot ensure that either $\Phi_j(A)$ is partial or $[\Phi_j(A)] \nsubseteq [\Lambda_k]$ by making this initial segment τ_υ sufficiently long, then actually $[\Lambda_k]$ must contain $[T]$ for some 2-branching computable T. In this latter case, the requirement is automatically satisfied and $\sigma \star 1$ will be an initial segment of A. When the module is passed control at stage s it performs instructions according to the first case below which applies. Let n be the greatest such that $\Phi_j(\tau_\upsilon; n)\downarrow$, or if there exists no such then let $n = -1$.

(1) Either:
   (i) $n \geq 0$ and there exists some leaf of $\Phi_j(\tau_\upsilon; n)$ which has no extension in Λ_k of length s, or;
   (ii) there does not exist σ' which is a leaf of the present value Υ extending τ_υ which has not been declared terminal, and such that $\Phi_j(\sigma'; n')\downarrow$ for some $n' > n$.

   In this case the module has outcome 0. Pass control to the 0-successor module if this is already defined. Otherwise let σ'' be the unique leaf of the present value Υ extending τ_υ, place the module θ_{i+1} on σ'', and define this to be the 0-successor module for υ.

(2) Since (1) does not apply there must exist σ' as in $(1)(ii)$, so let σ' be such. Remove all modules from strings extending $\sigma \star 0$, declare all leaves of Υ extending $\sigma \star 0$ to be terminal except for σ', and redefine $\tau_\upsilon = \sigma'$. Make the 0-successor of υ undefined. The outcome of the module is 1 at stage s. If the 1-successor module for υ is already defined then pass control to it. Otherwise let σ'' be the unique leaf of the present value Υ extending $\sigma \star 1$, place the module θ_{i+1} on σ'', and define this to be the 1-successor module for υ.

The construction.

Stage 0. Enumerate λ, 0 and 1 into Υ and place the module θ_0 on λ.

Stage $s > 0$. Pass control to the module θ_0 placed on λ, passing control to other modules and carrying out their instructions according to the instructions for each module detailed above, until some module defines a successor module. For each leaf σ of the present value Υ which has not been declared terminal, enumerate $\sigma \star 0$ into Υ and if a module is placed on σ then enumerate also the string $\sigma \star 1$ into Υ.

We define $A = \bigcup_i \sigma_i$ as follows. We let υ_0 be the module θ_0 placed on λ. Suppose given υ_i, and suppose that υ_i is placed on σ. We define $\sigma_i = \sigma$. If there exists an infinite number of stages at which υ_i is passed control and has outcome 1 then let υ_{i+1} be the 1-successor of υ_i. Otherwise there exists υ which is defined to be the 0-successor of υ_i at some stage of the construction after which the 0-successor is never made undefined. Define $\upsilon_{i+1} = \upsilon$.

The verification. It is clear Υ is downward closed and computable, that $A \in [\Upsilon]$ and that all requirements Θ_i are satisfied. It is also clear that A is the only non-computable element of $[\Upsilon]$ since all other elements are either a finite string concatenated with an infinite string of zeros, or else are the infinite string extending all values taken by τ_υ for some fixed υ. In order to see that all requirements Ξ_i are satisfied, consider the action of υ_{2i+1} (as in the definition of A). Let $i = \langle j, k\rangle$. If $\Phi_j(A)$ is total and $[\Phi_j(A)] \subseteq [\Lambda_k]$ then it must be the case that this module has outcome 1 at an infinite number of stages. Let B be the infinite string extending all values taken by $\tau_{\upsilon_{2i+1}}$. Then B is computable, $\Phi_j(B)$ is total and $[\Phi_j(B)] \subseteq [\Lambda_k]$.

□

Theorem 4.7. *A degree is weakly invisible iff it is bounded by an invisible degree.*

Proof. Any degree bounded by an invisible degree is clearly weakly invisible, so it suffices to show that any weakly invisible degree is bounded by an invisible degree. In order to see this suppose given B of degree $\boldsymbol{b}$ which is weakly invisible, let $\mathcal{P}_k$ be the k^{th} Π^0_1 class and consider running the construction of Theorem 4.1, but beginning with T_0 as the set of strings of the form $\tau_0 \oplus \tau_1$ such that $\tau_0 \subset B$. Now whenever case (4) applies we may deduce that $\mathcal{P}_k$ contains $[T]$ for some perfect pointed $\boldsymbol{b}$-computable T. Since $\boldsymbol{b}$ is weakly invisible, $\mathcal{P}_k$ therefore contains a member of every degree. □

Corollary 4.2. *There exist invisible degrees which are not strongly invisible.*

Proof. By Theorem 4.6 there exists a weakly invisible degree which is non-zero and which is not invisible. By Theorem 4.7 this degree is bounded by an invisible degree which cannot be strongly invisible. □

We conclude this section by showing, as promised, that there exist degrees $\boldsymbol{a}$ containing a set of rank 1, and such that $\{\mathbf{0}\} \cup \{\mathbf{b} : \mathbf{b} \geq \mathbf{a}\}$ is not the degree spectrum of a Π^0_1 class (in fact this theorem can also be deduced from an analysis of the proof of Theorem 4.6 but the following lemmas give a more direct proof). The proof of the following theorem is essentially a generalization of Kucera's proof that every degree above $\mathbf{0}'$ is random.

Theorem 4.8. *Let $\mathcal{P}$ be a Π^0_1 class. If there exists a 2-branching T and a computable function f such that $[T] \subset \mathcal{P}$ and such that, for all l, all strings in T of level l are of length $\leq f(l)$, then $\mathcal{P}$ contains a member of every degree above $\mathbf{0}'$.*

Proof. Given $\mathcal{P}$, T and f as in the statement of the theorem we prove first that there exists some computable function f^* such that the set of all T' satisfying the following condition is non-empty:

($*$) T' is 2-branching, $[T'] \subseteq \mathcal{P}$ and, for all l, all strings in T' of level l are of length $f^*(l)$.

We may suppose that $f(0)$ is the length of the string of level 0 in T. Define $f^*(0) = f(0)$ and $f^*(1) = f(1)$ and for all $l \geq 1$ define $f^*(l{+}1) = f(f^*(l){+}1)$. It is clear that the set of T' satisfying ($*$) above is non-empty, since for any $A \in [T]$ and any l there exist at least two incompatible strings in T extending $A \restriction f^*(l)$ and of length at most $f^*(l+1)$.

Note that the set of T' satisfying ($*$) can be thought of as a Π^0_1 class when effectively coded in the appropriate way (and that the existence of f was essential in order that this should be the case).

Next we define $T^* \leq_T \emptyset'$ satisfying ($*$) above by recursion. Let the string of level 0 in T^* be the string of level 0 in T. Suppose we have defined all the strings of level l in T^*. For each σ of level l let σ_0 be the leftmost string extending σ of length $f^*(l+1)$ and which is in some T' satisfying ($*$). Let σ_1 be the rightmost string extending σ of length $f^*(l+1)$ and which is in some T' satisfying ($*$). Let σ_0 and σ_1 be the two successors of σ in T^*.

Given A of degree above $\mathbf{0}'$ we define $B = \bigcup_l \sigma_l$ such that $B \in \mathcal{P}$ and $B \equiv_T A$. Define σ_0 to be the string of level 0 in T^*. Given σ_l, define σ_{l+1}

to be the leftmost successor of σ_l in T^* if $A(l) = 0$ and define σ_{l+1} to be the rightmost successor otherwise. That $B \leq_T A$ is clear and that $A \leq_T B$ follows from the fact that any σ_{l+1} can compute whether it is the leftmost or rightmost string extending σ_l of length $f^*(l+1)$ and which is in some T' satisfying $(*)$. □

Corollary 4.3. *If $\boldsymbol{a}$ is hyperimmune-free then any Π^0_1 class which contains a member of every degree above $\boldsymbol{a}$ must also contain a member of every degree above $\mathbf{0}'$.*

Proof. If $\mathcal{P}$ contains a member of every degree above $\boldsymbol{a}$ then by Theorem 4.3 there exists 2-branching T such that $[T] \subseteq \mathcal{P}$ and which is of degree $\boldsymbol{a}$. If $\boldsymbol{a}$ is hyperimmune-free there exists computable f such that, for all l, all strings in T of level l are of length $\leq f(l)$. Thus by Theorem 4.8, $\mathcal{P}$ contains a member of every degree above $\mathbf{0}'$. □

Corollary 4.4. *There exists a degree $\boldsymbol{a}$ containing a set of rank 1 for which it is not the case that $\{\mathbf{0}\} \cup \{\mathbf{b} : \mathbf{b} \geq \mathbf{a}\}$ is the degree spectrum of a Π^0_1 class.*

Proof. This follows immediately from Corollary 4.3 since Downey [RD] has shown that there exists a hyperimmune-free degree $\boldsymbol{a}$ containing a set of rank 1 (and which is therefore non-zero). □

5. Minimal degree spectra

Definition 5.1. *We say that $\boldsymbol{\alpha} \neq \mathbf{0}_{\mathfrak{P}}$ is subclass invariant if for any Π^0_1 class $\mathcal{P}$ with $S(\mathcal{P}) = \boldsymbol{\alpha}$ and any nonempty Π^0_1 class $\mathcal{P}' \subseteq \mathcal{P}$, $S(\mathcal{P}') = \boldsymbol{\alpha}$. We say that $\boldsymbol{\alpha} \neq \mathbf{0}_{\mathfrak{P}}$ is weakly subclass invariant if there exists a Π^0_1 class $\mathcal{P}$ with $S(\mathcal{P}) = \boldsymbol{\alpha}$ and for any nonempty Π^0_1 class $\mathcal{P}' \subseteq \mathcal{P}$, $S(\mathcal{P}') = \boldsymbol{\alpha}$.*

Clearly any $\boldsymbol{\alpha}$ which is minimal must be subclass invariant. Now suppose that $\boldsymbol{\alpha}$ is subclass invariant, let $\mathcal{P}$ be a Π^0_1 class with $S(\mathcal{P}) = \boldsymbol{\alpha}$ and suppose that $\mathcal{P}'$ is a nonempty Π^0_1 class with $S(\mathcal{P}') \subset \boldsymbol{\alpha}$. Then $\{0{\star}A : A \in \mathcal{P}\} \cup \{1{\star}A : A \in \mathcal{P}'\}$ is a Π^0_1 class with degree spectrum $\boldsymbol{\alpha}$ and witnesses the fact that $\boldsymbol{\alpha}$ is not subclass invariant, a contradiction. Thus being subclass invariant is equivalent to minimality.

Theorem 5.1. *Suppose that $\boldsymbol{\alpha}$ is weakly subclass invariant. If a Π^0_1 class contains any member of any hyperimmune-free degree in $\boldsymbol{\alpha}$ then it contains a member of every degree in $\boldsymbol{\alpha}$.*

Proof. Let $\mathcal{P}_0$ be a Π^0_1 class with $S(\mathcal{P}_0) = \boldsymbol{\alpha}$ and such that for any nonempty Π^0_1 class $\mathcal{P} \subseteq \mathcal{P}_0$, $S(\mathcal{P}) = \boldsymbol{\alpha}$. Let $\mathcal{P}_1$ be a Π^0_1 class which contains A of hyperimmune-free degree in $\boldsymbol{\alpha}$. Then there exists $B \equiv_{tt} A$ and which is in $\mathcal{P}_0$. Let i, j be such that the total functionals Φ_i and Φ_j satisfy $\Phi_i(A) = B$ and $\Phi_j(B) = A$. Now let $\mathcal{P}_2$ be the Π^0_1 class $\{C : C \in \mathcal{P}_0 \ \& \ (\exists D \in \mathcal{P}_1)[\Phi_i(D) = C \ \& \ \Phi_j(C) = D]\}$. Then $S(\mathcal{P}_2) \subseteq S(\mathcal{P}_1)$. Since $\mathcal{P}_2$ is nonempty and $\mathcal{P}_2 \subseteq \mathcal{P}_0$, $S(\mathcal{P}_2)$ and $S(\mathcal{P}_1)$ contain every degree in $\boldsymbol{\alpha}$. □

Corollary 5.1. *$\boldsymbol{\alpha}$ is minimal iff it is weakly subclass invariant.*

Proof. If $\boldsymbol{\alpha}$ is weakly subclass invariant then, by the hyperimmune-free basis theorem, any nonempty Π^0_1 class which contains only members of degree in $\boldsymbol{\alpha}$ contains a member of hyperimmune-free degree in $\boldsymbol{\alpha}$ and therefore, by Theorem 5.1, contains a member of every degree in $\boldsymbol{\alpha}$. □

Corollary 5.2. *$\boldsymbol{r}$ is minimal in $(\mathfrak{P}, <)$.*

Proof. Any nonempty Π^0_1 class containing only random sets is witness to the fact that $\boldsymbol{r}$ is weakly subclass invariant since any Π^0_1 class containing a random set is of positive measure and any Π^0_1 class of positive measure contains a member of every random degree [AK]. □

The random degrees, then, are an example of a degree spectrum $\boldsymbol{\alpha}$ which satisfies this nice property that there exists a Π^0_1 class $\mathcal{P}$ such that $S(\mathcal{P}) = \boldsymbol{\alpha}$ and such that any nonempty Π^0_1 class $\mathcal{P}' \subseteq \mathcal{P}$ has degree spectrum $\boldsymbol{\alpha}$. Corollary 5.1 says that the minimal elements of $\mathfrak{P}$ are precisely those degree spectra which satisfy this property.

Recall that a degree is PA iff it contains a $\{0,1\}$-valued DNC function. If $\mathcal{P}_0$ and $\mathcal{P}_1$ are Π^0_1 classes and there exist $A \in \mathcal{P}_0$, $B \in \mathcal{P}_1$ with $B \leq_{tt} A$ then it is easy to see that there exists a non-empty Π^0_1 class $\mathcal{P}_2 \subseteq \mathcal{P}_0$ such that every element of $\mathcal{P}_2$ tt-computes a member of $\mathcal{P}_1$. We may immediately deduce that no random set tt-computes a $\{0,1\}$-valued DNC function, since no hyperimmune-free random does (if $\boldsymbol{a}$ is random and hyperimmune-free then every non-zero $\boldsymbol{b} < \boldsymbol{a}$ is random [OD] and is not PA [FS]). Theorem 5.2 below was first proved by Simpson. A sketch of the following short proof was provided by Kucera in a private correspondence.

Theorem 5.2 (Simpson [SS2])**.** *Any non-empty Π^0_1 class containing only $\{0,1\}$-valued DNC functions contains a member of every PA degree.*

Proof. If Λ is computable and downward closed then consider $\Psi_i(\emptyset)$ such that $\Psi_i(\emptyset; i)\downarrow = n$ iff there exists some $l > i$ such that $\tau(i) = n$ for all $\tau \in \Lambda$ of length l. By the uniformity of the recursion theorem it follows that there exists computable f such that, whenever $[\Lambda_j]$ is non-empty and contains only $\{0,1\}$-valued DNC functions, there exist $A, B \in [\Lambda_j]$ with $A(f(j)) = 0$ and $B(f(j)) = 1$.

Now suppose given j_0 such that $[\Lambda_{j_0}]$ is non-empty and contains only $\{0,1\}$-valued DNC functions. Let A be a $\{0,1\}$-valued DNC function. We construct $B = \bigcup_s \sigma_s$ which is in $[\Lambda_{j_0}]$ and is of the same degree as A.

Stage 0. Define $\sigma_0 = \lambda$.

Stage $s > 0$. We have already decided j_{s-1} and σ_{s-1}. There exists $C \in [\Lambda_{j_{s-1}}]$ with $C(f(j_{s-1})) = A(s-1)$. Using the oracle for A we can therefore compute σ of length $f(j_{s-1}) + 1$ such that $\sigma(f(j_{s-1})) = A(s-1)$ and which is an initial segment of some $C \in [\Lambda_{j_{s-1}}]$ (this follows using the standard argument that any $\{0,1\}$-valued DNC function computes a member of any non-empty Π^0_1 class). Define $\sigma_s = \sigma$ and define j_s so that $[\Lambda_{j_s}]$ is the set of all $C \in [\Lambda_{j_{s-1}}]$ which extend σ.

That B computes A follows from the fact that an oracle for B allows us to retrace every step of the construction defining B.

□

Corollary 5.3 (Simpson [SS2])**.** *$\boldsymbol{p}$ is minimal in $(\mathfrak{P}, <)$.*

Proof. By Theorem 5.2 and Corollary 5.1. □

Corollary 5.3 tells us that, while there are sentences which are independent of PA, from a degree theoretic point of view it is already 'saturated', in the sense that any consistent and axiomatizable extension of the theory has complete and consistent extensions of precisely the same degrees as PA does. The latter result was originally proved by Kripke and Pour-El [KP] who proved, in fact, that the set of consistent completions of any consistent axiomatizable extension of PA is recursively homeomorphic to the set of consistent complete extensions of PA. In fact, since every Π^0_1 class is recursively homeomorphic to the class of complete extensions of some finitely axiomatizable theory (a result due to Hanf and Peretyakin [HP]), the minimal elements of $\mathfrak{P}$ are precisely the degree spectra corresponding to axiomatizable theories which are saturated in this sense.

We shall say that T is *homogenous* if all strings of the same level in T are of the same length and, whenever σ_0 and σ_1 are of the same level in T and $\sigma_0 \star \tau \in T$, $\sigma_1 \star \tau$ is also in T. Recall that a Π^0_1 class $\mathcal{P}$ is *thin* if for any Π^0_1 class $\mathcal{P}' \subset \mathcal{P}$ there exists a clopen set $\mathcal{Q}$ such that $\mathcal{P}' = \mathcal{P} \cap \mathcal{Q}$.

Theorem 5.3. *For any $\boldsymbol{\alpha} < \mathbf{1}_{\mathfrak{P}}$ there exists $\boldsymbol{\beta}$ which is minimal and such that $\boldsymbol{\beta} \not\leq \boldsymbol{\alpha}$.*

Proof. We suppose given downward closed computable Λ such that $S([\Lambda]) = \boldsymbol{\alpha}$ is not the set of all Turing degrees, and we approximate a 2-branching and homogenous T such that $\mathcal{P} = [T]$ is a thin Π^0_1 class. At every stage of the construction our approximation to T will be a finitely 2-branching set of strings. In order to ensure that $\mathcal{P}$ is a Π^0_1 class it suffices that, if at any stage we redefine the strings in T of level l, we define each of these strings to extend leaves of the previous version of T, and we enumerate strings into T properly extending each of its leaves at the end of each stage. The fact that T is 2-branching and homogenous and that $\mathcal{P}$ is thin, suffices to show that $\boldsymbol{\beta} = S(\mathcal{P})$ is minimal and does not contain $\mathbf{0}$. We shall also approximate $A \in \mathcal{P}$ such that any non-computable $B \leq_T A$ is not in $[\Lambda]$. Since $\mathcal{P}$ does not have any computable members, this suffices to ensure that $\boldsymbol{\beta} \not\leq \boldsymbol{\alpha}$. We must satisfy all requirements:

Θ_i: If $[\Lambda_i] \subset \mathcal{P}$ then there exists a clopen $\mathcal{Q}$ such that $[\Lambda_i] = \mathcal{P} \cap \mathcal{Q}$.

Ξ_i: If $\Psi_i(A)$ is total and non-computable then $\Psi_i(A) \notin [\Lambda]$.

There is a difficulty inherent in looking to satisfy the requirements Ξ_i. The basic idea behind the module to satisfy the requirement is that, given some string σ which is potentially an initial segment of A, we should search for Ψ_i-splittings above this string. Either we can find some string above which there is no Ψ_i-splitting, or we can find some string τ such that $\Psi_i(\tau)$ is not in Λ, or else we can enumerate a perfect Ψ_i-splitting tree whose image under Ψ_i is a perfect and computable tree with all infinite paths in $[\Lambda]$—contradicting the fact that the latter set does not contain a member of every degree. The complication, however, is that carrying out this procedure requires us to search for Ψ_i-splittings inside some $\mathcal{Q}$ which is a Π^0_1 subclass of $\mathcal{P}$. It is quite possible to have a Π^0_1 class $\mathcal{Q}$ and $A \in \mathcal{Q}$ such that $\Psi_i(A)$ is total and non-computable, but such that no two initial segments of elements of $\mathcal{Q}$ are Ψ_i-splitting. The solution we shall adopt is to construct A which is of hyperimmune-free degree. Then if $\Psi_i(A) = B$, there exists a total Turing functional Φ such that $\Phi(A) = B$. The totality of Φ means that $\{\Phi(C) : C \in \mathcal{Q}\}$ is a Π^0_1 class, and so cannot consist of a single noncomputable member. We therefore construct A which is of hyperimmune-free degree and act in order to satisfy all requirements Ξ_i. Since we succeed for all i such that Ψ_i is total, this suffices to show that, in fact, we succeed for all i.

Ultimately, then, we shall have to ensure for every i that either $\Psi_i(A)$ is partial, or else A lies on some computable T_i satisfying the property that, for any string σ of level $l + 1$, $\Psi_i(\sigma; l) \downarrow$. We shall ensure that each string in T_i eventually either has two successors defined, or else is irrevocably declared terminal. If A lies on T_i of this kind then it is not difficult to see that $\Psi_i(A)$ is majorized by a computable function. Rather than immediately presenting the full construction, however, we shall describe first a simplified version which satisfies all requirements Θ_i, and which satisfies all requirements Ξ_i such that Ψ_i is a total functional, but which does not act in order to ensure that A is of hyperimmune-free degree.

The module θ_i. In order to satisfy requirement Θ_i we may place at some stage of the construction and for some l, a module θ_i on all strings of level l in our present approximation to T. At every stage strictly after this, until it has been declared satisfied (or else is removed), the module θ_i placed on σ searches for some string $\sigma \star \tau$ which is a leaf of T and which is not in Λ_i. If it finds such a string then we say that the module requires attention. If we then act for this module, it will remove all strings of level $> l$ from T together with any modules or submodules placed on these strings, and for each string σ' of level l it will enumerate $\sigma' \star \tau \star 0$ and $\sigma' \star \tau \star 1$ into T, before declaring itself satisfied.

Any module θ_i placed on σ of level l will subsequently be removed iff our approximation to T changes at a level $\leq l$. Now suppose that, subsequent to placing at some stage of the construction a module θ_i on each string in T of level l, these modules are never removed. Suppose also that for each of these modules there is either a stage after which it never requires attention or else a stage at which it is declared satisfied. Then for each σ of level l in T, $[\Lambda_i]$ either contains every element of $\mathcal{P}$ extending σ, or else contains no such element of $\mathcal{P}$.

The module ξ_i. In order to satisfy requirement Ξ_i in the case that Ψ_i is total we utilize the fact that $[\Lambda]$ cannot contain the set of all paths through any 2-branching computable T'. In this simplified version of the construction we can define a string σ_s at each stage s, so that the sequence $\{\sigma_s\}_{s \in \omega}$ approximates A. Suppose that, at some stage s, σ is a string of level l in our present approximation to T and that σ extends σ_s. If we place the module ξ_i on σ then we also place a submodule $\xi_i(0)$ on this string. It is the role of the module ξ_i and all of its submodules to ensure that Ξ_i is satisfied.

At every stage strictly after being placed on σ, until it has been declared satisfied (or else is removed), the module ξ_i searches for $\sigma \star \tau$ which is a leaf of our present approximation to T and such that $\Psi_i(\sigma \star \tau) \notin \Lambda$. If it finds such a string then we say that the module requires attention. If we then act for this module, it will remove all strings of level $> l$ from T together with any modules or submodules placed on these strings, and for each string σ' of level l it will enumerate $\sigma' \star \tau \star 0$ and $\sigma' \star \tau \star 1$ into T, before declaring itself satisfied.

It is the role of the submodules to ensure that if Ψ_i is total and $\Psi_i(A)$ is non-computable and if the module ξ_i placed on σ is never declared satisfied, then $[\Lambda]$ must contain $[T']$ for some 2-branching computable T', a contradiction. As the construction progresses we may have to place more submodules $\xi_i(j)$ on strings in T extending σ, but this process will only continue for a finite number of stages—only a finite number of submodules of this form will be placed on strings in T. If the submodule $\xi_i(j)$ is placed on $\sigma' \supseteq \sigma$ then this will mean that σ' is of level $l' = l + j$ in T and that the set of strings of the form $\Psi_i(\sigma'')$ such that some submodule $\xi_i(j')$ for $j' \leq j$ is placed on σ'' is finitely 2-branching and of level j. At every stage strictly after being placed on σ' until it has been declared satisfied (or else is removed), the submodule $\xi_i(j)$ acts as follows. Let $\sigma' \star \tau_0$ and $\sigma' \star \tau_1$ be the leftmost and rightmost successors of σ' in T respectively. Let U be the set of all strings σ'' strictly to the left of σ' on which submodules $\xi_i(j)$ have been placed (all these strings will also be of level l'). If it is not the case that all submodules $\xi_i(j)$ placed on strings in U have already been declared satisfied, then the submodule performs no action at this stage.

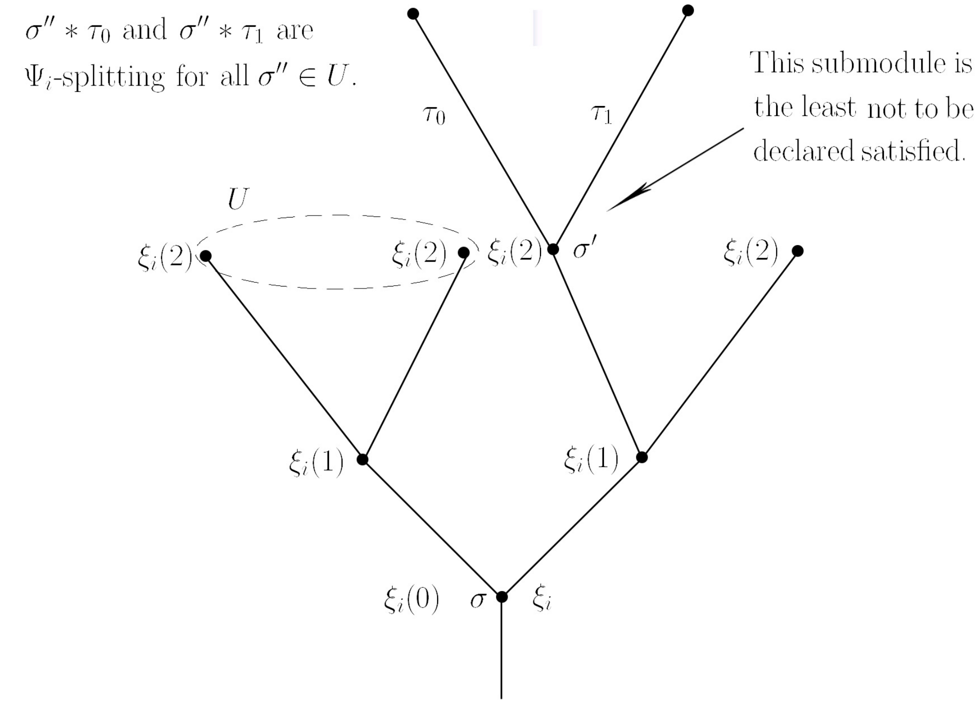
$\sigma'' * \tau_0$ and $\sigma'' * \tau_1$ are
Ψ_i-splitting for all $\sigma'' \in U$.
τ_0
τ_1
This submodule is the least not to be declared satisfied.
U
$\xi_i(2)$
$\xi_i(2)$
$\xi_i(2)$
σ'
$\xi_i(2)$
$\xi_i(1)$
$\xi_i(1)$
$\xi_i(0)$
σ
ξ_i

Figure 1. The placement of submodules.

Otherwise, it performs *one* of the instructions below according to the state that it is presently in. The submodules $\xi_i(j)$ placed on strings in U

will have chosen τ_0 and τ_1 so that for each string $\sigma'' \in U$, $\sigma'' \star \tau_0$ and $\sigma'' \star \tau_1$ are Ψ_i-splitting. Now this submodule aims to extend this process by finding a splitting above $\sigma' \star \tau_0$, a splitting above $\sigma' \star \tau_1$ and then choosing one string from each of these splittings in order to give $\tau_0^\star$ and $\tau_1^\star$ such that $\sigma'' \star \tau_0^\star$ and $\sigma'' \star \tau_1^\star$ are Ψ_i-splitting for each $\sigma'' \in U \cup \{\sigma'\}$. The submodule is initially in state 0.

State 0. The submodule searches for $\sigma' \star \tau_0 \star \tau_2$ and $\sigma' \star \tau_0 \star \tau_3$ which are leaves of the present approximation to T and which are Ψ_i-splitting. If it finds such strings then we say that the submodule requires attention. If we then act for the submodule it will remove all strings from T of level $> l' + 1$ together with any modules or submodules placed on these strings, and for each string σ'' of level $l' + 1$ it will enumerate $\sigma'' \star \tau_2$ and $\sigma'' \star \tau_3$ into T, before declaring itself to be in state 1.

State 1. The submodule searches for $\sigma' \star \tau_1 \star \tau_4$ and $\sigma' \star \tau_1 \star \tau_5$ which are leaves of the present approximation to T and which are Ψ_i-splitting. If it finds such strings then we say that the submodule requires attention. If we then act for the submodule it will proceed as follows. Choose $\tau_0^\star$ such that $\sigma' \star \tau_0^\star$ is a leaf of the present approximation to T extending one of $\{\sigma' \star \tau_0 \star \tau_2, \sigma' \star \tau_0 \star \tau_3\}$ and $\tau_1^\star \in \{\tau_1 \star \tau_4, \tau_1 \star \tau_5\}$, such that $\sigma' \star \tau_0^\star$ and $\sigma' \star \tau_1^\star$ are Ψ_i-splitting. Remove all strings of level $> l'$ from T together with any modules or submodules placed on these strings and for each σ'' of level l' enumerate $\sigma'' \star \tau_0^\star$ and $\sigma'' \star \tau_1^\star$ into T, before declaring the submodule to be satisfied.

Hopefully it is reasonably clear how we should place modules and submodules on strings in T in order to define this simplified version of the construction. Since this is not the final version of the construction we shall not describe it in absolute detail. On each level of T we place only one kind of module or submodule (except in the case of modules ξ_i when a module $\xi_i(0)$ will also be placed on the same string). If we have placed modules θ_i on all strings of the present approximation to T of level l, then we place a module ξ_i together with the submodule $\xi_i(0)$ on some string of level $l + 1$ which extends σ_s, our present approximation to A. If all submodules $\xi_i(j)$ placed on strings of level l' have been declared satisfied, then above each of these we place two submodules $\xi_i(j + 1)$ on strings of level $l' + 1$. If, on the other hand, one of these submodules is yet to be declared satisfied, we place modules θ_{i+1} on all strings of level $l' + 2$. If ξ_i has been placed on a string in T and is satisfied, then we place modules θ_{i+1} on all strings of the next level.

The modules are prioritized first according to the level of the string in our present approximation to T on which they are placed (with lower levels being of higher priority), and then in order from left to right. A module ξ_i placed on σ has higher priority than a submodule $\xi_i(0)$ placed on the same string. At each stage, if any module or submodule requires attention then we act for that of highest priority. The modules ξ_i and the submodules $\xi_i(j)$ determine our approximation to A at each stage in the obvious way. At the end of each stage, for each leaf σ of the present T, we enumerate $\sigma \star 0$ and

$\sigma \star 1$ into T.

Modifying the construction. In order to ensure that A is of hyperimmune-free degree, the entire construction must be carried out within nested trees of the kind described previously. Computable sets of strings T_i, that is, such that for any string σ of level $l+1$, $\Psi_i(\sigma; l) \downarrow$. Each string in T_i must eventually either be declared terminal or else have two successors defined.

We begin, then, by attempting to enumerate an infinite tree T_0. Responsibility for this task is given to a supermodule υ_0. At each stage of the construction this supermodule considers the leaves of T_0 to be ordered first according to their level in the tree and then from left to right, and the supermodule searches for strings to enumerate into the tree extending the least leaf which has not been declared terminal, σ say. If the supermodule never finds such a pair of strings then we define A so as to extend σ, and in doing so we ensure that $\Psi_0(A)$ is partial. While searching for strings to enumerate into T_0 extending σ we must continue with a construction which works above σ and which assumes that $\Psi_0(A)$ will be partial. Above σ, then, we work to satisfy requirements Θ_0 and then Ξ_0 before beginning construction of a tree T_1.

As the tree T_0 continues to be enumerated, on the other hand, we can proceed with a construction which takes place *inside* T_0 i.e. which assumes that A will lie on T_0. After placing modules θ_0 and ξ_0, however, we cannot proceed just as we did for the simplified construction but now working inside T_0, since so far we have only ensured (so long as A lies on T_0) that $\Psi_0(A)$ is majorized by a computable function. Next we must deal with Ψ_1. We therefore begin enumeration of a tree T_1 inside T_0, and this task is assigned to another supermodule. Note, however, that as observed in the discussion above, we also had to enumerate a tree T_1 above σ which was the least leaf of T_0 not to be declared terminal, while we searched for strings to enumerate into T_0 extending σ. In fact, then, we need to construct various candidates $T_{i,k}$ for each T_i, where $T_{i,k}$ may be regarded as the k^{th} such candidate. Suppose that $T_{0,0}$ (our only candidate for T_0) turns out to be infinite. If $T_{1,k}$, the candidate for T_1 that we construct inside T_0, turns out to be finite then A will extend the least leaf of this tree which is not declared to be terminal. If $T_{1,k}$ turns out to be infinite, then we will continue a construction which works within this tree, A will be a path through this tree, and so on.

There are some small technical issues that are worth discussing before describing the construction precisely. Up until this point, we have described modules which search for strings satisfying certain properties amongst the *leaves* of T. As previously described, the module ξ_i placed on σ of level l in T, for example, searches for $\sigma\star\tau$ which is a leaf of our present approximation to T and such that $\Psi_i(\sigma \star \tau) \notin \Lambda$. If such a string is found then we remove all strings from T of level $> l$ and redefine the strings of level $l+1$ in some appropriate way. Now, however, suppose that we are constructing $T_{1,k}$ inside $T_{0,0}$. Suppose that we are searching for a pair of strings to enumerate into $T_{1,k}$ extending σ, which is also a string in $T_{0,0}$ but which is not a leaf of this tree. If we only search amongst strings which are leaves of $T_{0,0}$, then these strings may not be leaves of T. Upon enumerating these stings into $T_{1,k}$ we would have to be careful about the way in which we redefine T in order to accommodate this enumeration. While this would not be very difficult to do, a slicker approach is actually just to search for strings to enumerate into $T_{1,k}$ amongst the leaves of T as before (which will extend leaves of $T_{0,0}$). Upon enumerating such strings into $T_{1,k}$ we may then have to redefine some of the leaves of $T_{0,0}$ extending σ so as to be extensions of their previous values, in order that $T_{1,k}$ should remain a subtree of $T_{0,0}$. Since it is basically only the leaves of any of the $T_{i,k}$ which may be redefined a finite number of times each in this way, each of the $T_{i,k}$ will still be computable. In order to avoid much repetition in what follows, then, we agree now to the following conventions which dictate action which is automatically to be taken whenever our approximation to T is modified.

Some conventions. Suppose that all strings of level $> l$ are removed from T and $\sigma \star \tau_0$ and $\sigma \star \tau_1$ are enumerated into T for every string σ of level l.

C1 We remove all supermodules, modules or submodules placed on strings of level $> l$ in T.

C2 Suppose σ is of level l in T, $\sigma' \supseteq \sigma$ is in $T_{i,k}$, τ_2 is a leaf of $T_{i,k}$, and that τ_2 and τ_3 are the successors of σ' in $T_{i,k}$.

- If $\tau_2 \subset \sigma \star \tau_0$ and $\tau_2 \subset \sigma \star \tau_1$ then we remove τ_2 and τ_3 from $T_{i,k}$ (together with any strings extending τ_3) and enumerate $\sigma \star \tau_0$ and $\sigma \star \tau_1$ into $T_{i,k}$.
- If $\tau_2 \subset \sigma \star \tau_0$ but $\tau_2 \not\subset \sigma \star \tau_1$, then we remove τ_2 from $T_{i,k}$ and enumerate $\sigma \star \tau_0$ into $T_{i,k}$.

C3 Any leaf of any tree which is not compatible with any leaf of T is declared terminal.

All submodules, modules and supermodules are prioritized just as before; first according to the level of the string in our present approximation to T on which they are placed, and then in order from left to right. A module ξ_i placed on σ has higher priority than a submodule $\xi_i(0)$ placed on the same string. All trees are considered to be prioritized according to the supermodule with which they are associated. When we pass control to a module it is useful to have some tidy way of recording the nested set of trees that this module works within. At any point in the construction and for any σ, we define $\boldsymbol{T}(\sigma)$ to be the set of all $T_{i,k}$ (associated with supermodules placed on strings in T and which have not been removed) such that $\sigma \in T_{i,k}$ but is not a leaf of this tree. $\boldsymbol{T}(\sigma)$, then, is the nested set of trees which any module placed on σ *believes* to be infinite. Any module, submodule or supermodule placed on the leaf of any tree believes that tree to be finite. Let T' be the lowest priority tree in $\boldsymbol{T}(\sigma)$ when the supermodule υ_i is first passed control, having been placed on σ (or if this set is empty then let $T' = T$). T' is the tree within which the supermodule must build its candidate for T_i.

The supermodule υ_i. When placed on the string σ the supermodule υ_i will find the least k such that $T_{i,k}$ is not yet associated with any supermodule (in other words it will choose some fresh tree to begin enumerating), and will enumerate the string σ into $T_{i,k}$. We shall say that $T_{i,k}$ is the tree associated with the supermodule. Let T' be the lowest priority tree in $\boldsymbol{T}(\sigma)$ when the supermodule is first passed control, or if this set is empty then let $T' = T$. T' is the tree within which the supermodule must build $T_{i,k}$.

At each stage strictly after that at which it is placed on σ (until any point at which it is removed) the supermodule will proceed as follows. Let σ' be the least leaf of $T_{i,k}$ which has not been declared terminal and suppose this string is of level l in $T_{i,k}$ and of level l' in T. If σ' is not a leaf of T' then the supermodule checks to see if there exists $\sigma' \star \tau$ which is a leaf of T and such that $\Psi_i(\sigma' \star \tau; l) \downarrow$ (if σ' is a leaf of T' then we cannot yet continue to build $T_{i,k}$ above this string and must wait for further strings to be enumerated into T'). If it finds such a string then we say that the supermodule requires attention. If we then act for the supermodule, it will perform the following instructions.

(1) Remove all strings of level $> l'$ from T. For each string σ'' of level l' in T, enumerate $\sigma'' \star \tau \star 0$ and $\sigma'' \star \tau \star 1$ into T.
(2) Enumerate $\sigma' \star \tau \star 0$ and $\sigma' \star \tau \star 1$ into $T_{i,k}$ (note that (1) may already have caused changes to $T_{i,k}$ according to C2).

In what follows, we say that $T_{i,k}$ *extends to level* l if all leaves of this tree which have not been declared terminal are of level at least l in T. Given any $\boldsymbol{T}(\sigma)$, let T' be the lowest priority tree in this set, or if this set is empty then let $T' = T$. We say that $\boldsymbol{T}(\sigma)$ extends to level l if T' does.

The instructions for the modules θ_i and ξ_i, and for the submodules $\xi_i(j)$ are exactly as previously described, except for one small modification. Suppose that the submodule $\xi_i(j)$ is placed on σ of level l in T. The submodule does not act before the first stage at which it is the least submodule (for the corresponding module ξ_i) not to be declared satisfied and $\boldsymbol{T}(\sigma)$ extends to level $l + 2$.

In the description of the module ξ_i we discussed being able to give an approximation to A at each stage of the construction. This situation no longer holds, but that fact does not have any effect on the instructions defined in that section.

Guessing initial segments of A. In the simplified version of the construction we could approximate A. Of course, now that A must be hyperimmune-free (and non-computable) it cannot be of degree below $\mathbf{0}'$. Rather than approximate A during the construction we simply define it at the end of the construction to be the only infinite path through all trees $T_{i,k}$ which turn out to be infinite. Nevertheless we do need to have some way of defining guesses as to initial segments of A during the course of the construction, based on guesses as to which of the trees $T_{i,k}$ will be infinite.

Let $\boldsymbol{T} = \{T_{i_0,k_0}, \cdots, T_{i_n,k_n}\}$ $(= \boldsymbol{T}(\sigma')$ for some string $\sigma')$ be such that, for $j < j'$, the supermodule associated with $T_{i_{j'},k_{j'}}$ is placed on a string in T_{i_j,k_j} which is not a leaf. The indices, then, order the trees according to the priority of their associated supermodules and, according to this ordering, each tree is built inside its immediate predecessor. We shall call a set of trees of this kind, a *cluster* of trees. At each point of the construction we define a value $\sigma(\boldsymbol{T})$ which is a finite binary string which should be an initial segment of A should all the trees in $\boldsymbol{T}$ turn out to be infinite, and all other trees associated with supermodules placed on strings in these trees at this point in the construction turn out to be finite. We define $\sigma(\boldsymbol{T})$ by a finite iterated process, where at each stage j in this process we define a value σ_j extending all previous values. At even stages in this process we look to see which string the next ξ_i module placed on a string extending σ_j inside this cluster of trees presently believes should be an initial segment of A. At odd stages we consider the next supermodule which tries to build a tree inside this cluster of trees, but which is not in $\boldsymbol{T}$. The least non-terminal leaf of this tree determines σ_{j+1}. Let σ_0 be the string on which T_{i_n,k_n} is placed and let $T' = T_{i_n,k_n}$, or if $\boldsymbol{T}$ is empty then let $\sigma_0 = \lambda$ and $T' = T$.

Given σ_j such that j is even: if there is a least module ξ_i (for any i and ordered according to their level in T) placed on a string $\sigma \in T'$ which is not a leaf and which extends σ_j then proceed as follows, and otherwise define $\sigma(\boldsymbol{T}) = \sigma_j$ and terminate the iterated process. If the module ξ_i placed on σ has been declared satisfied then define $\sigma_{j+1} = \sigma$. Otherwise let $\xi_i(j)$ placed on σ' be the lowest priority submodule of this module which has not been declared satisfied and define $\sigma_{j+1} = \sigma'$, or if there is no such define $\sigma(\boldsymbol{T}) = \sigma$ and terminate the iterated process.

Given σ_j such that j is odd: if there is a least supermodule υ_i placed on a string $\sigma \in T'$ which is not a leaf of T' and which extends σ_j then proceed as follows, and otherwise define $\sigma(\boldsymbol{T}) = \sigma_j$ and terminate the iterated process. Let σ' be the least leaf of the tree associated with υ_i placed on σ which has not been declared terminal. Define $\sigma_{j+1} = \sigma'$.

The construction. At each stage s we proceed as follows.

Stage $s = 0$. Enumerate λ, 0 and 1 into T. Place the supermodule υ_0 on λ.
Stage $s > 0$. If any supermodule requires attention, then act for that of highest priority. Then, if any other (different) supermodule requires attention act for that of highest priority, and so on, until no supermodule requires attention. If any module or submodule then requires attention act for that of highest priority.

Next we place sufficient supermodules, modules and submodules on strings in T (and no more), in order to guarantee that all of the following conditions are satisfied. These instructions basically just say that the previous simplified version of the construction should be carried out inside each cluster of trees, and that every so often we must begin building new trees.

(1) **We continue to build the simplified construction inside a cluster.**
   (a) If the module θ_i is placed on σ of level l in T, then let $\boldsymbol{T}(\sigma) = \boldsymbol{T}$. If $\boldsymbol{T}$ extends to level $l+2$ then the module ξ_i and the submodule $\xi_i(0)$ should both be placed on σ' if this string is the leftmost string of level $l+1$ in T compatible with $\sigma(\boldsymbol{T})$.
   (b) Suppose that ξ_i is placed on a string σ of level l in T and has not been declared satisfied. If, for some j, all submodules $\xi_i(j)$

have been declared satisfied and if $\boldsymbol{T}(\sigma)$ extends to level $l+j+2$, then a submodule $\xi_i(j+1)$ should be placed on each string in T of level $l+j+1$ extending σ.

(2) **We begin building a new tree inside the cluster.** If either:
  (a) the module ξ_i is placed on σ of level l in T and has been declared satisfied, or;
  (b) the module ξ_i is placed on a string in T and has not been declared satisfied, and $\xi_i(j)$ placed on σ of level $l-1$ in T is the leftmost submodule of this module not to have been declared satisfied,

then for σ' which is the leftmost string in T of level $l+1$ extending σ, the supermodule υ_{i+1} should be placed on σ' if $\boldsymbol{T}(\sigma)$ extends to level $l+2$.

(3) **We continue to build the simplified construction within an extended cluster.** If the supermodule υ_i is placed on σ of level l in T and if the tree associated with this supermodule extends to level $l+2$ then the module θ_i should be placed on all strings in T of level $l+1$.

(4) **We continue to build the simplified construction within the cluster, after a failed attempt to build another tree within the cluster.** If σ is of level l in T and is the least non-terminal leaf of some $T_{i,k}$, then let $\boldsymbol{T}(\sigma)=\boldsymbol{T}$. Let i' be the least such that no module $\xi_{i'}$ is placed on any string σ' of level $< l$ in T such that $\boldsymbol{T}(\sigma')\subseteq\boldsymbol{T}$. The module $\theta_{i'}$ should be placed on all strings in T of level $l+1$ if $\boldsymbol{T}$ extends to level $l+2$.

Once these instructions are completed, enumerate $\sigma\star 0$ and $\sigma\star 1$ into T for each leaf σ of T.

A is defined to be the infinite string extending all σ on which a supermodule is placed and such that the tree associated with that supermodule is infinite.

The verification. The instructions for the supermodule υ_i implicitly assume that any leaf of its associated tree $T_{i,k}$ which has not been declared terminal, will also be a string in T. That this assumption is correct can easily be seen by induction on stage s using convention C2. It is then clear that our approximation to T is finitely 2-branching at each stage. That our approximation to T converges and is 2-branching follows since:

(1) each module or submodule can only redefine T a finite number of times;
(2) each supermodule can only define any given level of T a finite number of times;
(3) any module, submodule or supermodule can only redefine levels of T strictly above that on which they are placed;
(4) only a finite number of modules, submodules or supermodules are ever placed on any given level of T.

That $\mathcal{P}=[T]$ is a Π^0_1 class follows immediately from the fact that, if at any stage we redefine the strings in T of level l, we define each of these strings to extend leaves of the previous version of T, and we enumerate strings into T properly extending each of its leaves at the end of each stage.

That each $T_{i,k}$ is computable follows from the fact that if $T_{i,k}$ is infinite and if $\sigma \in T_{i,k}$ has successors in this set at any stage of the construction then there exists a stage s such that at this stage:

- σ has successors τ_0 and τ_1;
- each τ_i has either been declared terminal or else has successors in $T_{i,k}$.

Neither of the τ_i will be removed from $T_{i,k}$ subsequent to stage s, and any τ_i which has been declared terminal will remain a leaf of $T_{i,k}$.

Suppose that some module (or some submodule or supermodule) is placed on σ and that T' is the lowest priority tree in $\boldsymbol{T}(\sigma)$ when this module is first passed control, or that $T' = T$ if this set is empty. We call T' the *parent tree* for the module. If a module (or a submodule or supermodule) is placed on some string in T and is never subsequently removed, then we say that the module is *permanent.* The fact that A is properly defined and is hyperimmune-free, and that all other requirements are satisfied as required then follows easily from the discussion prior to the precise description of the construction and from the existence of a sequence of strings $\tau_{\upsilon_0} \subset \tau_{\theta_0} \subset \tau_{\xi_0} \subseteq \cdots \tau_{\upsilon_i} \subset \tau_{\theta_i} \subset \tau_{\xi_i} \subseteq \cdots$ such that the following conditions are satisfied.

(i) Each string in this sequence is a string in the limit value T.

(ii) No permanent supermodule, whose associated tree is infinite, is placed on a string incompatible with any member of the sequence. Any member of the sequence τ, is eventually always compatible with $\sigma(\boldsymbol{T}(\tau))$.

(iii) Either:

(a) A permanent supermodule υ_i is placed on τ_{υ_i} and the tree associated with this supermodule is infinite, or;

(b) τ_{υ_i} is a leaf of a tree $T_{i,k}$ associated with a permanent supermodule, whose parent tree is infinite. Furthermore, τ_{υ_i} is the least leaf of $T_{i,k}$ which is never declared terminal.

(iv) The string τ_{θ_i} is of some level, l say, in the limit value of T. Permanent modules θ_i are placed on all strings in T of level l. The parent tree of the permanent module θ_i placed on τ_{θ_i} is infinite.

(v) Either:

(a) A permanent module ξ_i, with infinite parent tree, is placed on some string σ in T and is eventually declared satisfied. The leftmost successor of σ in the limit value of T is τ_{ξ_i}, or;

(b) A permanent submodule $\xi_i(j)$, with infinite parent tree, is placed on some string σ of level l in T and is never declared satisfied, although all permanent submodules $\xi_i(j)$ placed on strings strictly to the left of σ are declared satisfied. The leftmost string extending σ of level $l+2$ in the limit value of T is τ_{ξ_i}.

In order to see that there exists a sequence of this kind, suppose that we have already defined $\tau_{\upsilon_0} \subset \tau_{\theta_0} \subset \tau_{\xi_0} \subseteq \cdots \tau_{\upsilon_{i-1}} \subset \tau_{\theta_{i-1}} \subset \tau_{\xi_{i-1}}$ satisfying all of the required conditions, or if $i = 0$ suppose that $\tau_{\xi_{-1}} = \lambda$. If $i = 0$ then the instructions for stage 0 of the construction ensure that a permanent supermodule υ_i is placed on $\tau_{\xi_{i-1}}$. If $i > 0$ then either (a) or (b) from (v) above applies to $\tau_{\xi_{i-1}}$. Since the parent tree associated with the permanent module ξ_{i-1} (and with the permanent submodule $\xi_{i-1}(j)$ if (b) applies) is infinite, (2) from the description of the construction will ensure that a permanent supermodule υ_i is placed on $\tau_{\xi_{i-1}}$, which then has infinite parent tree.

First suppose that the tree associated with the permanent supermodule υ_i placed on $\tau_{\xi_{i-1}}$ is infinite. In this case we can define $\tau_{\upsilon_i} = \tau_{\xi_{i-1}}$ and (i), (ii) and (iii) will remain satisfied. Let τ_{υ_i} be of level l in the limit value of T. Since there will be a stage after which the tree associated with υ_i always extends to level $l+2$, we can define τ_{θ_i} to be the leftmost successor of τ_{υ_i} in the limit value of T and (3) from the description of the construction will ensure that (iv) remains satisfied for the induction step. The parent tree for the permanent module θ_i placed on τ_{θ_i} will be the tree associated with υ_i.

Next suppose that the tree associated with the permanent supermodule υ_i placed on $\tau_{\xi_{i-1}}$ is finite. Let $T_{i,k}$ be the tree associated with this supermodule. Then the parent tree of υ_i is infinite, so that if we define τ_{υ_i} to be the least leaf of $T_{i,k}$ which is never declared terminal then (i), (ii) and (iii) will be maintained for the induction step. Let τ_{υ_i} be of level l in the limit value of T. Clause (4) from the description of the construction will place a permanent module θ_i on all strings of level $l+1$ in the limit value of T. We can therefore define τ_{θ_i} to be the leftmost successor of τ_{υ_i} in the limit value of T, maintaining satisfaction of (iv) for the induction step. The parent tree for the permanent module θ_i placed on τ_{θ_i} will be the parent tree of the supermodule υ_i.

Whether or not the tree associated with the permanent supermodule υ_i placed on $\tau_{\xi_{i-1}}$ is finite, then, we have described how to define τ_{θ_i} so that all required conditions are satisfied. Since the parent tree for the permanent module placed on this string is infinite, and by (ii) of the induction hypothesis, clause (1) from the description of the construction will place a permanent module ξ_i and a submodule $\xi_i(0)$ on the leftmost successor of τ_{θ_i} in the limit value of T. If the permanent module ξ_i is eventually declared successful then we can define τ_{ξ_i} to be its leftmost successor in the limit value of T in order to complete the induction step. Otherwise, if there exists a least submodule $\xi_i(j)$ for this module which is placed on a string σ of level l (say) in the limit value of T and which is not declared satisfied, then we can define τ_{ξ_i} to be the leftmost successor of σ of level $l+2$ in the limit value of T. So suppose that there exists no such submodule. In this case it is easily seen, by using an induction on j, that the strings on which the submodules are placed form a perfect Π_i-splitting tree. Note that the modification of the instructions for the simplified construction—that any submodule $\xi_i(j)$ placed on any string σ of any level l in T should not act before the first stage at which it is the least submodule for the corresponding module ξ_i not to be declared satisfied and $\boldsymbol{T}(\sigma)$ extends to level $l+2$—was necessary in order that action taken in order to extend the parent tree of ξ_i does not interfere with the process by which the submodules build splittings. Then the set of strings of the form $\Psi_i(\tau)$ such that τ is in this perfect Ψ_i-splitting tree forms a perfect and computable tree with all paths in $[\Lambda]$, which gives the required contradiction. That this set of strings is computable follows from the fact that no submodule for the permanent module ξ_i will ever be removed from a string in T once it has been placed upon it.

□

Corollary 5.4. *For every $\boldsymbol{\alpha} < \mathbf{1}_{\mathfrak{P}}$ there exists $\boldsymbol{\beta} > \mathbf{0}_{\mathfrak{P}}$ such that $\boldsymbol{\alpha} \wedge \boldsymbol{\beta} = \mathbf{0}_{\mathfrak{P}}$.*

Corollary 5.5. *$(\mathfrak{P}, <)$ has an infinite number of minimal elements.*

Proof. Suppose that $(\mathfrak{P}, <)$ has only a finite number of minimal elements, $\boldsymbol{\alpha}_0, \cdots, \boldsymbol{\alpha}_k$ say. Then $\boldsymbol{\alpha}_0 \cup \cdots \cup \boldsymbol{\alpha}_k$ is an element $\boldsymbol{\alpha}$ of $\mathfrak{P}$ and by Theorem 5.3 there exists $\boldsymbol{\beta}$ which is minimal and such that $\boldsymbol{\beta} \not\leq \boldsymbol{\alpha}$, a contradiction. □

Following Martin and Pour-El [MP] and Cholak, Coles, Downey and Herrmann [CC], define a *Martin-Pour-El class* to be a thin special Π^0_1 class of the form $\{C|C$ separates $A, B\}$ where A, B is a disjoint computably enumerable pair of c.e. sets. One can prove that, given any special Π^0_1 class $\mathcal{P}$, there exists a Martin-Pour-El class $\mathcal{Q}$ such that no member of $\mathcal{Q}$ computes any member of $\mathcal{P}$. This has been noted by Simpson (unpublished). Moreover, it is easy to see that the degree spectrum of a Martin-Pour-El class is minimal in $(\mathfrak{P}, <)$. Combining these two results we get that given any $\boldsymbol{\alpha} \in \mathfrak{P}$ not containing $\mathbf{0}$ there exists $\boldsymbol{\beta} \in \mathfrak{P}$ not containing $\mathbf{0}$ which is minimal and such that $\boldsymbol{\alpha} \cap \boldsymbol{\beta} = \emptyset$. As pointed out Simpson in a private correspondence, and also by the anonymous referee, this suffices in order to give an alternative proof of Corollary 5.5.

Theorem 5.4. *There exists $\boldsymbol{\alpha}$ with no minimal predecessor.*

Proof. Jockusch and Soare [JS2] have constructed a special Π^0_1 class $\mathcal{P}$ in which all elements are Turing incomparable. Let $\boldsymbol{\alpha} = S(\mathcal{P})$ and suppose that $S(\mathcal{P}') \subset \boldsymbol{\alpha}$ for some non-empty Π^0_1 class $\mathcal{P}'$. Then, by the hyperimmune-free basis theorem there exist i, j, A and B such that Ψ_i and Ψ_j are total Turing functionals, $A \in \mathcal{P}$, $B \in \mathcal{P}'$, and such that $\Psi_i(A) = B$ and $\Psi_j(B) = A$. Consider $\mathcal{P}''$ which is the set of all $C \in \mathcal{P}$ such that $\Psi_i(C)$ is in $\mathcal{P}'$ and $\Psi_j(\Psi_i(C)) = C$. Then $\mathcal{P}''$ is a Π^0_1 class and $S(\mathcal{P}'') \subseteq S(\mathcal{P}')$. In order to show that $S(\mathcal{P}'')$ is not minimal, let σ_0 and σ_1 be incomparable strings which each have infinite extensions in $\mathcal{P}''$. The set of all elements in $\mathcal{P}''$ which extend σ_0 is a Π^0_1 class, with degree spectrum a proper subset of $S(\mathcal{P}'')$. □

6. Π^0_1-mates

The existence of invisible degrees serves to highlight the fact that there exist degrees $\boldsymbol{a} \neq \boldsymbol{b}$ such that any Π^0_1 class which contains a member of degree $\boldsymbol{a}$ must also contain a member of degree $\boldsymbol{b}$. In this section we shall further investigate this kind of relationship between degrees.

Definition 6.1. *We define the Π^0_1 mates of $\boldsymbol{a}$ to be $\bigcap\{\boldsymbol{\alpha} \in \mathfrak{P} \ : \ \boldsymbol{a} \in \boldsymbol{\alpha}\}$.*

It is clear that there exist degrees $\boldsymbol{a}$ such that the Π^0_1-mates of $\boldsymbol{a}$ are just $\{\boldsymbol{a}\}$. In order to see that, in fact, there exist continuum many such $\boldsymbol{a}$, consider any non-empty Π^0_1 class $\mathcal{P}$ in which all elements are Turing incomparable. Suppose $A \in \mathcal{P}$ and is of degree $\boldsymbol{a}$. Then the Π^0_1-mates of $\boldsymbol{a}$

must be a subset of $S(\mathcal{P})$, and for each $B \neq A$ in $\mathcal{P}$ there exists a Π^0_1 class $\mathcal{P}' \subset \mathcal{P}$ which contains A but does not contain B.

If C is of hyperimmune-free degree and $B \leq_T C$ then $B \leq_{tt} C$. Cenzer and Smith [CS] have shown that whenever $B \leq_{tt} C$ and C is ranked, B is also ranked and has rank less than or equal to that of C. Downey has constructed [RD] a set A of hyperimmune-free degree $\boldsymbol{a}$ and which is of rank 1, and so is a member of a Π^0_1 class with degree spectrum $\{\mathbf{0}, \boldsymbol{a}\}$. Since $\boldsymbol{a}$ must therefore be completely ranked and a set is ranked iff it is a member of a countable Π^0_1 class, it follows that the Π^0_1-mates of $\boldsymbol{a}$ are $\{\mathbf{0}, \boldsymbol{a}\}$, since if any special Π^0_1 class contained a member B of degree $\boldsymbol{a}$ then its intersection with any countable class with B as an element, would be a countable special Π^0_1 class. By Theorem 5.1, if $\boldsymbol{a}$ is any hyperimmune-free element of $\boldsymbol{\alpha}$ which is minimal, then the Π^0_1-mates of $\boldsymbol{a}$ are $\boldsymbol{\alpha}$. So far then, we have seen $\boldsymbol{a}$ for which the Π^0_1-mates of $\boldsymbol{a}$ are of cardinalities 1 and 2, and also $\boldsymbol{a}$ for which the Π^0_1-mates of $\boldsymbol{a}$ are uncountable. The following theorem, however, provides a wealth of further examples.

Theorem 6.1. *For any $\boldsymbol{\alpha} < \mathbf{1}_{\mathfrak{P}}$ there exists $\boldsymbol{a} \notin \boldsymbol{\alpha}$ such that the Π^0_1-mates of $\boldsymbol{a}$ are $\boldsymbol{\alpha} \cup \{\boldsymbol{a}\}$ and, moreover, such that $\boldsymbol{\alpha} \cup \{\boldsymbol{a}\} \in \mathfrak{P}$.*

Proof. We write $\sigma <_L \sigma'$ if there exists some least n such that $\sigma(n) \downarrow\neq \sigma'(n)$ and if, for this n, $\sigma(n) = 0$. We call a (possibly partial) function $\kappa : 2^{<\omega} \rightarrow 2^{<\omega}$ a *tree function* if both of the following conditions are satisfied:

(i) Whenever $\sigma \subset \sigma'$, if $\kappa(\sigma') \downarrow$ then $\kappa(\sigma) \downarrow$ and $\kappa(\sigma) \subset \kappa(\sigma')$;

(ii) If $\sigma <_L \sigma'$, $\kappa(\sigma) \downarrow$ and $\kappa(\sigma') \downarrow$, then $\kappa(\sigma) <_L \kappa(\sigma')$.

If Λ is a set of finite binary strings we shall write $\kappa(\Lambda)$ in order to denote $\{\kappa(\sigma) : \sigma \in \Lambda \ \&\ \kappa(\sigma) \downarrow\}$. We shall say that $\mathcal{Q}$ is a *similar copy* of $\mathcal{P}$ if there exists a partial computable tree function κ and downward closed computable Λ such that $\kappa(\sigma) \downarrow$ for all $\sigma \in \Lambda$, $\mathcal{P} = [\Lambda]$ and $\mathcal{Q} = [\kappa(\Lambda)]$.

Given downward closed computable Λ such that $\mathcal{P} = [\Lambda]$ does not contain a member of every Turing degree, we construct downward closed computable Υ and $A \in [\Upsilon]$ of degree $\boldsymbol{a}$ such that $S([\Upsilon]) = S(\mathcal{P}) \cup \{\boldsymbol{a}\}$ and such that the following requirements are satisfied:

Θ_i: Either $\Psi_i(A)$ is partial or computable, or $\Psi_i(A) \notin \mathcal{P}$.

Ξ_i: Let $i = \langle j, k\rangle$. Either $\Psi_j(A)$ is partial or computable or $\Psi_j(A) \notin [\Lambda_k]$, or else $[\Lambda_k]$ has a subset which is a similar copy of $\mathcal{P}$.

Having defined a construction that satisfies all of these requirements we shall then be able to make an application of Posner's lemma for minimal degree constructions, in order to show that A is not computable (so that the satisfaction of these requirements suffices to give the theorem).

Once again, the proof is most easily described using modules. Essentially the proof is a synthesis of the techniques used in the proof of Theorem 5.3 and also those techniques used in the proof of Theorem 4.6. As in the proof of Theorem 5.3, we must deal with the difficulty that in order to satisfy the requirements we must search for splittings inside Π^0_1 classes, and the solution is once again to ensure that A is of hyperimmune-free degree. As before, the approach we take is firstly to describe modules for a simplified version of the construction which acts in order to satisfy all requirements and which succeeds for those corresponding to functionals which are total, but which does not act in order to ensure that A is of hyperimmune-free degree. We shall then have to modify these simplified modules and fit them into the framework for constructing A of hyperimmune-free degree, but this should now seem straightforward given that it can be done in the same way as in the proof of Theorem 5.3. Now, in fact, things are much simpler, since we needn't worry about maintaining homogeneity.

The module θ_i (simplified version). We act to satisfy the requirement Θ_i in almost exactly the same way that we acted in order to satisfy the requirement Ξ_i in the proof of Theorem 5.3, except that now matters are considerably simpler since no homogeneity condition need be satisfied, so that there is no need to delegate work to submodules. It will be the role of the module θ_i placed on σ to ensure that Θ_i is satisfied if $\sigma \subset A$, and the module will construct a (finite) splitting tree $T(i,\sigma)$.

At every stage at which it is passed control after being placed on σ, the module carries out one of the instructions below according to the first case which applies. $T(i,\sigma)$ is initially defined to be $\{\sigma\}$.

(1) The module has already been declared successful. This means that we will already have found a string σ' such that $\Psi_i(\sigma') \notin \Lambda$ and we will have placed the *final successor module* on this string. Pass control to this module.

(2) If there exists $\sigma' \supset \sigma$ which is a leaf of Υ which has not been declared terminal, and such that $\Psi_i(\sigma') \notin \Lambda$, then we declare the module to be successful. In this case proceed as follows. Remove all modules from strings properly extending σ, remove any module which θ_i has placed on σ, and declare all leaves of Υ extending σ to be terminal other than σ'. Place the module ξ_{i+1} on σ' and define this to be the final successor module. Enumerate the pair (σ,σ').

    (The enumeration of this pair (σ,σ') is just something that will be technically convenient later on when describing a new version of the convention C2 which we used in the proof of Theorem 5.3. The point is that in that proof, whenever a module acted, it would enumerate strings into T. Here this does not happen, so we need some other way of recording action which we may need to accommodate by adjusting the leaves of trees.)

(3) Let σ' be the least leaf of the present $T(i,\sigma)$ (ordered, as for all trees in the construction, according to their level in the tree and then from left to right). If there exist σ_0,σ_1 extending σ', which are leaves of Υ which have not been declared terminal and which are Ψ_i-splitting, then proceed as follows. Remove all modules (but not θ_i itself) from strings extending σ', declare all leaves of Υ extending σ' other than σ_0 and σ_1 to be terminal and enumerate σ_0 and σ_1 into $T(i,\sigma)$. Enumerate the pairs (σ',σ_0) and (σ',σ_1).

(4) If cases (1),(2) and (3) do not apply then let σ' be defined as in (3). Pass control to the module ξ_{i+1} placed on σ', or place this module on this string if it has not already been placed there.

The module ξ_i (simplified version). It is the role of the module ξ_i placed on σ to ensure that Ξ_i is satisfied if $\sigma \subset A$. This module will act in a way which is a mix between the module θ_i described above and the module ξ_i of Theorem 4.6. The basic idea is that the module will try to place a similar copy of $\mathcal{P}$ above $\sigma \star 0$, in the form of a Ψ_j-splitting set of strings $\kappa_{i,\sigma}(\Lambda)$ for some partial computable tree function $\kappa_{i,\sigma}$ such that $\kappa_{i,\sigma}(\sigma') \downarrow$ for all $\sigma' \in \Lambda$. If it finds some string $\tau = \kappa_{i,\sigma}(\sigma')$ above which it is unable to find a Ψ_j-splitting then putting this string as an initial segment of A ensures that $\Psi_j(A)$ is either partial or computable. If, on the other hand, it finds some string τ in $\kappa_{i,\sigma}(\Lambda)$ such that $\Psi_j(\tau) \notin \Lambda_k$ then it can satisfy the requirement by insisting that this string should be an initial segment of A. If neither of these possibilities hold then $[\Lambda_k]$ has a subset which is a similar copy of $\mathcal{P}$. In this case the requirement is automatically satisfied and $\sigma \star 1$ will be an initial segment of A.

At every stage at which it is passed control the module carries out one of the instructions below according to the first case which applies. When the module is first passed control $\kappa_{i,\sigma}(\lambda)$ is already defined to be $\sigma \star 0$.

(1) The module has already been declared successful. This means that we will already have found σ' such that $\Psi_j(\sigma') \notin \Lambda_k$ and placed a final successor module on this string. Pass control to this module. The module ξ_i has outcome 0 at this stage.

(2) If there exists $\sigma' \supset \sigma \star 0$ which is a leaf of the present Υ which has not been declared terminal and such that $\Psi_j(\sigma') \notin \Lambda_k$ then we declare the module to be successful. In this case proceed as follows. Remove all modules from strings properly extending σ. Declare all leaves of the present Υ extending σ except σ' to be terminal. Place the module θ_i on σ' and define this to be the final successor module. Enumerate the pair (σ, σ'). The module ξ_i has outcome 0 at this stage.

(3) Let σ' be the least leaf which has not been declared terminal in $\kappa_{i,\sigma}(\Lambda)$. If there exist σ_0, σ_1 extending σ', which are leaves of the present Υ which have not been declared terminal (in Υ) and which are Ψ_j-splitting, then proceed as follows. Let σ_0 be to the left of σ_1. Remove all modules from strings extending σ'. Let τ be such that $\sigma' = \kappa_{i,\sigma}(\tau)$ and if τ has no proper extension in Λ then declare σ' to be terminal in $\kappa_{i,\sigma}(\Lambda)$. Otherwise, for each $d \in \{0,1\}$ such that $\tau \star d \in \Lambda$, define $\kappa_{i,\sigma}(\tau \star d) = \sigma_d$ and enumerate the pair (σ', σ_d). Declare all leaves of the present Υ extending σ' except those strings which we have just enumerated into $\kappa_{i,\sigma}(\Lambda)$ to be terminal.

    In this case the module has outcome 1 at this stage and it passes control to the module θ_i placed on $\sigma \star 1$ (or places this module on this string if it has not previously done so).

(4) If cases (1), (2) and (3) do not apply then the module proceeds as follows. Let σ' be defined as in (3). The module passes control to the module θ_i placed on σ' (or places this module on this string if it has not previously done so). The module has outcome 0 at this stage.

Just as in the proof of Theorem 4.6, then, the module either has outcome 0 or 1 at each stage at which it is passed control. If it is passed control at an infinite number of stages and has outcome 1 then it places a similar copy of $\mathcal{P}$ above $\sigma \star 0$ in the form of a Ψ_j-splitting tree. Since all paths of the image of this tree under Ψ_j are also elements of $[\Lambda_k]$, this means that $[\Lambda_k]$ contains a similar copy of $\mathcal{P}$. In this case $\sigma \star 1$ will be an initial segment of A. If the module is passed control at an infinite number of stages and has outcome 0 at all but finitely many of these stages then $\sigma \star 0$ will be an initial segment of A. In this case we are either able to ensure that $\Psi_j(A)$ is partial or computable, or else we are able to ensure that $\Psi_j(A)$ is not in $[\Lambda_k]$.

Our only remaining significant task is to fit modified versions of these modules into the framework for constructing A of hyperimmune-free degree. Since this can be done in the same way as in the proof of Theorem 5.3, this should now seem straightforward. There is, however, one small further point which must be considered. In the proof of Theorem 4.6, it was necessary sometimes to keep the construction 'alive' above strings where the construction was not presently taking place but which it might be necessary to work above later on—above the string $\sigma \star 1$, for example, when the module θ_i was placed on σ and had not yet observed $\sigma \star 0$ to be an initial segment of $\Psi_i(\emptyset)$. In that construction, we could simply continue to concatenate these strings which we might later wish to work above with zeros in order to keep some extension in the Π^0_1 class. Now, however, we must ensure that $S([\Upsilon]) = S(\mathcal{P}) \cup \{\boldsymbol{a}\}$, and $S(\mathcal{P})$ may not contain $\mathbf{0}$. In order to deal with this difficulty, copies of Λ, rather than sequences of zeros, must be concatenated above strings where the construction is not presently taking place, but to which it may later be necessary to return. This is basically easy but a little bit fiddly. To this end it is convenient to assume that $\mathcal{P}$ has elements which extend both 0 and 1. The following paragraph describes precisely the procedure by which we concatenate copies of Λ.

A module υ_i will enumerate some tree $T_{i,k}$ (in this construction we shall consider only modules rather than supermodules, modules and submodules), a module ξ_i placed on σ will construct a computable tree $\kappa_{i,\sigma}(\Lambda)$ and a module θ_i placed on σ will construct some finite Ψ_i-splitting tree $T(i,\sigma)$. Each of these is called the tree *associated* with the corresponding module. Suppose that at some stage of the construction σ' is the least leaf of $T_{i,k}$ which has not been declared terminal, where $T_{i,k}$ is the tree associated with the module υ_i. Suppose further that σ' is of level l in this tree, we find τ which is a leaf of Υ extending σ' which has not been declared terminal such that $\Psi_i(\tau; l) \downarrow$ and we are given the instruction to *add appropriate extensions* to $T_{i,k}$. In this case we proceed as follows. It is assumed that whenever we refer to a module or its associated tree, this is a module which has been placed on a string in Υ and not removed. Remove any module which υ_i has placed on σ' (unless, and this is just a small technical point, σ' is the leftmost string in $T_{i,k}$ of level 1, in which case any module ξ_i on this string is initialized rather than removed, and now has parent tree $T_{i,k}$ according to the definition in C2 below). Let $\sigma \subseteq \sigma'$ be the longest such that either:

- σ is a leaf of some tree associated with a module, other than $T_{i,k}$, or;
- there is a module placed on σ.

Let κ be the finite tree function whose image is the set of strings in $T_{i,k}$ extending σ and let τ' be such that $\kappa(\tau') = \sigma'$. The domain of κ will be an initial segment of Λ. Remove all modules from strings properly extending σ'. If τ' has no proper extension in Λ then declare σ' to be terminal in $T_{i,k}$. Otherwise, for each $d \in \{0,1\}$ such that $\tau' \star d \in \Lambda$, enumerate $\tau \star d$ into $T_{i,k}$ and Υ (where τ is the leaf of Υ as above) and enumerate the pair $(\sigma', \tau \star d)$. Declare all leaves of the present Υ extending σ' except those strings which we have just enumerated into $T_{i,k}$ to be terminal.

In altering the construction we fix the following conventions and modifications, all of which are necessary for an accurate proof but some of which only constitute small technical details of the kind which can reasonably be ignored upon a first reading.

C1 When any module is removed, we remove also any modules placed on strings by that module.

C2 Modules of the form υ_i will pass control to modules of the form ξ_i which will pass control to modules of the form θ_i, which in turn will pass control to modules of the form υ_{i+1} (as opposed to modules ξ_{i+1} as in the simplified construction), and so on. Any mention of the module ξ_{i+1} in the instructions for the module θ_i, then, must now be replaced with υ_{i+1}. The *parent tree* of any module is defined basically just as in the proof of Theorem 5.3. Small technical differences, however, mean that we now need to define this term inductively. We define the parent tree for υ_0 placed on λ to be Υ. Any module has the same parent tree as that which passes control to it, except for a module ξ_i which is passed control by υ_i and is placed on a string which is not a leaf of the tree associated with υ_i. Such a module ξ_i has the tree associated with υ_i as its parent tree.

C3 No module placed on a string σ is passed control at a stage at which the parent tree for the module does not yet contain any strings properly extending σ (so that upon receiving any instruction to do so at an earlier stage we simply terminate activity for that stage). No module will enumerate a string into its associated tree extending a leaf of that tree σ until a stage at which the parent tree for the module contains strings properly extending σ (with the same proviso for earlier stages at which instructions violating this condition are given).

C4 If σ is a leaf of $T_{i,k}$ and all leaves of Υ extending σ are declared terminal in Υ, then σ is declared terminal in $T_{i,k}$.

C5 When the module ξ_i placed on σ is first passed control we remove all modules from strings properly extending σ, we choose $\tau_0 <_L \tau_1$ to be two leaves of Υ extending σ which have not been declared terminal and we enumerate the pairs (σ, τ_0) and (σ, τ_1). We place the module θ_i on τ_1 and define $\kappa_{i,\sigma}(\lambda) = \tau_0$. At all subsequent stages at which the module is passed control we then carry out the first of the instructions (1)–(4) as previously described (although subject to the other conventions and modifications listed here), but with τ_0 and τ_1 taking the place of $\sigma \star 0$ and $\sigma \star 1$ respectively.

C6 This convention is essentially just a version of C2 from the proof of Theorem 5.3, but modified for the present construction.

(a) Suppose that at some stage of the construction a module enumerates two pairs (σ, σ_0) and (σ, σ_1). Suppose further that $\sigma' \supseteq \sigma$ is in $T_{i,k}$ and that τ is a leaf of $T_{i,k}$ and a successor of σ'. If both of σ_0 and σ_1 extend τ then we remove all proper extensions of σ' from $T_{i,k}$ and enumerate in σ_0 and σ_1. If precisely one of σ_0 and σ_1 extend τ, then we replace τ in $T_{i,k}$ with that string.

(b) Suppose that at some stage of the construction a module enumerates a single pair (σ, σ_0). Suppose further that $\sigma' \supseteq \sigma$ is in $T_{i,k}$ and that τ is a leaf of $T_{i,k}$ and a successor of σ'. If σ_0 properly extends τ, then we replace τ in $T_{i,k}$ with σ_0.

The module υ_i. When placed on the string σ the module υ_i will find the least k such that $T_{i,k}$ is not yet associated with any module, and will enumerate the string σ into $T_{i,k}$ (in other words it will choose a fresh tree to begin enumerating). Let T' be the parent tree for υ_i. As before, T' is the tree within which the module will try to build $T_{i,k}$.

At each stage strictly after that at which it is placed on σ and at which it is passed control, the module will proceed as follows. First we look to see whether we can further build $T_{i,k}$. Let σ' be the least leaf of $T_{i,k}$ which has not been declared terminal and let σ' be of level l in this tree. If there exist strings in T' properly extending σ' then check to see if there exists τ which is a leaf of Υ extending σ' which has not been declared terminal and such that $\Psi_i(\tau; l)$ is defined.

If so: then *add appropriate extensions* to $T_{i,k}$ as described previously. In this case, since $T_{i,k}$ presently appears to be infinite we now try to make progress with the simplified construction inside $T_{i,k}$. Pass control to the module ξ_i placed on the leftmost string in T of level 1, or place this module on that string if it it not already placed there.

If not: then we must pass control to a module which works on the assumption that $T_{i,k}$ will be finite. Let σ' be defined as above. If the module ξ_i has already been placed on σ' then pass control to this module, providing this does not violate convention C3. Otherwise place ξ_i on σ'.

The construction. At stage 0 we enumerate $\lambda, 0$ and 1 into Υ and place the module υ_0 on λ. At all subsequent stages we proceed as follows.

(1) Pass control to the module υ_0 placed on λ and then proceed to pass control to other modules as directed until some module is passed control and does not pass control to another.

(2) For each leaf σ of Υ which has not been declared terminal proceed as follows. Let τ be the longest initial segment of σ which is a

leaf of some tree associated with a module or which has a module placed upon it. We must continue to place a copy of Λ above τ. Let $\sigma = \tau \star \sigma'$. Follow the instructions for the first case below which applies.

(a) If σ' has no proper extensions in Λ then declare σ to be terminal.
(b) For each $d \in \{0,1\}$ with $\sigma' \star d \in \Lambda$, enumerate $\sigma \star d$ into Υ.

A is defined to be the infinite string extending all σ on which a module υ_i is placed and such that the tree associated with that module is infinite.

Some final details for verification. The modules θ_i and ξ_i function in ways which are extremely similar to modules described in previous constructions, and the way in which they operate should already be completely clear from the preceding discussion. The framework for constructing A of hyperimmune-free degree is basically just the same as that used in the proof of Theorem 5.3. It remains, however, to verify aspects of the construction which are significantly different in some respect to previous constructions.

In order to show that A is not computable we make an application of Posner's lemma for minimal degree constructions. So suppose that A is computable. Then there exists i such that $A = \Psi_i(\emptyset)$. Consider Ψ_j such that, for any σ, $\Psi_j(\sigma) = \lambda$ if $\sigma \subset \Psi_i(\emptyset)$ and $\Psi_j(\sigma) = \sigma$ otherwise. Let τ be an initial segment of A upon which a module θ_j with infinite parent tree is placed at some stage of the construction and never removed. Since there are more than two strings in $[\Upsilon]$ extending τ, θ_i will eventually enumerate strings into $T(i,\tau)$ properly extending τ. These strings will all be incompatible with A, but A must extend one of them. This gives the required contradiction.

We must also show that every element of $[\Upsilon]$ other than A is of the same degree as an element of $\mathcal{P}$. So suppose that B is an element of $[\Upsilon]$ other than A. There are three possibilities to consider.

Case 1 There exists a module υ_i with infinite parent tree, which is placed on an initial segment of A and never removed, but whose associated tree $T_{i,k}$ is finite. A extends the least leaf of $T_{i,k}$ which is never declared terminal, but B extends another leaf of this tree which is not declared terminal, σ say.

Then the set of infinite paths through the parent tree of υ_i extending σ is a similar copy of $\mathcal{P}$. B must be one of these paths.

Case 2 There exists a module ξ_i with infinite parent tree which is placed on an initial segment of A, σ say, and never removed. Let σ' be the string upon which a module θ_i is placed and which the module ξ_i passes control to when it has outcome 1. We subdivide into three further cases.

(a) A extends σ', while B does not.
In this case B must be an infinite path through $\kappa_{i,\sigma}$.

(b) B extends σ' while A does not.
In this case the module only has outcome 1 at a finite number of stages. Any infinite path through $[\Upsilon]$ extending σ' must be consist of some finite initial segment τ above which the set of paths through the parent tree of ξ_i consists of a similar copy of

$\mathcal{P}$, concatenated with a path through the parent tree above this string.

(c) A and B both extend different elements of $\kappa_{i,\sigma}(\Lambda)$, which is finite. Let σ'' be the element of $\kappa_{i,\sigma}(\Lambda)$ which B extends. Then the set of infinite paths through the parent tree of ξ_i extending σ'' is a similar copy of $\mathcal{P}$. B must be one of these paths.

Case 3 There exists a module θ_i with infinite parent tree which is placed on σ, which is an initial segment of A, and is never removed. This module is never declared successful. A extends the least leaf of $T(i,\sigma)$ while B extends another leaf, σ' say.

Once again, the set of infinite paths through the parent tree of θ_i extending σ' is a similar copy of $\mathcal{P}$. B must be one of these paths.

□

We shall say that $\boldsymbol{\beta}$ is a *strong minimal cover* for $\boldsymbol{\alpha}$ if the elements of $\mathfrak{P}$ strictly below $\boldsymbol{\beta}$ are precisely those below and including $\boldsymbol{\alpha}$.

Corollary 6.1. *Every $\boldsymbol{\alpha} < \mathbf{1}_{\mathfrak{P}}$ has a strong minimal cover.*

Proof. By Theorem 6.1, let $\boldsymbol{a} \notin \alpha$ be such that the Π^0_1-mates of $\boldsymbol{a}$ are $\alpha \cup \{\boldsymbol{a}\}$. Let $\mathcal{P}$ be a Π^0_1 class such that $S(\mathcal{P}) \subseteq \alpha \cup \{\boldsymbol{a}\}$. If $\boldsymbol{a} \in S(\mathcal{P})$ then clearly $S(\mathcal{P}) = \alpha \cup \{\boldsymbol{a}\}$, and otherwise $S(\mathcal{P}) \subseteq \alpha$. □

Corollary 6.1 suffices to show that there do not exist any maximal $\boldsymbol{\alpha} < \mathbf{1}_{\mathfrak{P}}$. We observed already that there are no maximal elements amongst the degree spectra of special Π^0_1 classes. If the Π^0_1 class $\mathcal{P}$ is countable with rank β then there exists a computable ordinal $\alpha > \beta$ and all members of $\mathcal{P}$ have degree $< \mathbf{0}^{(2\alpha)}$. Since there exists a ranked set of degree $\mathbf{0}^{(2\alpha)}$ [CCS] and a set is ranked iff it is a member of a countable Π^0_1 class, this suffices to show that there are no maximal elements amongst the countable members of $\mathfrak{P}$.

Recall that a Π^0_1 class is *rank faithful* if the rank of any member of the class relative to the class is its actual rank. Theorem 6.1 also gives an interesting corollary regarding sets which are not a member of any rank faithful class:

Corollary 6.2. *There exists a non-computable set A which is ranked, such that every $B \leq_T A$ is ranked, and such that no non-computable $B \leq_T A$ is a member of any rank faithful class.*

Proof. Let $\mathcal{P}$ be any countable Π^0_1 class which is not rank faithful. The construction described in the proof of Theorem 6.1 gives Υ which is countable and $A \in [\Upsilon]$ which is of non-zero hyperimmune-free degree. Every set Turing reducible to A is therefore ranked [CS]. No non-computable set Turing reducible to A is a member of any Π^0_1 class unless it contains a similar copy of $\mathcal{P}$ and is therefore not rank faithful. □

7. Two anti-basis theorems

In this section we shall describe two anti-basis theorems for Π^0_1 classes. Before doing so, however, let us describe our initial motivation for proving the first of these two theorems. From this point onwards we will allow the variables $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ to range over sets of degrees which are not necessarily elements of $\mathfrak{P}$.

Definition 7.1. *We say that $\boldsymbol{\alpha}$ is a* sufficiency set *for $\boldsymbol{a}$ if every Π^0_1 class that contains a member of every degree in $\boldsymbol{\alpha}$ also contains a member of degree $\boldsymbol{a}$.*

In the previous section we considered the Π^0_1 mates of individual degrees. The previous definition could also be phrased in terms of the Π^0_1 mates of larger sets of degrees; if we define the Π^0_1 mates of $\boldsymbol{\alpha}$ (which is not necessarily an element of $\mathfrak{P}$) to be $\bigcap\{\boldsymbol{\beta} \in \mathfrak{P} \ : \ \boldsymbol{\alpha} \subseteq \boldsymbol{\beta}\}$, then $\boldsymbol{\alpha}$ is a sufficiency set for precisely those degrees which are an element of the Π^0_1 mates of $\boldsymbol{\alpha}$.

If $\boldsymbol{\alpha}$ is a sufficiency set for $\boldsymbol{a}$ then there exists some countable $\boldsymbol{\beta} \subseteq \boldsymbol{\alpha}$ which is a sufficiency set for $\boldsymbol{a}$ (for each element of $\mathfrak{P}$ which does not contain every element of $\boldsymbol{\alpha}$ choose some element of $\boldsymbol{\alpha}$ not in this set). For every countably infinite sufficiency set $\boldsymbol{\alpha}$ for $\boldsymbol{a}$ there exists some proper subset which is also a sufficiency set for $\boldsymbol{a}$. In order to see this let $\boldsymbol{a}_0$ and $\boldsymbol{a}_1$ be distinct elements of $\boldsymbol{\alpha}$. If each $\boldsymbol{\alpha} - \{\boldsymbol{a}_i\}$ is not a sufficiency set for $\boldsymbol{a}$ then let each $\mathcal{P}_i$ be a Π^0_1 class which contains a member of every degree in $\boldsymbol{\alpha} - \{\boldsymbol{a}_i\}$ but does not contain a member of degree $\boldsymbol{a}$. Then $\mathcal{P}_0 \cup \mathcal{P}_1$ witnesses the fact that $\boldsymbol{\alpha}$ is not a sufficiency set for $\boldsymbol{a}$, a contradiction. Almost exactly the same argument suffices to show that if $\boldsymbol{\alpha}$ is a finite sufficiency set for $\boldsymbol{a}$ then there exists some element $\boldsymbol{b}$ of $\boldsymbol{\alpha}$ such that $\{\boldsymbol{b}\}$ is a sufficiency set for $\boldsymbol{a}$. The following theorem shows, however, that it is possible to find $\boldsymbol{a}$ and a countable $\boldsymbol{\alpha}$ which is a sufficiency set for $\boldsymbol{a}$, such that no finite subset of $\boldsymbol{\alpha}$ is a sufficiency set for $\boldsymbol{a}$.

Theorem 7.1 (Low Anti-basis Theorem)**.** *Any Π^0_1 class that contains a member of every low degree contains a member of every degree.*

Proof. Let j be such that $[\Lambda_j]$ does not contain a member of every degree and for every i, τ let $\sigma(i, j, \tau)$ be defined as in the proof of Theorem 4.5. We define non-computable $A = \bigcup_i \sigma_i$ which is of low degree and such that for each i, if $\Psi_i(A)$ is total and non-computable then it is not an element of $[\Lambda_j]$. The fact that A is of low degree follows because we run the construction using an oracle for $\emptyset'$ and decide whether $\Psi_i(A; i) \downarrow$ at each stage $2i + 2$.

Stage 0. Define $\sigma_0 = \lambda$.

Stage $2i + 1$. Define $\sigma_{2i+1} = \sigma(i, j, \sigma_{2i})$.

Stage $2i + 2$. If there exists $\sigma \supset \sigma_{2i+1}$ such that $\Psi_i(\sigma; i) \downarrow$ then define σ_{2i+2} to be some extension of σ which is not an initial segment of $\Psi_i(\emptyset)$. Otherwise define σ_{2i+2} to be some extension of σ_{2i+1} which is not an initial segment of $\Psi_i(\emptyset)$.

□

Now if $\boldsymbol{a}$ is not low then $\boldsymbol{\alpha}$ which is the set of all low degrees is a sufficiency set for $\boldsymbol{a}$, such that no finite subset is a sufficiency set for $\boldsymbol{a}$ (since if $\boldsymbol{a}$ is low then $\{\boldsymbol{0}, \boldsymbol{a}\} \in \mathfrak{P}$).

A modified version of the proof of theorem 7.1 can also be made to work below any degree which is non-GL$_2$:

Theorem 7.2. *If $\boldsymbol{b}$ is non-GL$_2$ then, for any Π^0_1 class $\mathcal{P}$ which does not contain a member of every degree, there exists non-zero $\boldsymbol{a} \leq \boldsymbol{b}$ such that $\mathcal{P}$ does not contain any member of non-zero degree below $\boldsymbol{a}$.*

Proof. (Sketch) We simply use the fact that, since $\boldsymbol{b}$ is non-GL$_2$, for every function f of degree below $\mathbf{0}'$, there exists some function g of degree $\boldsymbol{b}$ which is not dominated by f.

Suppose given $\boldsymbol{b}$ which is non-GL$_2$ and j such that $[\Lambda_j]$ does not contain a member of every degree. Suppose that in the proof of theorem 7.1 we had not been concerned with ensuring that A is of low degree (but just of degree below $\mathbf{0}'$), so that we acted at stages $2i+1$ in order to satisfy requirement:

Θ_{2i+1}: $\Psi_i(A)$ is partial or computable, or is not in $[\Lambda_j]$,

and we acted at even stages $2i+2$ in order to satisfy requirement:

Θ_{2i+2}: $A \neq \Psi_i(\emptyset)$.

Rather than using an oracle for $\emptyset'$ directly we could have used some function f to bound the number of steps for which we need to perform searches at each stage i of the construction in order to define σ_i. The function f, then, would be used to bound the number of steps we search for in order to compute $\Psi_i(\emptyset)$, or in order to determine whether any string is a leaf of T as in the definition of $\sigma(i, j, \sigma_{2i})$ from the proof of Theorem 4.5. Let f_0 be a function of degree below $\mathbf{0}'$ which grows sufficiently quickly that if f_0 is used to bound the searches at each stage then each σ_i will be defined correctly. Let f_1 be a function of degree below $\mathbf{0}'$ which grows sufficiently quickly that if f_1 is used to bound the searches at any particular stage i, then σ_i will be defined so as to satisfy Θ_i, no matter how many steps we have searched for at previous stages i' in order to define $\sigma_{i'}$. Finally, consider a modified construction which looks to ensure satisfaction of all requirements $\Theta_{i'}$ such that $i' \leq i$ at any stage i, and let f_2 be a function of degree below $\mathbf{0}'$ such that if this modified construction searches for $f_2(i)$ many steps at any particular stage i, then this is sufficient to satisfy all requirements of priority higher than Θ_i, no matter how many steps we have searched for at previous stages i' in order to define $\sigma_{i'}$. Let g be a function of degree $\boldsymbol{b}$ which is not dominated by f_2. If the function g is used in order to bound the searches at each stage of the construction then all requirements Θ_i will eventually be satisfied.

□

8. $(\mathfrak{P}, <)$ IS A LATTICE

We noted in the introduction that the intersection of the degree spectra of two Π^0_1 classes need not be the degree spectrum of a Π^0_1 class. The following result suffices to show, however, that $(\mathfrak{P}, <)$ is a lattice.

Theorem 8.1. *The intersection of the degree spectra of two Π^0_1 classes is the degree spectrum of a Π^0_1 class if it is the superset of the degree spectrum of a non-empty Π^0_1 class.*

Proof. Let $\mathcal{P}_0$ and $\mathcal{P}_1$ be Π^0_1 classes with degree spectra $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ respectively, and suppose that $\mathcal{P}_2$ is a non-empty Π^0_1 class with degree spectrum $\boldsymbol{\gamma} \subseteq \boldsymbol{\alpha} \cap \boldsymbol{\beta}$.

For each $i \leq 2$ let Λ_i be a downward closed and computable set of strings with $[\Lambda_i] = \mathcal{P}_i$. For each (i, j) we define:

$$\mathcal{Q}_{i,j} = \{A \in \mathcal{P}_0 : \ \Psi_i(A) \in \mathcal{P}_1 \text{ and } \Psi_j(\Psi_i(A)) = A\}.$$

In order to define a downward closed and computable set of strings Λ such that $\mathcal{P} = [\Lambda]$ satisfies $S(\mathcal{P}) = \boldsymbol{\alpha} \cap \boldsymbol{\beta}$, we proceed as follows. We begin by putting all strings in Λ_2 into Λ. Let us say $\sigma \notin \Lambda_2$ is terminal in Λ_2, if all proper initial segments of σ are in Λ_2. We consider strings to be ordered first by length and then from left to right. All strings which are terminal in Λ_2 we place in Λ and above the x^{th} terminal string we place a Π^0_1 class $\mathcal{P}_{i,j}$ with degree spectrum $S(\mathcal{Q}_{i,j}) \cup \boldsymbol{\gamma}$, where $x = \langle i, j \rangle$. It remains only to define the classes $\mathcal{P}_{i,j}$ and in order to do so we define downward closed and computable sets of strings $\Lambda_{i,j}$.

The rough idea behind the construction of $\Lambda_{i,j}$ is that we shall put strings into this set which code pairs of strings, one from each of Λ_0 and Λ_1, and which compute each other via Ψ_i and Ψ_j. While we wait to find convergent computations we place copies of Λ_2 to fill the time. We shall actually construct $\Lambda_{i,j}$ as a subset of $6^{<\omega}$ (the set of all finite strings of elements of $\{0, 1, 2, 3, 4, 5\}$) whose downward closure is computable (while $\Lambda_{i,j}$ itself is not necessarily downward closed). Of course it is easy then to define $\Lambda'_{i,j}$ from $\Lambda_{i,j}$, which is a downward closed and computable set of finite binary strings and which has the same degree spectrum.

Before describing the construction of $\Lambda_{i,j}$ precisely, some simple definitions are required. For any $i, n \in \omega$ and any σ we assume that $\Psi_i(\sigma; n)$ is only defined if this computation converges in less than $|\sigma|$ many steps, $n < |\sigma| - 1$, and only if $\Psi_i(\sigma; n')$ is defined for all $n' < n$. Thus if $\Psi_i(\sigma) = \sigma'$ then $|\sigma| > |\sigma'|$ unless $|\sigma'| = 0$. For any $\sigma \in 2^{<\omega}$ we define $g_0(\sigma)$ to be the string obtained from σ by replacing each 0 with a 2 and replacing each 1 with a 3. We define $g_1(\sigma)$ to be the string obtained from σ by replacing each 0 with a 4 and replacing each 1 with a 5. Suppose given $\psi \in 6^{<\omega}$ and let ψ be written as $\sigma_0 * \tau_0 * \phi_0 * \cdots * \sigma_k * \tau_k * \phi_k$ where each $\sigma_{k'}$ is a string of elements of $\{0, 1\}$, each $\tau_{k'}$ is a string of elements of $\{2, 3\}$ and each $\phi_{k'}$ is a string of elements of $\{4, 5\}$. We define $f_0(\psi) = g_0^{-1}(\tau_0 * \cdots * \tau_k)$ and $f_1(\psi) = g_1^{-1}(\phi_0 * \cdots * \phi_k)$. We shall think of ψ as coding a string in Λ_0 via f_0, and coding a string in Λ_1 via f_1. For any $\psi \in 6^{<\omega}$ we define the *coding part* of ψ to be the shortest $\psi' \subseteq \psi$ such that $\psi(n) < 2$ for all n with $|\psi'| \leq n < |\psi|$. We say that $\psi \in 6^{<\omega}$ *requires* $\mathcal{P}_0$ *extension* if $|f_0(\psi)| \leq |f_1(\psi)|$, and otherwise we say that ψ requires $\mathcal{P}_1$ extension.

Stage 0. For each $\tau \in \Lambda_0$ of length 1, enumerate $g_0(\tau)$ into $\Lambda_{i,j}$.
Stage $s + 1$. For every string ψ enumerated into $\Lambda_{i,j}$ at stage s proceed as follows. Let $\psi = \psi_0 * \sigma$, where ψ_0 is the coding part of ψ. For each σ' which is a one element extension of σ and such that $\sigma' \in \Lambda_2$, enumerate $\psi_0 * \sigma'$ into $\Lambda_{i,j}$. Let $|\sigma| = l$, let $f_0(\psi) = \tau$ and let $f_1(\psi) = \phi$.
If ψ requires $\mathcal{P}_0$ extension: then check to see if there is $\tau * \tau'$ in Λ_0 of length l such that $\phi \subseteq \Psi_i(\tau * \tau')$ and such that this is not true of any proper initial

segment of $\tau * \tau'$. If so, and if σ is the leftmost string in Λ_2 of length l, then enumerate $\psi * g_0(\tau')$ into $\Lambda_{i,j}$ for each such τ'.
If ψ requires $\mathcal{P}_1$ extension: then check to see if there is $\phi * \phi'$ in Λ_1 of length l such that $\tau \subseteq \Psi_j(\phi * \phi')$ and such that this is not true of any proper initial segment of $\phi * \phi'$. If so, and if σ is the leftmost string in Λ_2 of length l, then enumerate $\psi * g_1(\phi')$ into $\Lambda_{i,j}$ for each such ϕ'.

Any infinite string has an infinite number of initial segments in $\Lambda_{i,j}$ iff it has an initial segment enumerated into $\Lambda_{i,j}$ at every stage of the construction iff it is an infinite path through the downward closure of $\Lambda_{i,j}$. It is clear that the downward closure of $\Lambda_{i,j}$ is a computable set of strings. For each $A \in \mathcal{Q}_{i,j}$ there is a unique element of $[\Lambda_{i,j}]$, $\pi(A)$ say, such that $f_0(\pi(A)) = A$. Uniqueness here follows from the fact that when we enumerate $\psi * g_0(\tau')$ into $\Lambda_{i,j}$ at stage $s+1$, for example, we insist that (using the terminology of that stage) $\tau * \tau'$ in Λ_0 is of length $|\sigma|$, $\phi \subseteq \Psi_i(\tau * \tau')$ and that $\phi \not\subseteq \Psi_i(\tau'')$ for any $\tau'' \subset \tau * \tau'$. Since $\pi(A)$ is computable in A it must hold that $A \equiv_T \pi(A)$. Any element of $[\Lambda_{i,j}]$ which is not $\pi(A)$ for some $A \in \mathcal{Q}_{i,j}$ is a finite string ψ concatenated with an element of Λ_2, and so is of degree in $\boldsymbol{\gamma}$. Thus $S([\Lambda_{i,j}]) = S(\mathcal{Q}_{i,j}) \cup \boldsymbol{\gamma}$ as required.

□

Corollary 8.1. $(\mathfrak{P}, <)$ *is a lattice.*

9. Acknowledgements

The authors would like to thank Doug Cenzer, Steve Simpson, Antonin Kucera and Carl Jockusch for their interest and for many very useful comments and suggestions.

References

DC. D. Cenzer, Π^0_1 classes, in the *Handbook of Computability Theory*, Studies in logic and the foundations of mathematics, vol 140, Elsevier, 1999.

CS. D. Cenzer and R. Smith, The ranked points of a Π^0_1 set, *Journal of Symbolic Logic*, 54, pp. 975-991, 1989.

CCS. D. Cenzer, P. Clote, R. Smith, R. Soare and S. Wainer, Members of countable Π^0_1 classes, *Annals of Pure and Applied Logic*, 31, pp. 145-163, 1986.

CC. P. Cholak, R. Coles, R. Downey and E. Herrmann, Automorphisms of the Lattice of Π^0_1 Classes; Perfect Thin Classes and ANC Degrees, *Transactions of the American Mathematical Society*, Vol. 353, No. 12, pp. 4899-4924, 2001.

SC. J. Cole and S. Simpson, Mass problems and hyperarithmeticity, *Journal of Mathematical Logic*, volume 7, issue 2, pp. 125-143, December 2007.

BC. S.B. Cooper, *Computability theory*, Chapman & Hall, CRC Press, Boca Raton, FL, New York, London (2004).

OD. O. Demuth, Remarks on the structure of tt-degrees based on constructive measure theory, *Comment. Math. Carolinae*, 29, pp. 233-247, 1988.

RD. R. Downey, On Π^0_1 classes and their ranked points, *Notre Dame Journal of Formal Logic*, 32, pp. 499-512, 1991.

DH. R. Downey and D. Hirschfeldt, Algorithmic Randomness and Complexity, monograph in preparation.

JM. C. Jockusch and T. McLaughlin, Countable retracing functions and Π^0_2 predicates, *Pacific J. Math.*, 30, pp. 67-93, 1969.

JS1. C. Jockusch and R. Soare, Π^0_1 classes and degrees of theories, *Transactions of the American Mathematical Society*, 173, pp. 33-56, 1972.

JS2. C. Jockusch and R. Soare, Degrees of members of Π^0_1 classes, *Pacific Journal of Mathematics*, 40, pp. 605-616, 1972.

SK. S. Kleene, Recursive predicates and quantifiers, *Transactions of the American Mathematical Society*, 53, pp. 41-73, 1943.

KP. S. Kripke and M.B Pour-El, Deduction-preserving "recursive isomorphisms" between theories, *Bulletin of the American Mathematical Society*, 73, pp. 145-148, 1967.

AK. A. Kucera, Measure, Π^0_1 classes and complete extensions of PA, in *Recursion Theory Week (Proceedings Oberwolfach 1984)*, Lecture Notes in Mathematics, vol 1141, pp. 245-259, 1985.

AL. A.E.M. Lewis, Π^0_1 classes, strong minimal covers and hyperimmune-free degrees, *Bulletin of the London Mathematical Society*, December 2007.

MP. D. Martin and M.B. Pour-El, Axiomatixable Theories with Few Axiomatizable Extensions, *Journal of Symbolic Logic*, 35 (2), pp. 205-209, 1970.

AN. A. Nies, Computability and Randomness, monograph in preparation.

HP. M.G. Peretyakin, Finitely Axiomatizable Theories, Scientific Books, ISBN 0-306-11062-8.

FS. F. Stephan, Martin-Löf Random and PA-complete Sets. (TR 58) *ASL Lecture Notes in Logic*, 27, pp. 342-348, 2006.

SS. S. Simpson, An extension of the recursively enumerable Turing degrees, *Journal of the London Mathematical Society*, 75, pp. 287-297, 2007.

SS2. S. Simpson, Mass problems and randomness, *Bulletin of Symbolic Logic*, 11, pp. 1-27, 2005.