NONCUPPABLE ENUMERATION DEGREES VIA FINITE INJURY

CHARLES M. HARRIS*  
DEPARTMENT OF MATHEMATICS,  
LEEDS, UK.

Abstract. We exhibit finite injury constructions of a high $\Sigma^0_2$ enumeration degree incomparable with all intermediate $\Delta^0_2$ enumeration degrees, as also of both an upwards properly $\Sigma^0_2$ high and a low$_2$ noncuppable $\Sigma^0_2$ enumeration degree. We also outline how to apply the same methods to prove that, for every $\Sigma^0_2$ enumeration degree $b$ there exists a noncuppable degree $a$ such that $b' \leq a'$ and $a'' \leq b''$, thus showing that there exist noncuppable $\Sigma^0_2$ enumeration degrees at every possible level of the high/low jump hierarchy.

1. Introduction

In the local structure of enumeration degrees, the noncuppable degrees form a subclass of the properly $\Sigma^0_2$ degrees. This follows from the result that every $\Delta^0_2$ degree cups [CSY96]. In fact every noncuppable degree $a$ is downwards properly $\Sigma^0_2$ in the sense that every non zero degree below $a$ is properly $\Sigma^0_2$. Accordingly the study of noncuppable degrees can be seen as a way forward in the investigation of the distribution of the properly $\Sigma^0_2$ degrees. A central question in this regard is that of the distribution of such degrees relative to the high/low jump hierarchy. Cooper and Copestake, who pioneered research in this area, showed in [CC88] that there exists a high properly $\Sigma^0_2$ enumeration degree. More recently Giorgi proved in [Gio08] that there exists a high noncuppable degree. On the other hand, another recent result by Giorgi, Sorbi and Yang has shown that every total nonlow $\Sigma^0_2$ enumeration degree bounds a noncuppable degree. Now, the standard embedding of the Turing degrees into the enumeration degrees preserves the jump operation. So we can deduce, via standard results [Sac63] on the high/low jump hierarchy in the context of the local Turing degrees, that there exists a (relative to $\leq_e$) low$_2$ $\Sigma^0_2$ noncuppable enumeration degree. We note that constructive proofs of high properly $\Sigma^0_2$ and noncuppable enumeration degrees in the literature involve $\psi''$ priority tree arguments whereas the proof of the existence of a low$_2$ noncuppable degree just mentioned is carried out using a derivative construction. With this in mind, the main aim of the present paper is to show, using a simple observation


I wish to emphasise that, although this paper presents a simplified approach to known results in the local enumeration degrees, many of the underlying ideas and techniques used are attributable to the authors Bereznyuk, Coles, Cooper, Copestake, Giorgi, Sorbi, Yang and Yi in the following publications: [BCS00, CC88, CSY96, Glo08, GSY].
on the relationship between enumeration reducibility and relative computable enumerability, that both results can be obtained using finite injury constructions. In preparation for these results we firstly prove the existence of a high $\Sigma^0_2$ enumeration degree incomparable with any intermediate $\Delta^0_2$ degrees. Moreover, in conclusion, we outline the strategy underlying the priority tree proof presented in [Harb] of the result that there exist noncuppable $\Sigma^0_2$ enumeration degrees lying at every possible level of the high/low jump hierarchy. We refer the reader to [Coo90] and [Sor97] for an introduction to both the global and local structure of enumeration degrees and we assume the reader to be conversant with Turing and other basic decision reducibilities, as also with the standard notation used in this context as found for example in [Soa87], [Odi89] or [Coo04].

We assume $\{W_e\}_{e \in \omega}$ to be a standard listing of c.e. sets with associated finite c.e. approximations $\{W_{e,s}\}_{e \in \omega}$, and $\{D_n\}_{n \in \omega}$ to be a computable listing of finite sets. We also assume $(x, y)$ to be a standard computable pairing function over the integers. A set $A$ is defined to be enumeration reducible to a set $B$ ($A \leq_e B$) if there exists an effective procedure that, given any enumeration of $B$ as input, will output an enumeration of $A$. Equivalently, $A \leq_e B$ iff there exists a c.e. set $W$ such that, for all $x \in \omega$,

$$x \in A \iff \exists n [\langle x, n \rangle \in W \ & D_n \subseteq B].$$  \hfill (1.1)

We define $\{\Phi^X_e\}_{e \in \omega}$ to be the effective listing of enumeration operators such that for any set $X$,

$$\Phi^X_e = \{ x \mid \exists n [\langle x, n \rangle \in W_e \ & D_n \subseteq B] \}.$$  

Also, we use the notation $\Phi^{W_e}_{X, s}$ to denote the finite approximation to $\Phi^X_e$, derived from $W_{e,s}$. For simplicity we allow a certain amount of ambiguity in our notation, by sometimes equating $W_e$ with the operator $\Phi_e$, and in the case of finite sets, using the letter $D$ or similar to denote both a finite set and its index in the listing of finite sets specified above. Likewise, in the context of enumeration reducibility we identify functions with their graphs, so that for example, $g \leq_e f$ means that the graph of $g$ is enumeration reducible to the graph of $f$.

For $r \in \{e, T\}$, we use the notation $x_r$ for the equivalence classes generated by the reducibility preordering $\leq_r$ or, in other words, the enumeration and Turing degrees respectively, whereas $\text{deg}_e(X)$ is notation for the $\leq_e$ degree of $X$. $0_e$ is the enumeration degree of the c.e. sets and $0_T$ the Turing degree of the computable sets. $\langle D_r, \leq \rangle$ denotes the upper semilattice of $\leq_r$ degrees in which the join operation is defined by $\text{deg}_r(X) \cup \text{deg}_r(Y) = \text{deg}_r(X \oplus Y)$. Note that we use $D_r$ not only to denote the class of degrees underlying $\langle D_v, \leq \rangle$ but also, for simplicity, as shorthand for the structure itself. $D_r(\leq x_r)$ denotes the class (and substructure) of degrees $\{ y_r \mid y_r \leq x_r \}$.

We use $K$ to denote the (Turing) halting set $\{ e \mid e \in W_e \}$ and $K_X$ to denote the Turing jump $\{ e \mid \varphi^X_e(e) \downarrow \}$ of $X$ where $\{\varphi^X_e\}_{e \in \omega}$ is a standard computable enumeration of oracle Turing machines. The enumeration semihalting set relative to $X$ is defined to be the set $K^X_\downarrow = \{ e \mid e \in \Phi^X_e \}$ and the enumeration jump of $X$ is defined to be the set $J^X_K = K^X_\downarrow \cup K^X$ (or $\leq_1$ equivalently $\chi^X_K$). The associated jump and double jump of $\leq_r$ degree $x_r$ are written $x_r'$ and $x_r''$. We also use the notation $K^n_X$, $J^n_X$ and $x^n_r$ for the iterated versions of the set and degree related jumps respectively. For $n > 0$ we say that that $x_r$ is $r$-low if $x_r^{n} = 0^n_r$, is $r$-high if $x_r^{n} = 0^{n+1}_r$, and we use the terms $r$-low and $r$-high in the case when $n = 1$. If the
context is unambiguous we use the shorthand $\text{low}_n$, and high$_n$ as also the notation $\mathbb{L}_n$ and $\mathbb{H}_n$ to denote the respective classes of degrees (with $\mathbb{L}_0 = 0$ and $\mathbb{H}_0 = \mathbf{0}'$ by convention). Likewise also we use $\mathbb{I}$ to denote the class of intermediate degrees $\{x \mid \forall n[0^a < x^a < 0^{a+1}]\}$.

We say that a set $A$ is total if $A \leq_e A$ (i.e. $\chi_A \leq_e A$). The paradigm example of a total set is of course a total function. Note also that any $\Pi^0_1$ set $X$—and in particular $\mathbb{K}$—is total since $\mathbb{X}$ is computably enumerable. We say that an enumeration degree is total if it contains a total set.

**Lemma 1.1.** If $Y$ is total then $X \leq_e Y$ iff $X$ is c.e. in $Y$.

**Proof.** To prove the non trivial direction, suppose that $X$ is c.e. in $Y$. Then $X = \{z \mid \varphi^Y(z) \downarrow\}$ for some Turing oracle machine $\varphi$. Also, by totality, $\overline{Y} = \overline{\Phi}^Y$ for some enumeration operator $\Phi$. Define the enumeration operator

$$\Phi = \{\langle z, D \rangle \mid \exists E[E \subseteq \overline{\Phi}^D \& \varphi^D(z) \downarrow \& Q^{-}(\varphi^D(z), z) = E]\}$$

where $Q^{-}(\varphi^D(z), z)$ denotes the set of negative queries made during the computation of $\varphi^D(z)$. It is now easily checked that $X = \Phi^Y$.

From this follows the following standard result.

**Corollary 1.2.** If both $X$ and $Y$ are total then $X \leq_e Y$ iff $X \leq_T Y$.

**Note 1.3** ([McEs84]). For any sets $X$ and $Y$, $X \leq_1 K_X$ and $K_X \leq_0 X$ whereas, if $X \leq_e Y$, then $K_X \leq_1 K_Y$. Thus, by Lemma 1.1 and 1-completeness of $K_X$, if $X$ is total then, $K_X \equiv_1 K_X$. In particular $\mathbb{K} \equiv_1 \mathbb{K}$ and thus, for any set $Z$, $\mathbb{K} \leq 1 \mathbb{K}$ (i.e. $\mathbb{K} \equiv_1 J_Z$), and $\mathbf{0}'_e = \text{deg}_{e}(\mathbb{K}) (= \text{deg}_{e}(\mathbb{K} \oplus \mathbb{K}))$. Notice that this also implies that $\mathbb{K} \rightarrow \mathbb{K}_Z$, i.e. that $J_Z \equiv_{e} \mathbb{K}_Z$, if $Z$ is $\Sigma^0_0$ since in this case $\mathbb{K} \equiv_{e} Z \equiv_{e} \mathbb{K} \equiv_{e} \mathbb{K}_Z$.

This leads us to the well known result that the embedding $\iota : \mathcal{D} \rightarrow \mathcal{D}_e$ induced by the map $Y \mapsto \chi_Y$, has as image the class (substructure) of total enumeration degrees, preserves semilattice structure and zero; also that the same applies between the local structures (defined below). Moreover by Note 1.3 the jump is preserved under this embedding in that, for any Turing degree $x_T$, if $x_e$ denotes $\iota(x_T)$, then $\iota(x_T') = x'_e$, where $x'_e$ denotes the jump inside $\mathcal{D}_e$ and $\mathcal{D}_e$ respectively.

For $\Gamma \in \{\Sigma^0_1, \Pi^0_1, \Delta^0_2, \Sigma^0_2\}$ we say that a degree $x$, is $\Gamma$ if it contains a set $X \in \Gamma$. Accordingly $\mathcal{D}_e(\leq \mathbf{0}'_e)$ denotes the structure comprising the $\Delta^0_0$ Turing degrees whereas $\mathcal{D}_e(\leq \mathbf{0}'_e)$ denotes the structure comprising the $\Sigma^0_0$ enumeration degrees.

We call these two structures the local Turing and enumeration degrees respectively.

**Lemma 1.4** ([CM85]). Enumeration degree $x$ is low iff $x$ only contains $\Delta^0_0$ sets.

Indeed, if $x$ is low and $X \in x$, then $J_X \leq_e J_0$, i.e. $K_X \oplus \overline{K_X} \leq_e \overline{K}$. But, by Note 1.3, $X \leq_1 K_X$. Thus also $\overline{X} \leq_1 K_X$ and so both $X$ and $\overline{X}$ are $\Sigma^0_2$. On the other hand, if $x$ only contains $\Delta^0_0$ sets then, since $K_X \equiv_e X$, $J_X$ is $\Delta^0_0$; i.e. $J_X \leq_T K$. or, equivalently, $J_X \leq_e J_0$.

**Lemma 1.5** ([CC88]). If $b \leq e_0' \mathbf{0}_e$, there exists an enumeration degree $0_e < a < b$ such that, for every $\Delta^0_0$ enumeration degree $0_e < c < b$, $a \perp c$.

**Definition 1.6.** Enumeration degree $x < e_0' \mathbf{0}_e$ is properly $\Sigma^0_0$ if it contains no $\Delta^0_2$ sets, and is downwards properly (upwards properly) $\Sigma^0_2$ if every $y \in \{\langle z \mid 0_e < z \leq y\rangle \mid y \in \{\langle z \mid 0_e < z \leq e_0' \mathbf{0}_e\} \}$ is properly $\Sigma^0_0$. $x$ is cappable if there exists $y < e_0' \mathbf{0}_e$ such that $\mathbf{0}'_e = x \cup y$ and is noncappable otherwise.
Notice that in the special case of \( b = 0' \) in Lemma 1.5, \( a \) is incomparable with every intermediate \( \Delta^0_2 \) degree. In other words \( a \) is both downwards and upwards properly \( \Sigma^0_2 \).

**Lemma 1.7** ([CSY96]). If \( 0 < x < 0' \) is \( \Delta^0_2 \) then \( x \) is cuppable.

**Corollary 1.8** ([CSY96]). Every noncuppable \( 0 < x < 0' \) is downwards properly \( \Sigma^0_2 \).

This implies that Lemmas 1.9 and 1.10 below also apply with the property of being noncuppable replaced by that of being downwards properly \( \Sigma^0_2 \).

**Lemma 1.9** ([Gio08]). There exists a high noncuppable degree \( x < 0' \).

**Lemma 1.10** ([GSY]). If \( x \leq 0' \) is total and nonlow, then there exists a noncuppable degree \( 0 < y < x \).

The proof of this Lemma 1.10 in [GSY] involves the notion of \( \mathcal{K} \)-hypersimplicity defined in [NS00]. Using the above results this proof can be formulated as follows. Let \( X \in x \) be total and let \( Z \in X \) be a properly \( \Sigma^0_2 \) (i.e. non \( \Delta^0_2 \)) set in \( x \). Then it follows from Lemma 1.1 that there exists \( f \leq_T X \) such that \( Z = \text{Ran}(f) \). By Corollary 1.2, \( X \leq_T \mathcal{K} \) and so \( f \leq_T \mathcal{K} \). Set \( Y = \{ z \mid (3y > z)(f(y) < f(z)) \} \). Then \( Y \leq_n f \leq X \) and also, by relativisation of Dekker’s Theorem, it follows that \( Y \) is \( \mathcal{K} \)-hypersimple. But then \( \text{deg}_n(Y) \) is noncuppable by Corollary 2.5 of [NS00].

**Lemma 1.11.** For every \( n \geq 0 \) there exist total degrees \( x, y \leq 0' \) such that \( x \in \mathbb{H}_{n+1} - \mathbb{H}_n \) and \( y \in \mathbb{L}_{n+1} - \mathbb{L}_n \). There also exist total \( z \leq 0' \) such that \( z \in I \) (the class of intermediate degrees).

**Proof.** Apply the equivalent results [Sac63, Sac67] proved in the context of the \( \Sigma^1 \) Turing degrees in conjunction with the jump preservation properties of the embedding \( \iota : \mathcal{D}(\leq \mathcal{U}^2) \to \mathcal{D}(\leq 0' \mathcal{U}) \) that follow from Note 1.3. \( \Box \)

The next result now follows directly from Lemma 1.10 and Lemma 1.11.

**Corollary 1.12.** There exists a low \( \mathcal{K} \) noncuppable enumeration degree \( x < 0' \).

For the last result of this Section we assume the reader to be conversant with the notion of a good approximation as defined in [LS92].

**Lemma 1.13** ([Joc68]). Every \( \Sigma^0_2 \) set has a good approximation.

2. \( \Delta^0_2 \) Incomparability Via Finite Injury

By Lemma 1.5 applied to the special case of \( b = 0' \) we know that there exists a \( \Sigma^0_2 \) enumeration degree \( a \) such that \( a \perp c \) for every \( \Delta^0_2 \) enumeration degree \( c \notin \{0, 0'\} \). We strengthen this result below by showing that \( a \) can be taken to be high and, moreover, that this can be proved using a finite injury construction.

The standard method for this kind of construction is to define a \( \Sigma^0_2 \) approximation of a set \( A \). Here however we will take advantage of Lemma 1.1 by constructing \( A \) c.e. in \( \mathcal{K} \).

**Note 2.1.** By Corollary 1.2 \( X \leq_n \mathcal{K} \) iff \( X \) is computably enumerable in \( \mathcal{K} \). Thus there exists a uniformly computable in \( \mathcal{K} \) enumeration of finite sets \( \{\hat{B}_{e,s}\}_{e,s \in \omega} \) such that \( \hat{B}_{e,s} \subseteq \hat{B}_{e,s+1} \) for all \( e, s \in \omega \), and such that for every \( \Sigma^0_2 \) set \( B \) there exists \( \varepsilon \) such that \( B = \bigcup_{s \in \omega} \hat{B}_{\varepsilon,s} \). We use the term c.e. in \( \mathcal{K} \) approximation for
the associated approximation \( \{ \hat{B}_{c,s} \}_{s \in \omega} \). We use the term canonical non decreasing c.e. in \( \mathcal{K} \) approximation if moreover \(|\hat{B}_{c,s+1} - \hat{B}_{c,s}| \leq 1 \) for all \( s \in \omega \), and such that, if \( \langle x \rangle = \hat{B}_{c,s+1} - \hat{B}_{c,s} \), then \( x > \max \hat{B}_{c,s} \).

**Lemma 2.2.** There exists a uniformly computable in \( \mathcal{K} \) enumeration of finite sets \( \{ \hat{C}_{i,s} \}_{i,s \in \omega} \) such that \( \{ \hat{C}_{i,s} \}_{s \in \omega} \) is a canonical non decreasing c.e. in \( \mathcal{K} \) approximation for all \( i \in \omega \), and such that for every \( \Delta^0_2 \) set \( C \) there exists \( e \) with the property that \( \{ \hat{C}_{e,s} \}_{s \in \omega} \) is (such) an approximation of \( C \).

**Proof.** Supposing \( (n)_e \) to be the projections of the pairing function, we informally define the action of an appropriate algorithm \( \mathcal{A} \) relative to \( e \in \omega \) as follows. \( \mathcal{A} \) starts at \( e \)-level 0 in \( e \)-state on. If at step \( s \), \( \mathcal{A} \) is still at \( e \)-level 0 and in \( e \)-state on, then \( \mathcal{A} \) checks if \( \hat{B}_{(e)_n,s} \cap \hat{B}_{(e)_1,s} = \emptyset \). If not it switches to \( e \)-state off. Otherwise it checks whether \( 0 \in \hat{B}_{(e)_n,s} \cup \hat{B}_{(e)_1,s} \). If not \( \mathcal{A} \) passes to step \( s + 1 \) (i.e. \( \hat{C}_{e,s} = \emptyset \) and \( \mathcal{A} \) is still at \( e \)-level 0 and in \( e \)-state on). Otherwise \( \mathcal{A} \) enumerates 0 into \( \hat{C}_{e,s} \) (\( \subseteq \hat{C}_{e,t} \) for all \( t \geq s \)) if \( 0 \in \hat{B}_{(e)_n,s} \) and passes to \( e \)-level 1. \( \mathcal{A} \) now repeats the same search, but now for \( 1 \in \hat{B}_{(e)_0,t} \cup \hat{B}_{(e)_1,t} \), for \( t \geq 0 \). Accordingly, for as long as \( \mathcal{A} \) remains in \( e \)-state on it processes each number \( n \geq 0 \) (at \( e \)-level \( n \)) in increasing order. If on the other hand at some stage \( t + 1 \), \( \mathcal{A} \) has reached some \( e \)-level \( m \) but has switched to \( e \)-state off, then \( \mathcal{A} \) trivially passes to step \( t + 2 \), thus resetting \( \hat{C}_{e,t+1} = \hat{C}_{e,t} \).

The algorithm \( \mathcal{A} \) operates by stages, processing one step \( s \) of each \( e \)-level, for every \( e < t \), at stage \( t \). Accordingly, it is easily checked, on the basis of the properties of the enumeration \( \{ \hat{B}_{e,s} \}_{e,s \in \omega} \)—see Note 2.1—that the resulting enumeration of finite sets \( \{ \hat{C}_{e,s} \}_{e,s \in \omega} \) defined by \( \mathcal{A} \) has the appropriate properties. \( \square \)

**Notation.** For any set \( X \) we use the notation \( \text{Inf} = \{ e \mid W_e \text{ infinite} \} \), and \( \text{InfSet}(X) = \{ e \mid \Phi^X_e \text{ infinite} \} \). Also \( X[\hat{e}] \) denotes the set \( \{ (e,x) \mid \langle e,x \rangle \in X \} \) and \( Y = \ast Z \) means that \( Y = Z \) modulo a finite set. We use the standard shorthand \( \Upsilon[s] \) for \( \Upsilon_s \) when the latter is an expression involving a combination of stage \( s \) approximations of operators and sets. For example \( B_e[s] \) and \( \Phi^A_e[s] \) are shorthand for \( B_{e,s} \) and \( \Phi^{A \chi}_{e,s} \) respectively.

**Lemma 2.3** ([Gri03]). If the set \( Y = \{ e \mid C[\hat{e}] \text{ is finite} \} \) for some \( C \leq_e X \), then \( Y \leq_e J_X \).

**Lemma 2.4** ([Hara]). If the set \( X \) has a good approximation, then \( \text{InfSet}(X) \equiv_e J_X^2 \).

We now adapt the techniques introduced in [CC88] and refined in [BCS00] to prove the main result of this Section.

**Theorem 2.5.** There exists a high enumeration degree \( a < 0_4' \) such that, for every \( \Delta^0_2 \) enumeration degree \( 0_e < c < 0_4' \), \( a \perp c \).

**Proof.** We construct a set \( A \) c.e. in \( \mathcal{K} \) such that (for all \( e \in \omega \)) the following requirements are satisfied,

\[
\begin{align*}
P_e : & \quad C_e = \Phi^A_e \quad \Rightarrow \quad C_e \text{ is c.e.} \\
N_e : & \quad A = \Phi^{C_e}_e \quad \Rightarrow \quad K \leq_e C_e \\
H_e : & \quad W_e \text{ infinite } \Leftrightarrow \quad A^{[2e+1]} \text{ finite}
\end{align*}
\]
where \(\{W_e, \varphi_e, C_e\}_{e \in \omega}\) is a computable listing of all c.e. sets, enumeration operators and \(\Delta^0_2\) sets with associated finite c.e. approximations \(\{W_{e,s}\}_{s \in \omega}, \{\varphi_{e,s}\}_{s \in \omega}\) and canonical non-decreasing c.e. in \(K\) approximations \(\{C_{e,s}\}_{s \in \omega}\) for each \(e \in \omega\).

Supposing \(a\) to be the enumeration degree of \(A\), satisfaction of \(H_a\) for all \(e\) ensures that \(a_0 = \emptyset\) since it entails that \(\text{Inf} = \{ e \mid A^{[2e+1]} \}\) is finite \}. Therefore, since \(\text{Inf} \equiv_1 \text{InfSet}(\emptyset)\), as is easily proved, it follows from Lemma 2.3 (setting \(Y = \text{Inf}\) and \(X = C = A\)) that \(\text{InfSet}(\emptyset) \leq_e J_A\). Thus, by Lemma 2.4, \(J^2_0 \leq_e J_A\). The requirements are ordered \(N_e : H_e < P_e < N_{e+1}, H_{e+1}\) for all \(e \in \omega\). (There is no injury between \(N\) and \(H\) requirements.)

**Note 2.6.** Alternatively—using only Lemma 2.3—satisfaction of \(H_e\) can be obtained by ensuring that \(e \in \Phi_e^A\) iff \(A^{[2e+1]}\) is infinite, i.e. that \(K_{\Phi} = \{ e \mid A^{[2e+1]} \}\), where we note that \(K_{\Phi}\) is itself c.e. in \(\Phi\). This approach is used in the proof of Theorem 3.4 below (with the given set \(B\) in place of \(\Phi\)).

There are four parameters that we will use explicitly in the proof. \(n(e, s) \in \omega^{[2e]}\) and \(h(e, s) \in \omega^{[2e+1]}\) are used as upper limits for (numbers enumerated into \(A\) by) \(N_e\) and \(H_e\) respectively. \(n(e, s)\) is also used as a witness in the construction’s strategy for the satisfaction of \(N_e\). The role of \(p(s)\) is to record the least (if any) \(e\) such that \(P_e\) requires attention at stage \(s\) whereas \(u(s)\) records the level of adjustment necessary to \(N\) and \(H\) requirements of lower priority.

The construction can be thought of as essentially comprising 3 modules, one for each type of requirement \(P, N\) and \(H\). In anticipation of the formal proof, a brief description of these modules is given in the following paragraphs.

The \(P\) module working at index \(e\) tries to diagonalise \(C_e = \Phi_e^A\). This module operates under the assumption that \(A^{[\leq 2e+1]}\) is computable. Its strategy is to search for \(x \notin C_e\) such that \(x \in \Phi_e^{[\leq 2e+1]}\). This search starts from stage \(s + 1 > e + 1\) onwards and entails looking for \((x, D) \in \Phi_e\) such that \(x \in C_{e,s}\max C_{e,s} \subseteq C_e\) (see the definition of \(\{C_{e,s}\}_{s \in \omega}\)) \((x, D) \in \Phi_e\) and \(D \subseteq A^{[\leq 2e+1]} \cup \omega^{[2e+1]}\). If such an axiom \((x, D)\) is found, the \(P\) module ensures that at some stage \(P_e\) receives attention—i.e. that for one such axiom, all of \(D^{[> 2e+1]}\) is enumerated into \(A\) and hence that \(C_e(x) \notin \Phi_e^A(x)\). If no such axiom is ever found it will follow that, if \(C_e = \Phi_e^A\) then \(C_e = \Phi_e^{[\leq 2e+1]} \cup \omega^{[> 2e+1]}\), i.e. that \(C_e \leq_e A^{[\leq 2e+1]}\).

Note that the action of enumerating some finite set \(D^{[> 2e+1]}\) into \(A\) might injure lower priority \(N\) and \(H\) requirements since satisfaction of the latter depend on membership of numbers in \(\omega^{[> 2e+1]}\). However this injury is finitary in the sense that for each \(N\) and \(H\) requirement there are only finitely many \(P\) requirements of higher priority, and each such requirement can enumerate into \(A\) at most one finite set. Now \(u(s + 1)\) is defined by the \(P\) module during stage \(s + 1\) to be the (least) number bounding all numbers that it has enumerated into \(A\) up to this point in the construction. The value \(u(s + 1)\) is then passed to the \(N\) and \(H\) modules which use this value as a lower bound when working to satisfy \(N\) and \(H\) requirements of lower priority.

Note also that for \(\langle r, r, r, (R, s, 0), \langle H, h, 1\rangle \rangle\) and \(e \in \omega\), the \(R\) module working at index \(e\) enumerates at the end of any stage \(s + 1 > e\), precisely the set \(\omega^{[2e+1]} \mid r(e, s + 1)\). Moreover, \(r_e(t) = \text{def} r(e, t)\) is a nondecreasing function—and hence \(A^{[\leq 2e+1]} = \omega^{[2e+1]} \mid r(e, s + 1)\). In other words (in the case of \(r = n\) part
of) the role of \( r(e, s + 1) \) is to act as a measure of the length of the initial segment of \( \omega^{[2e+4]} \) put into \( A \) by the end of stage \( s + 1 \).

The \( N \) module working at index \( e \) uses column \( \omega^{[2e]} \) to either diagonalise \( A = \Phi^C e \) or else witness that \( \overline{K} \leq_e C e \). At stage \( s + 1 > e \) the \( N \) module adjusts the value of \( n(e, s) = (2e, y) \) (say). Notice, as explained in the last paragraph, that \( A^{[2e]} = \omega^{[2e]} | n(e, s) \subseteq A \). If \( s = e \), or \( s > e \) and some \( P \) requirement of higher priority receives attention at this stage then \( n(e, s + 1) \) is set to a value \( z \in \omega^{[2e]} \) bounding both \( n(e, s) \) and \( u(s + 1) \). If no such \( P \) requirement receives attention, the \( N \) module operates under the following assumption.

\((*)\) **No \( P \) requirement of higher priority will receive attention subsequent to stage \( s + 1 \).**

The action it takes is to shift \( n(e, s + 1) \) one step upwards to \( \langle e, y + 1 \rangle \) unless it is found that \( y \in K \) and that either of the following two cases holds.

(a) The maximum element of \( C_e[s] \) does not bound the stage \( t_y \) at which \( y \) entered the c.e. approximation \( \{K[s] \}_{x \in \omega} \) to \( K \) specified on page 8 (which can only happen at finitely many stages of the construction if \( C_e \) is infinite).

(b) Case (a) does not hold and \( n(e, s) = (2e, y) \) satisfies condition (2.2) below, i.e. there exists some finite set \( D \subseteq \omega | t_y \), such that \( \langle (2e, y), D \rangle \in \Phi_e \) and \( D \subseteq C_e[s] \).

In both of these cases \( n(e, s + 1) \) is reset to \( n(e, s) \). Now, under the hypothesis that \( C_e \) is infinite and that \((*)\) is valid at stage \( s + 1 \), if case (b) holds, then \( n(e, t) \) will remain at value \( n(e, s + 1) \) at all stages \( t \geq s + 1 \), thus ensuring that \( (2e, y) \in \Phi_e \) (and that \( A^{[2e]} = \omega^{[2e]} | (2e, y) \)). Likewise if case (b) does not hold at stage \( s + 1 \) but does hold at some later stage \( t > s + 1 \) then diagonalisation of \( N_e \) will apply in the same way at stage \( t \) (relative to \( n(e, t) > n(e, s + 1) \)). On the other hand, under the same hypothesis, if for all \( t > s + 1 \) case (b) never holds it will follow that \( A^{[e]} = \omega^{[e]} \) and that, for all \( x \geq y \), the condition (2.3) below holds, and hence that \( \overline{K} \leq C e \).

The \( H \) module working at index \( e \) processes the column \( \omega^{[2e+1]} \). At stage \( s + 1 > e \) the \( H \) module adjusts the value of \( h(e, s) = (2e + 1, z) \) (say). Notice once again in this case that, as explained above, \( A^{[2e+1]} = \omega^{[2e+1]} | h(e, s) \subseteq A \) by construction. If \( s = e \), or \( s > e \) and some \( P \) requirement of higher priority receives attention at this stage, then \( h(e, s + 1) \) is set to a value \( w \in \omega^{[2e+1]} \) bounding both \( h(e, s) \) and \( u(s + 1) \). If no such \( P \) requirement receives attention then the \( H \) module, working under assumption \((*)\), sets \( h(e, s + 1) = (2e + 1, z + 1) \) if \( W_e \) is a subset of \( \{0, \ldots, s\} \); otherwise it resets \( h(e, s + 1) = (2e + 1, z) \). Proceeding in this way, it follows that the \( H \) module enumerates all of \( \omega^{[2e+1]} \) into \( A \) if \( W_e \) is finite. Otherwise, provided that assumption \((*)\) does indeed hold, if \( W_e \) is infinite then \( A^{[2e+1]} = \omega^{[2e+1]} | h(e, s) \).

Finally notice that, for \( (R, i) \in \{(N, 0), (H, 1)\} \) the outcome of the strategy of the \( R \) module working at any index \( d \) that is either \( A^{[2d+1]} \) is a finite initial segment of \( \omega^{[2d+4]} \) or \( A^{[2d+1]} = \omega^{[2d+1]} \).

**The Construction.** \( A \) is enumerated in stages such that \( A = \bigcup_{s \in \omega} A_s \) and \( A_s \) is finite for all \( s \).

**Stage \( s = 0 \).** Define \( A_0 = \emptyset \), \( p(s) = u(s) = 0 \) Also, for all \( e \in \omega \), define \( n(e, 0) = (2e, 0), h(e, 0) = (2e + 1, 0) \).

**Stage \( s + 1 \).** Using \( K \) as Turing oracle proceed as follows.
Step A. (P requirements.) For each $e < s$, such that $P_e$ is not already satisfied, check whether there exists $x < \max C_e, s$ such that $x \notin C_{e, s}$ (i.e. $x \notin C_e$) and finite set $D$, such that

$$\langle x, D \rangle \in \Phi_e \text{ and } D^{\leq 2e+1} \subseteq A_s.$$  \hspace{1cm} (2.1)

If so, we say that $P_e$ requires attention. There are two cases.

a) No $P_e$ ($e < s$) requires attention. Then set $p(s+1) = s$ and $u(s+1) = u(s)$ and go to Step B.

b) Otherwise pick the least $e$ such that $P_e$ requires attention and the least axiom $\langle x, D \rangle$ satisfying (2.1) relative to $e$ and enumerate $D^{\geq 2e+1}$ into $A$. We say in this case that $P_e$ receives attention at stage $s+1$ and that $P_e$ is (henceforth) satisfied. Set $p(s+1) = e+1$ and $u(s+1) = \max \{D^{\geq 2e+1} \cup \{u(s-1)\}\} + 1$, and go to Step B.

Step B. (N requirements.)

Process $N_e$ for each $e \leq s$, under the assumption that $\{K[s]\}_{s \in \omega}$ is a standard c.e. approximation of $\mathcal{K}$. There are two cases.

a) $e \geq p(s+1)$ (and so, if no $P$ requirement has received attention at this stage the only such number is $e = s$). Set $n(e, s+1) = \max \{n(e, s), (2e, u(s+1))\}$.

b) $e < p(s+1)$. Then suppose that $y$ is such that $n(e, s) = (2y, y)$. Proceed according to the following cases.

i) If $y \notin K$ set $n(e, s+1) = (2e, y+1)$.

ii) If $y \in K$ then...

ii.a) If $\max C_e, s \leq \max \{t \mid y \notin K[t]\}$ then set $n(e, s+1) = n(e, s)$.

ii.b) If $\max C_e, s > \max \{t \mid y \notin K[t]\}$ and

$$\exists D[\langle n(e, s), D \rangle \in \Phi_e \text{ and } y \notin K[\max D] \text{ and } D \subseteq C_e] \hspace{1cm} (2.2)$$

then set $n(e, s+1) = n(e, s)$

ii.c) Otherwise. Set $n(e, s+1) = (2e, y+1)$.

To terminate Step B, enumerate $\omega^{[2e]}|n(e, s+1)$ into $A$, for each $e \leq s$.

Step C. (H requirements.) Process $H_e$ for each $e \leq s$. There are two cases.

a) $e \geq p(s+1)$. Then set $h(e, s+1) = \max \{h(e, s), (2e+1, u(s+1))\}$.

b) $e < p(s+1)$. Then suppose that $z$ is such that $h(e, s) = (2e+1, z)$. Test, using oracle $\mathcal{K}$ whether $W_e \subseteq \omega|s+1$. (Yes or No)

i) Yes: set $h(e, s+1) = (2e+1, z+1)$.

ii) No: set $h(e, s+1) = h(e, s) = (2e+1, z)$.

To terminate Step C, enumerate $\omega^{[2e+1]}|h(e, s+1)$ into $A$, for each $e \leq s$. Proceed to stage $s+2$.

Remark. Notice that our use of $\mathcal{K}$ as oracle allows the application of an unbounded search for axiom $\langle n(e, s), D \rangle \in \Phi_e \text{ in } (2.2)$. The same observation applies to the search for axioms in $\Phi_e$ in (3.3) below. Note also that the use of an unbounded search in each of these cases is essential to our strategy for satisfying the associated requirements. On the other hand in (2.1), (3.1), and (3.4), an unbounded search for axioms in $\Phi_e$ is not needed for the satisfaction of the associated requirements.

Verification. Consider $e \in \omega$. Let $s_e > e$ be such that, for all $s \geq s_e$, $P_i$ does not receive attention at stage $s$ for any $i < e$. Note that the existence of $s_e$ follows from the fact that any $P$ requirement can receive attention at most once.
Case $N_e$. Suppose that $A = \Phi^e_c$ and that $C_e$ is infinite. Let $y_e$ be such that $n(e, s_e) = (2e, y_e)$. We show that in this case

$$x \in \mathcal{K} \iff \exists D \left[ \langle 2e, x \rangle, D \rangle \in \Phi_e \& x \notin \mathcal{K}[\max D] \& D \subseteq C_e \right]$$

for all $x \geq y_e$, and thus that $\mathcal{K} \subseteq C_e$.

We firstly show that

$$A^{[2e]} = \omega^{[2e]}.$$ (2.4)

Suppose not. Then by construction there exists $y$ such that $A^{[2e]} = \{ \langle 2e, z \rangle \mid z < y \}$. Clearly it cannot be the case that $y \in \mathcal{K}$. So $y \in \mathcal{K}$ and, since $C_e$ is infinite, it must be the case that Case b.i.B (of Step B) holds at some stage $s \geq s_e$ for $n(e, s) = (2e, y)$ (and for all $t \geq s$). However this implies that $\langle 2e, y \rangle \in \Phi^{e_x}_C - A$, a contradiction.

We now note that, for all $x \geq y_e$, if $x \in \mathcal{K}$ then the right hand side of (2.3) must hold (since $\langle 2e, x \rangle \in A$ and $A = \Phi^c_d$). Also if $x \in \mathcal{K}$ then the right hand side of (2.3) cannot hold as otherwise the construction would ensure that $A^{[2e]} \subseteq \omega^{[2e]} \backslash \langle 2e, x \rangle$ in contradiction with (2.4). Indeed if $x' \in \mathcal{K}$ is the least such $x \geq y_e$ then $\lim_{x \rightarrow \infty} n(e, s) = \langle 2e, x' \rangle$ and $A^{[2e]} = \omega^{[2e]} \backslash \langle 2e, x' \rangle$ by construction.

Case $H_e$. Suppose that $W_e$ is infinite. Then $h(e, s) = h(e, s_e)$ for all $s \geq s_e$ and so $A^{[2e+1]} = \omega^{[2e+1]} \setminus h(e, s_e)$. If, on the other hand, $W_e$ is finite, there exists $t_e \geq s_e$ such that $W_e \subseteq \omega \upharpoonright t_e + 1$. It follows that $h(e, t_e + 1) > h(e, t)$ for all $t \geq t_e$ and that the whole of the set $\omega^{[2e+1]}$ is eventually enumerated into $A$.

Case $P_e$. Suppose that $C_e$ is infinite and that $C_e = \Phi^A_\omega$. We show that this implies that

$$C_e = \Phi^{[\leq 2e+1] \upharpoonright \omega \setminus [> 2e+1]}_e,$$ (2.5)

and hence that $C_e$ is computably enumerable since by construction either $A^{[i]} = \omega^{[i]}$ or is finite for all $i$ (ensuring that $A^{[\leq 2e+1]}$ is computable).

$(\subseteq)$ Obvious since $C_e = \Phi^A_\omega$ and $A \subseteq A^{[\leq 2e+1]} \cup \omega^{[> 2e+1]}$.

$(\supseteq)$ Suppose that $x \in \Phi^{[\leq 2e+1] \upharpoonright \omega \setminus [> 2e+1]}_e$. Let $s_x \geq s_e$ be a stage such that $\max C_e \upharpoonright [s_x] > x$ and $x \in \Phi^{[\leq 2e+1] \upharpoonright \omega \setminus [> 2e+1]}_e \upharpoonright [s_x]$. Then, if $x \notin C_e$, it follows that $x \notin C_e \upharpoonright [s_x]$ and so, if $P_e$ has not already received attention, then it will do so at stage $s_x$. In both cases $C \neq \Phi^A_\omega$, a contradiction. So $x \in C_e$.

From the verification above we are able to conclude that all $P$ and $H$ requirements are satisfied. We have also shown that $N_e$ is satisfied provided that $C_e$ is infinite. On the other hand, satisfaction of $N_e$ in the case when $C_e$ is finite is a corollary of the satisfaction of the $H$ requirements implying that $A$ is high and therefore not computably enumerable. (Note that we can also prove this directly. Indeed, suppose that $C_e$ is finite and that $A = \Phi^c_d$. Then $A$ is c.e. and so $A \leq_w \omega$. Choose $d$ such that $C_d = \omega$ and $A = \Phi^d_{C_d}$. Then, as $C_d$ is infinite, we know that $\mathcal{K} \leq C_d$. But this implies that $\mathcal{K}$ is computably enumerable, a contradiction.) Thus every $N$ requirement is also satisfied.

To conclude this Section we note that Theorem 2.5 can be generalised in the sense that, for every $n \geq 1$ there exists an enumeration degree $a \prec 0^a_n$ such that, for every $\Delta^0_{n+1}$ degree $b, c, < b < 0^a_n, a \perp c$. The proof is the same using the $n$th

---

1I am grateful to the anonymous referee who identified an error at this point in an earlier version of the proof and who suggested this solution.
jump (i.e. $K^n$) as oracle. Moreover it is clear that the degree $a$ of Theorem 2.5 is such that $a \cap d = 0$, for every low degree $d$ (since every degree $0 < c < d$ is $\Delta^0_2$).

In a similar way, for the $n$th level generalisation of Theorem 2.5 with $n > 1$ the corresponding degree $a$ is such that $a \cap d = 0$, for every $d \leq 0^{n-1}$. For example if $n = 2$ then there exists $\Sigma^0_3$ degree $a$ such that $a \cap d = 0$, for every $\Sigma^0_2$ degree $d$.

3. Noncupping $\Sigma^0_2$ Degrees and the High/Low Jump Hierarchy

By Lemma 1.9 and Corollary 1.12, we know that there are noncupppable $\Sigma^0_2$ enumeration degrees of high and low$_2$ jump class. We will now show that the existence of such degrees can be proved using finite injury constructions. Again we will take advantage of Lemma 1.1 by constructing a set $A$ c.e. in $K$ and, likewise, to prove Theorem 3.3, by not only constructing a set $A$ in this way, but also by constructing an auxiliary set $C$ c.e. in $K$.

By inspection of the proof of Theorem 2.5 we are able to see that the construction of the set $A$ can be modified so as to achieve noncuppability of $a = \text{deg}_u(A)$. Thus we obtain the proof that there exist high noncuppable degrees that also have the property of being upwards properly $\Sigma^0_2$.

**Theorem 3.1.** There exists a high noncuppable enumeration degree $a < 0'_c$ such that every $x \in [a, 0'_c]$ is properly $\Sigma^0_2$.

**Proof.** Construct a set $A$ c.e. in $K$ such that (for all $e \in \omega$) the following requirements are satisfied,

\[
P_e : \quad K = \Phi^{B_e \oplus A}_e \Rightarrow K \leq B_e \\
N_e : \quad A = \Phi^{C_e}_e \Rightarrow K \leq C_e \\
H_e : \quad W_e \text{ infinite } \iff A^{[2e+1]} \text{ finite}
\]

where \{$W_e, \Phi_e, B_e, C_e$}$_{e \in \omega}$ is a computable listing of all c.e. sets, enumeration operators, $\Sigma^0_2$ sets and $\Delta^0_2$ sets, with associated finite c.e. approximations \{$W_{e, s}$}$_{s \in \omega}$, \{$\Phi_{e, s}$}$_{e \in \omega}$ c.e. in $K$ approximations \{$B_{e, s}$}$_{s \in \omega}$, and canonical non decreasing c.e. in $K$ approximations \{$C_{e, s}$}$_{s \in \omega}$ for each $e \in \omega$.

The proof proceeds as for Theorem 2.5 except that, during Step A, $P_e$ requires attention if it is not already satisfied and there exists $x \leq s$ and a pair of finite sets $(E, D)$ such that

\[
x \in K \cap (x, E \oplus D) \in \Phi_e[s] \& E \subseteq B_e[s] \& D^{[\leq 2e+1]} \subseteq A^{[\leq 2e+1]}[s] .
\]  

(3.1)

The rest of Step A proceeds just as in the proof of Theorem 2.5 with the finite set $D$ of the pair $(E, D)$ replacing its namesake of (2.1). Note that once again $P_e$ can only receive attention at most once and hence that the verification of the proof proceeds in a similar manner to that of Theorem 2.5 by showing that $K = \Phi^{B_e \oplus A}_e$ implies that $K = \Phi^{B_e \oplus A^{[\leq 2e+1]}(\omega^{[\leq 2e+1]})}_e$. (See also the proof of Theorem 3.3 below.) $\Box$

**Corollary 3.2 (BCS00).** For every enumeration degree $b < 0'_c$ there exists $b \leq c < 0'_c$ such that every $x \in [c, 0'_c]$ is properly $\Sigma^0_2$.

**Proof.** Let $a$ be the enumeration degree in the statement of Theorem 3.1. Choose any $b < 0'_c$ and set $c = b \cup a$. Then $c < 0'_c$ since $a$ is noncuppable. Moreover $[c, 0'_c] \subseteq [a, 0'_c]$ and so every $x \in [c, 0'_c]$ is properly $\Sigma^0_2$. $\Box$
In contrast to the degree $a$ in Theorem 3.1 which has highest possible jump, we now show how to construct a $\Sigma^0_2$ set $A$ such that its enumeration degree is noncuppable and of lowest possible jump (given that all low enumeration degrees are cuppable).

**Theorem 3.3 ([GSY]).** There exists a noncuppable $\Sigma^0_2$ enumeration degree $a > 0'$ such that $a'' = 0''_o$ (i.e. such that $a$ is low$_2$).

**Proof.** We construct sets $A$ and $C$ c.e. in $K$ such that (for all $e \in \omega$) the following requirements are satisfied:

\[
N_e : \quad A \neq W_e \\
L_e : \quad \Phi^A_e \text{ infinite} \iff C[e] \text{ finite} \\
P_e : \quad \overline{K} = \Phi^B_e \Rightarrow \overline{K}_e = B_e
\]

where \{\(W_e, \Phi_e, B_e\)\}$_{e \in \omega}$ is a computable listing of all c.e. sets, enumeration operators and $\Sigma^0_2$ sets with associated finite c.e. approximations \{\(W_{e,s}\)\}$_{s \in \omega}$, \{\(\Phi_{e,s}\)\}$_{s \in \omega}$ and c.e. in $K$ approximations \{\(B_{e,s}\)\}$_{s \in \omega}$ for each $e \in \omega$.

Supposing $a$ to be the enumeration degree of $A$, satisfaction of $L_e$ for all $e$ ensures that $a'' = 0''_o$ since it entails that $\text{InfSet}(A) = \{e \mid C[e] \text{ finite}\}$ and so, by Lemma 2.3, $\text{InfSet}(A) \leq_e J_{\overline{K}}$, since $C \leq_e \overline{K}$ by construction. Thus $J^2_{\overline{K}} \leq_e J_{\overline{K}} = e J^2_0$ by Lemma 2.4.

**Definitions and Notation.** The construction will proceed by stages $s$, each stage being computable in $K$. We use $A_s$ to denote the finite set of numbers enumerated into $A$ by the end of stage $s$.

1) The *Priority of Requirements.*

For $R \in \{N, L, P\}$, the requirements $R_e$ are ordered in terms of priority such that $N_e < L_e < P_e < N_{e+1}$ for all $e \in \omega$.

2) *Environment Parameters.*

We define a number of parameters used by the construction for the satisfaction of individual requirements. Firstly, for clarity and notational convenience, we define the enumerating parameter $W(s) \in F$ (the class of finite sets).

- **Parameters for the $N_e$ requirements.** The outcome function $N(e,s) \in \{0, 1, 2\}$, the witness parameter $w(e,s) \in \omega \cup \{-1\}$ and the restraint parameter $\varepsilon(e,s) \in S' \cup \{0\}$ ($S'$ being the class of singleton sets).

- **Parameters for the $L_e$ requirements.** The outcome parameter $L(e,s) \in \{0, 1\}$, the restraint parameter $\delta(e,s) \in F$, the individual axiom parameter $v(e,s) \in \omega \cup \{-1\}$, the enumerating parameter $V(s) \in F$ and the height (or overall usage) parameter $h(s) \in \omega$.

- **Parameters for the $P_e$ requirements.** The outcome parameter $P(e,s) \in \{1, 2\}$, and the avoidance parameter $\Omega(e,s) \in F$. The definition of $\Omega(e,s+1)$ is:

\[
\Omega(e,s+1) = \bigcup_{i \leq e} (\varepsilon(i,s) \cup \delta(i,s)).
\]  

Accordingly, $\Omega(e,s+1)$ records the finite set of elements that the construction wants to keep out of $A$ for the sake of higher priority requirements $N_i$ and $L_i$ and that it thus cannot enumerate into $A$ at stage $s+1$ for the sake of $P_e$. 

\[
\text{InfSet}(A) \leq_e J_{\overline{K}} = e J^2_0
\]
3) Requiring attention.

Case \( N_e \). We say that \( N_e \) requires attention at stage \( s + 1 \) if \( N(e,s) = 0 \).

Case \( L_e \). We say that \( L_e \) requires attention at stage \( s + 1 \) if \( L(e,s) = 0 \) and for all \( x \in \omega \) and \( D \subseteq F \),

\[
x \notin \Phi_e^A[s] \& (x,D) \in \Phi_e \Rightarrow D \cap (\omega \setminus h(s) - A_s) \neq \emptyset
\]  

(3.3)

where we note that \( h(s) = \max \{ z \mid z \in A_s \} \) by definition.

Case \( P_e \). We say that \( P_e \) requires attention at stage \( s + 1 \) if \( P(e,s) = 1 \) and there exists \( x \leq s \) and a pair of finite sets \( (D,E) \) such that

\[
x \in K \& (x,D \oplus E) \in \Phi_e[s] \& D \subseteq B_x[s] \& E \cap \Omega(e,s + 1) = \emptyset
\]  

(3.4)

where we note that \( \Omega(e,s + 1) \) is a finite set.

4) Resetting.

Resetting \( N_e \). When we say that the construction resets \( N_e \) at stage \( s + 1 \) we mean the following. If \( N(e,s) \in \{0, 2\} \) the construction does nothing (and in this case \( w(e,s + 1) = w(e,s), \varepsilon(e,s + 1) = \varepsilon(e,s) = \emptyset \) and \( N(e,s + 1) = N(e,s) \)). On the other hand, if \( N(e,s) = 1 \) then we set \( w(e,s + 1) = -1, \varepsilon(e,s) = \emptyset \) and \( N(e,s) = 0 \).

Resetting \( L_e \). When we say that the construction resets \( L_e \) at stage \( s + 1 \) we mean the following. If \( L(e,s) = 0 \) we do nothing (and in this case \( \delta(e,s + 1) = \delta(e,s) = \emptyset \) and \( L(e,s + 1) = L(e,s) = 0 \)). On the other hand, if \( L(e,s) = 1 \) then we set \( \delta(e,s + 1) = \emptyset \) and \( L(e,s) = 0 \).

As in the case of Theorem 2.5 we can think of the construction as comprising a module for each of the three types of requirement. Once again, in anticipation of the formal proof a brief description of each of these modules follows below.

The \( P \) module working at index \( e \) tries to diagonalise \( \overline{K} = \Phi_e^{B_e \oplus A} \). Its strategy is to search for \( x \notin \overline{K} \) such that \( x \in \Phi_e^{B_e \oplus (\omega \setminus \Omega)} \) for some finite set \( \Omega \subseteq \overline{A} \). This search starts from stage \( s + 1 > e + 1 \) onwards with \( \Omega \) being the set of elements restrained out of \( A \) by higher priority \( N \) and \( L \) requirements at stage \( s \)—i.e. the set \( \Omega(e,s + 1) \) in the notation of the proof. If the module finds such an \( x \) it will at some stage \( s + 1 \) enumerate a requisite finite set \( E \subseteq \omega \setminus \Omega(e,s + 1) \)—i.e. where, for some \( D \subseteq B_x[s], (x,D \oplus E) \) is an axiom in \( \Phi_e[s] \)—into \( A \) thus ensuring that \( x \in \Phi_e^{B_e \oplus A} - \overline{K} \). On the other hand if this search fails then, under the assumption that \( \Omega(e,s + 1) \) converges in the limit (over stages \( s \in \omega \)) to a finite set \( \Omega(e) \subseteq \overline{A} \), it will follow that \( \overline{K} = \Phi_e^{B_e \oplus A} \) implies that \( \overline{K} = \Phi_e^{B_e \oplus (\omega \setminus \Omega(e))} \), i.e. that \( \overline{K} \leq_e B_e \).

Note that the action of enumerating some finite set \( E \) (for the sake of \( P_e \)) into \( A \) might injure lower priority \( N \) and \( L \) requirements. Indeed, suppose that \( i,j > e \) are such that \( N(i,s) = 1 \) and \( L(j,s) = 1 \). Then this means that the singleton set \( \varepsilon(i,s) \) and the finite set \( \delta(j,s) \) are being restrained out of \( A \) in order to satisfy requirements \( N_i \) and \( L_j \). Then clearly the insertion of \( E \) into \( A \) might interfere with either (or both) of these restraints. So all such \( N \) and \( L \) requirements are reset to their initial state. Now since each \( P \) requirement receives attention at most once and, for any \( N \) and \( L \) requirement there are only finitely many \( P \) requirements of higher priority, the latter can only be reset (i.e. injured) finitely often. Accordingly, for any index \( e \), at every stage \( s + 1 > e + 1 \) the \( N \) and \( L \) modules process \( e \) under the assumption
that no $P$ requirement of higher priority will receive attention subsequence to stage $s+1$. We use $(\ast)$ to denote this assumption once again.

The $N$ module working at index $e$ tries to diagonalise $A = W_e$. Its strategy (at stage $s+1$) is essentially to pick some witness $w(e,s+1) = x$ (say) that has not yet appeared in the construction and and either restrain $x$ out of $A$ (so that $x(e,s+1) = \{x\}$) or put $x$ into $A$ if respectively $x \in W_e$ or $x \notin W_e$. Thus, if the $N$ module performs one of these actions at stage $s+1$ (i.e. $N_e$ receives attention), and assumption $(\ast)$ is correct, then $N_e$ will already be satisfied from this point on in the construction.

The $L$ module working at index $e$ tries to make $\Phi^4_e$ infinite. In doing this it uses at stage $s+1 > e+1$ the parameter $h(s)$ which bounds both the finite set $A_s$ (i.e. every number put into $A$ by the end of stage $s$) and the finite set of numbers being restrained out of $A$ by $N$ and $L$ requirements at the end of stage $s$. Accordingly at stage $s+1$ (provided that $L(e,s) \neq 1$, i.e. that $L_e$ does not appear to be already satisfied) the $L$ module will try to put some finite subset $D \subseteq \omega - (\omega \upharpoonright h(s) - A_s)$ into $A_{s+1} \subseteq A$ to ensure that some $x \notin \Phi^4_e[s]$ enters $\Phi^4_e[t] \subseteq \Phi^4_e$ at some stage $t \geq s+1$ (when the requisite axiom $(x,D)$ has entered $\Phi^4_e[t]$). This is the role of $V(s+1)$ which is simply the union of all those sets that the $L$ module enumerates into $A_{s+1}$ for the sake of requirements $L_i$ such that $i \leq s$. Note that, due to the definition of $h(s)$ this action will cause no injury to any $N$ and (other) $L$ requirements. This is important since, the $L$ module may carry out this action infinitely often for the sake of $e$ in order to make $\Phi^4_e$ infinite. If the $L$ module does succeed in putting some $x \in \Phi^4_e - \Phi^4_e[s]$ it enumerates no numbers into $C[e]$ at stage $s+1$. If on the other hand it cannot achieve this, it knows that for every axiom $(x,D) \in \Phi_e$ such that $x \notin \Phi^4_e[s], D \cap (\omega \upharpoonright h(s) - A_s) = \emptyset$. Accordingly it restrains $\delta(e,s+1) = \omega \upharpoonright h(s) - A_s$ out of $A_{s+1}$. It also enumerates all of $w[e] \upharpoonright s$ into $C_{s+1}$. Now, if assumption $(\ast)$ holds this restraint will stay in place forcing $\Phi^4_e = \Phi^4_e[s]$ and the module will enumerate all of $w[e] \upharpoonright t$ into $C$ at every subsequent stage $t+1 > s+1$ thus making $C[e] = \omega[e]$. Note that the $L$ module can be mistaken when it enumerates numbers into $C[e]$ on the grounds that $\Phi^4_e$ appears to be the same as $\Phi^4_e[s]$, but that the restraint $\delta(e,s+1)$ is later destroyed by the resetting activity for the sake of some higher priority $P$ requirement—and that $\Phi^4_e$ in fact turns out to be infinite. However, as explained above, this situation can only arise finitely often and so no more than a finite initial segment of $\omega[e]$ will be enumerated into $C$ if $\Phi^4_e$ does indeed turn out to be infinite.

Before proceeding to the formal construction note the difference in roles of $V(s+1)$ and $W(s+1)$ at stage $s+1$. The former as described above is enumerated into $A_{s+1}$ for the sake of forcing $\Phi^4_e - \Phi^4_e[s] \neq \emptyset$ for each $e \leq s$, where this turns out to be possible respecting the above conditions. $W(s+1)$ on the other hand is a finite set (perhaps $= \emptyset$) to be enumerated into $A_{s+1}$ if either a $P$ or $N$ requirement receives attention at stage $s+1$.

The Construction. $A$ and $C$ are enumerated in stages such that, for $X \in \{A,C\}$, $X = \bigcup_{s \in \omega} X_s$ and $X_s$ is finite for all $s$.

\footnote{Note that this action might be repeated at different stages $r+1$ for the same $x$, for as long as $(x,D) \notin \Phi[r]$. However since clearly this can only happen finitely often for any such $x$, this “side effect” does not have an effect on the outcome of the construction.}
Stage $s = 0$. Define $A_0 = C_0 = \emptyset$ and, for all $e \in \omega$, $w(e, 0) = v(e, 0) = -1$, $\varepsilon(e, 0) = \delta(e, 0) = 0$, $N(e, 0) = L(e, 0) = h(0) = 0$ and $P(e, 0) = 1$. Note that accordingly $\Omega(e, 0) = \emptyset$ for all $e \in \omega$ by definition. Also define $V(0) = W(0) = \emptyset$.

Stage $s + 1$. Using $\mathcal{K}$ as Turing oracle proceed as follows.

**Step A.** For all $e \leq s$, define $v(e, s + 1)$ as follows. If $L(e, s) = 1$ (i.e. $L_e$ is satisfied for the moment) or $L(e, s) = 0$ and $L_e$ requires attention at stage $s + 1$ then set $v(e, s + 1) = -1$. Otherwise—i.e. if $L(e, s) = 0$ and $L_e$ does not require attention at stage $s + 1$—choose in a uniformly consistent (and computable in $\mathcal{K}$) manner some $(x, D)$ such that $x \notin \Phi_e^A[s]$, $(x, D) \in \Phi_e$ and $D \subseteq A_s \cup \{ z \mid z \geq h(s) \}$ and set $v(e, s + 1) = (x, D)$. Now define the finite set

$$V(s + 1) = \bigcup_{e \leq s, \ x \in \omega, \ v(e, s + 1) = (x, D)} D.$$

**Step B.** Test whether there exists requirement $R \in \{ N_e, L_e, P_e \mid e \leq s \}$ such that $R$ requires attention.

I) If not go straight to Step C.

II) Otherwise. Choose the highest priority $R$ requiring attention. Supposing $e$ to be the index of $R$, there are three cases as described below. In each of the three cases we say that $R$ receives attention.

a) $R = N_e$. In this case choose the least $z$ such that $(e, z) \notin \omega \cup \{ h(s) \cup V(s + 1) \}$.

b) $R = L_e$. In this case, for any axiom $(x, D)$ such that $x \notin \Phi_e^A[s]$ and $(x, D) \in \Phi_e$ it holds that $D \cap \{ \omega \cup \{ h(s) - A_s \} \} \neq \emptyset$. Accordingly set $\delta(e, s + 1) = \omega \cup h(s) - A_s$ and define $L(e, s + 1) = 1$.

c) $R = P_e$. In this case choose the least axiom $(x, D \uplus E)$ satisfying (3.4). Set $W(s + 1) = E$ and define $P(e, s) = 2$ (permanently satisfied). Reset—as defined on page 12—all $N_i$ and $L_i$ such that $i > e$.

**Step C.** If $W(s + 1)$ has not already been defined—as in (a) or (c) above—then set $W(s + 1) = \emptyset$. For all $e \in \omega$ and $\gamma \in \{ w, \varepsilon, \delta, N, L, P \}$, if $\gamma(e, s + 1)$ is not yet defined then define $\gamma(e, s + 1) = \gamma(e, s)$. Now set

$$A_{s+1} = A_s \cup V(s + 1) \cup W(s + 1).$$

Let $z^*$ denote the maximum number in

$$A_{s+1} \cup \bigcup_{e \leq s} (\varepsilon(e, s + 1) \cup \delta(e, s + 1))$$

and define $h(s + 1) = z^* + 1$. To end stage $s + 1$, define

$$C_{s+1} = C_s \cup \{ (e, z) \mid e \leq s \& z \leq s \& L(e, s) = 1 \},$$

and proceed to stage $s + 2$. 

Consider any $e \in \omega$. As Induction Hypothesis we suppose that every requirement $R \in \{ N_i, L_i, P_i \mid i < e \}$ only receives attention at most finitely often. (Notice that as in the proof of Theorems 2.5 and 3.1, it is obvious by construction that each $P$ requirement receives attention at most once.) Accordingly, let $s_e > e$ be such that every such requirement $R$ does not receive attention at or after stage $s_e$. Note that this means that, for every $i < e$ and $\gamma \in \{ w, \varepsilon, \delta, N, L, P \}$, $\gamma(i, t) = \gamma(i, s_e)$ for all $t \geq s_e$. We write this limiting value $\gamma(i)$. We now check that $N_e$, $L_e$ and $P_e$ are satisfied, and that the Induction Hypothesis is justified for $N_e$ and $L_e$. We proceed according to descending priority, noting that $N_e \leq L_e < P_e$ in the priority ordering.

Case $N_e$. At the end of stage $s_e$ the construction ensures that $w(e, s_e) \in A_{s_e}$ iff $w(e, s_e) \notin W_e$. But, for all $t \geq s_e$, $w(e, t) = w(e, s_e) = w(e)$ since $N_e$ cannot be reset after stage $s_e$. If $w(e) \notin W_e$ then $w(e) \in A_{s_e} \subseteq A$ by construction. If $w(e) \notin W_e$ then $\varepsilon(e, s_e) = \varepsilon(e, t) = \varepsilon(e) = \{ w(e) \} \subseteq \overline{A}$, and so $w(e) \notin A$. In other words $N_e$ no longer receives attention and is satisfied.

Case $L_e$. We firstly show that
\[ \Phi_e^A \text{ infinite } \iff \text{ } C[e] \text{ finite.} \tag{3.6} \]

Set $\tilde{s}_e = s_e + 1$. (Thus $\tilde{s}_e$ is such that $N_e$ does not receive attention at any stage $t \geq \tilde{s}_e$.)

Consider any $t \geq \tilde{s}_e$ and suppose that $L(e, t) = 1$. Then there exists some $r < t$ such that $L_e$ received attention at stage $r + 1$ and $L_e$ has not been reset since stage $r + 1$. But, by (3.3), this means that $\delta(e, r + 1) = \omega|\delta(r) - A_e$ and that, for all $(x, D)$,
\[ x \notin \Phi_e^A[r] \text{ } \& \text{ } (x, D) \in \Phi_e \implies D \cap \delta(e, r + 1) \neq \emptyset. \]

Moreover, since by definition of $\tilde{s}_e$ it is also the case that no requirement of higher priority receives attention—and so as a result that $L_e$ cannot be reset at any stage $s \geq t$—it follows that $\delta(e, r + 1) = \delta(e)$. On the other hand, for the same reasons, we know, by an easy induction over stages $s$, that $\delta(e) \subseteq \overline{A}$. So $\Phi_e^A = \Phi_e^A[\omega]$. In other words $\Phi_e^A$ is finite contradicting the hypothesis. Therefore $L(e, s) = 0$ for all $s \geq \tilde{s}_e$ and so $C[e] \subseteq \omega[\omega][e, \tilde{s}_e]$. i.e. $C[e]$ is finite.

\[ \iff \text{ Now suppose that } \Phi_e^A \text{ is finite, and note that by construction } \Phi_e^A = \bigcup_{t \in \omega} \Phi_e^A[\omega] + 1, \text{ and } \Phi_e^A[t] \subseteq \Phi_e^A[t + 1] \text{ for all } t. \text{ Accordingly let } s \geq \tilde{s}_e \text{ be a stage such that } \Phi_e^A = \Phi_e^A[s]. \text{ Then, if } L(e, s) \neq 1 \text{ it is clear that } L_e \text{ will require—and hence receive—attention at stage } s + 1 \text{ since otherwise the construction would ensure that } \Phi_e^A - \Phi_e^A[s] \neq \emptyset, \text{ due to action taken during step } A \text{ of stage } s + 1. \text{ Hence } L(e, s + 1) = 1. \text{ Furthermore, as } L_e \text{ cannot be reset after this stage (by definition of } \tilde{s}_e), \text{ it follows that } L(e, t) = 1 \text{ for all } t \geq s + 1. \text{ So by construction (see (3.5)), } C[e] = \omega[e]. \text{ i.e. } C[e] \text{ is infinite.} \]

Finally notice that the above implies that $L_e$ only receives attention at most once after stage $\tilde{s}_e$, and is satisfied.

Case $P_e$. Let $\tilde{s}_e \geq \tilde{s}_e$ be a stage at or after which $L_e$ does not receive attention at any stage $t \geq \tilde{s}_e$. Thus, by definition of $\tilde{s}_e$, for all such $t$, $\Omega(e, t) = \Omega(e, \tilde{s}_e)$. Accordingly we define $\Omega(e)$ to be this set.

Now suppose that $\overline{K} = \Phi_e^{B_e \oplus A_e}$. We show that, in this case, $\overline{K} = \Phi_e^{B_e \oplus (\omega - \Omega(e))}$. 

NONCUPABLE ENUMERATION DEGREES VIA FINITE INJURY 15
If \( x \in \overline{K} \) then, since \( \Omega(e) \subseteq \overline{A} \)—as is easily proved by a simple induction over \( s \)—it is clear that \( x \in \Phi_e^{B_e(\omega-\Omega(e))} \) follows from our supposition that \( \overline{K} = \Phi_e^{B_e(\omega)} \).

If \( x \notin \overline{K} \) and \( x \in \Phi_e^{\omega(\omega-\Omega(e))} \) then we know that there exists \( s \geq s_e \) and a least axiom \( \langle x, D \oplus E \rangle \in \Phi_e[s] \), \( D \subseteq B_e[s] \) and \( E \cap \Omega(e) = \emptyset \). There are 2 cases.

1) \( P(e, s) = 2 \). Then there exists \( t < s, z \leq t \) and a pair of finite sets \( (F, G) \) such that \( z \notin \overline{K}, \langle z, F \oplus G \rangle \in \Phi_e[t], F \subseteq B_e[t], \) and \( G \cap \Omega(e, t + 1) = \emptyset \) and such that \( G \) was enumerated into \( A \) at stage \( t + 1 \). But then \( z \notin \Phi_e^{B_e(\omega)} \) (since \( B_e[t] \subseteq B_e \) and \( A_{t+1} \subseteq A \)) whereas \( z \notin \overline{K} \). Contradiction.

2) Otherwise \( P(e, s) = 1 \). In this case the construction enumerates \( E \) into \( A \) at stage \( s + 1 \), so obtaining \( x \in \Phi_e^{B_e(\omega)} \) and \( x \notin \overline{K} \), once again a contradiction.

This proves that if \( x \notin \overline{K} \) then \( x \notin \Phi_e^{\omega(\omega-\Omega(e))} \).

We thus conclude that \( \overline{K} = \Phi_e^{\omega(\omega-\Omega(e))} \), i.e. that \( \overline{K} \leq_e B_e \), since \( \Omega(e) \) is finite. This concludes the proof.

Having seen in Theorems 3.1 and 3.3 that there exist noncuppable \( \Sigma^0_2 \) enumeration degrees of highest possible and—by Lemma 1.4—lowest possible jump class, a natural question to ask is whether there exist such degrees at every possible level of the high/low jump hierarchy. By this we mean that, excluding the case \( \mathbb{L}_1 \) — \( \mathbb{L}_0 \), Lemma 1.11 applies with the property of being total replaced by that of being noncappable. This question is answered affirmatively as follows.

**Theorem 3.4.** [Harb] For every \( \Sigma^0_2 \) enumeration degree \( b \) there exists a noncappable \( \Sigma^0_2 \) enumeration degree \( a > 0 \) such that \( b' \leq a' \) and \( a'' \leq b'' \).

This is proved—via application of the methodology of the proofs above, as also of priority tree techniques as found for example in Chapter XIV of [Soo87]—using a priority tree construction computable in \( K \) with finite injury along the true path combined with an auxiliary oracle construction. The proof, as presented in [Harb], proceeds by showing that, for every \( \Sigma^0_2 \) set \( B \) there exists a \( \Sigma^0_2 \) set \( A \) such that \( J_B \leq_e J_A, J_A \leq_e J_B^2 \) and also such that \( \text{deg}_e(A) \) is noncappable. To do this, both a set \( A \) c.e. in \( K \) and a set \( C \) c.e. in \( K_B \) are constructed such that, for all \( e \in \omega \),

\[
\begin{align*}
H_e & : e \in \Phi_e^B & \iff & A^{[e]} \text{ infinite } (\equiv \omega^{[e]}) \\
L_e & : \Phi_e^A \text{ infinite } & \iff & C^{[e]} \text{ finite } \\
P_e & : \overline{K} = \Phi_e^{B_e(\omega)} & \Rightarrow & \overline{K} \leq_e B_e
\end{align*}
\]

where \( \{ \Phi_e, B_e \}_{e \in \omega} \) is a computable listing of pairs of all enumeration operators and \( \Sigma^0_2 \) sets with associated finite c.e. approximations \( \{ \Phi_{e,s} \}_{s \in \omega} \) and c.e. in \( K \) approximations \( \{ B_{e,s} \}_{s \in \omega} \) for each \( e \in \omega \).

Now, satisfaction of \( \{ H_e \}_{e \in \omega} \) ensures\(^3\) that \( J_B \leq_e J_A \) by Lemma 2.3 and the last sentence of Note 1.3. Moreover, satisfaction of \( \{ L_e \}_{e \in \omega} \) ensures that \( \text{InfSet}(A) \leq_e J_B^2 \), also by Lemma 2.3, which means that \( J_A^2 \leq_e J_B^2 \) by Lemma 2.4 and Lemma 1.13. On the other hand, satisfaction of \( \{ P_e \}_{e \in \omega} \) clearly ensures that \( \text{deg}_e(A) \) is noncappable.

\(^3\)The reader will notice in the formulation of requirement \( H_e \) and in contrast with that of \( H_e \) of Theorems 2.5 and 3.1—which is a special case—the strategy mentioned in Note 2.6.
Note also that judicious choice of \( B \) ensures that \( \deg_{\omega}(A) > 0 \). Indeed, consider any \( \Sigma^0_2 \) degree \( b \). If \( b' > 0' \), apply the construction to some set \( B \in b \) and let \( a = \deg_e(A) \). Accordingly, \( b' \leq a' \) and \( a'' \leq b'' \) whereas the conjunction of the conditions \( b' > 0' \) with \( b' \leq a' \) ensures that \( a > 0 \). If, on the other hand, \( b' = 0' \), using Lemma 1.11 pick \( c \) such that \( c' > 0' \) and \( c'' = 0'' \)—i.e. such that \( c \in \mathbb{I}_2 - L_1 \). Apply the construction to some set \( B \in c \) and again let \( a = \deg_e(A) \). Then \( (0'' \cup b' < c' \leq a' \) whereas \( a'' \leq c'' = b'' (= 0'' ) \). Moreover, once again \( a > 0 \), due to the conjunction of \( c' > 0' \) with \( b' \leq a' \).

**Sketch of the strategy used to satisfy the requirements of Theorem 3.4.** The requirements are ordered in terms of priority so that \( H_e < L_e < P_e < H_{e+1} \) for all \( e \in \omega \). Moreover the construction uses the tree \( L = 2^{<\omega} \) for the satisfaction of \( L \) requirements. Accordingly each node \( \gamma \in L_e = \operatorname{def} \{ \alpha \mid \alpha \in L \land |\alpha| = e \} \) is assigned to the requirement \( L_e \). At each stage \( s + 1 \) the construction defines a path \( \alpha_{s+1} \) of depth \( s + 1 \) in \( L \) and a finite tree \( T_{s+1} \) defined inductively as the collection of nodes \( \{ \gamma \mid \gamma \leq \alpha_{s+1} \lor (\gamma \in T_s \land \gamma < L \alpha_{s+1}) \} \) where \( T_0 = \{ \lambda \} \) (with \( \lambda \) being the null string and \( <_L \) the lexicographical ordering over \( 2^{<\omega} \)). We say that the stage \( s + 1 \) is \( \sigma \)-true if \( \sigma \) is on the path through \( L \) at stage \( s + 1 \), i.e. if \( \sigma \subseteq \alpha_{s+1} \). We define \( T \) to be the union of all the finite approximations \( T_s \). Clearly \( T \) is infinite. Moreover as \( T \subseteq 2^{<\omega} \) it has a leftmost infinite path \( f \) which we call the true path.

Each stage \( s + 1 \) of the construction comprises two parts. In the first part the requirement of highest priority in the set \( \{ H_e \mid e \leq s \} \cup \{ P_e \mid e < s \} \) that requires attention—if there is any such—receives attention (i.e. is processed). In the second part the \( s + 1 \) stage path \( \alpha_{s+1} \) is constructed in \( L \) and every \( \tau \subseteq \alpha_{s+1} \) of length \( n \) \( s \) is processed. Note that *processing* a requirement or an \( L \) node means nontrivially redefining parameters associated with that requirement or node.

The main parameters that we shall mention in this outline are as follows: \( H_e \)'s restraint \( \varepsilon(e, s) \in \text{COF}_s \cup \{ 0 \} \) where \( \text{COF}_s \) is the class of cofinite subsets of \( \omega^{|e|} \); \( L \) node \( \sigma \)'s outcome \( L(\sigma, s) \in \{ 0, 1 \} \); \( \sigma \)'s oracle call outcome \( c (\sigma, s) \in \{ 0, 1 \} \); \( L_e \)'s overall oracle call outcome \( c(e, s) \in \{ 0, 1 \} \); \( \sigma \)'s index restraint \( \chi(\sigma, s) \) defined to be either \( 0 \) or a finite index set of the form \( \{ i \mid |\sigma| < i \leq t \} \) for some \( t < s \); also \( P_e \)'s avoidance parameter \( \Omega(e, s+1) \) defined to be either \( 0 \) or a set of the form \( \bigcup \{ \varepsilon(i, s) \mid i \leq r \} \) for some \( r \leq s \). Each of these parameters is set to a default value at stage 0: for all \( e \in \omega \) and \( \sigma \in L \), \( L(\sigma, 0) = C(\sigma, 0) = C(e, 0) = 0 \) whereas \( \varepsilon(e, 0) = \chi(\sigma, 0) = \Omega(e, 0) = 0 \). Note that, for every requirement \( R \in \{ H_e \}_{e \in \omega} \cup \{ P_e \}_{e \in \omega} \) and node \( \sigma \in L \), if \( R \) or \( \sigma \) is not processed at stage \( s + 1 \) then its associated parameters are automatically reset to their values at stage \( s \). We call this automatic resetting. We also define parameter \( W(s + 1) \) (with \( W(0) = 0 \)) to be either \( 0 \) or, in the case when a \( P \) requirement receives attention at stage \( s + 1 \), the finite set to be enumerated into \( A_{s+1} \) at the end of stage \( s + 1 \) for the sake of this requirement.

Similarly to the proofs above we can consider the construction as being essentially comprised of three modules, one for each type of requirement. In order to give an overview of the construction we now give an informal description of each module relative to some index \( e \).

Note that we say that \( H_e \) and its restraint \( \varepsilon(e, s+1) \) are active at stage \( s + 1 \) if \( e \leq s \). Note also that by definition of the construction each \( H \) and \( P \) requirement can receive attention at most once—as in the proofs of Theorems 2.5 and 3.1—after which the requirement no longer requires attention. Hence each such requirement
only requires attention finitely often.

The Case $H_e$. At stage $s + 1 = e + 1$ the $H$ module defines the $H_e$ restraint $\varepsilon(e, s + 1) = \omega^{[e]} - (A_s \cup W(s + 1))^{[e]}$. (Note that $\varepsilon(e, t) = \emptyset$ for all $t \leq e$.) In other words, $\varepsilon(e, s + 1)$ is precisely the cofinite set of numbers in $\omega^{[e]}$ that have not been enumerated into $A$ up to this point in the construction. At subsequent stages $s + 1 > e + 1$, for as long as $e \notin \Phi^B[t]$ the $H$ module resets $\varepsilon(e, s + 1) = \varepsilon(e, s)$ unless some $P$ requirement of higher priority receives attention. In this latter case the $H$ module sets $\varepsilon(e, s + 1) = \varepsilon(e, s) - W(s + 1)^{[e]}$. Now, every $P$ requirement can receive attention at most once. Also, by definition, the construction prohibits the $L$ module from enumerating any numbers into $\varepsilon(e, t)$ at any stage $t$. Thus if $e \notin \Phi^B[t]$ it follows that $\varepsilon(e, s + 1) = \omega^{[e]} - A^{[e]}_{s+1}$ at every stage $s + 1 \geq e + 1$ and also that $\varepsilon(e, s + 1) = \varepsilon(e, s)$ for cofinitely many stages $s$. It follows that $\lambda^{[e]}$ is finite in this case. On the other hand, if $e \in \Phi^B[t]$, then there exists stage $r$ such that $e \in \Phi^B[t]$ at all stages $t \geq r$. Accordingly from stage $r + 1$ onwards the $H$ module signals to the construction that $H_e$ requires attention. Since every $H$ and $P$ requirement of higher priority only requires attention at most finitely often there will be a stage $t + 1 \geq r + 1$ at which no such requirement requires attention. At the least such stage $t + 1$ the $H$ module registers that $e \in \Phi^B[t]$ and resets $\varepsilon(e, t + 1) = \emptyset$. We say that $H_e$ receives attention at stage $t + 1$ in this case. Then, at every subsequent stage $s + 1 > t$ the $H$ module enumerates $\omega^{[e]}|s$ into $A$ ensuring that $\lambda^{[e]} = \omega^{[e]}$.

The Case $L_e$. The $L$ module comprises a main module and an oracle module. The role of the main module is to enumerate numbers into $A$ in such a way as to ensure that either $\Phi^A_e$ is infinite or, if not, that the oracle module has enough information to establish that $\Phi^A_e$ is indeed finite. The oracle module, assesses at every stage the state of the construction with the aid of information from $K_B$ and enumerates numbers into $C$ according to this assessment.

The main module processes an $L$ node $\sigma \in \mathcal{L}_e$ (i.e., of length $e$) at every stage $s + 1 \geq e + 1$. It decides the outcome $L(\sigma, s + 1) \in \{0, 1\}$ which dictates which node of length $e + 1$ is eligible to be processed next (i.e., either $\sigma^0$ or $\sigma^1$). At the same time it passes information to the oracle module enabling the latter to decide its own outcome—relative to $e - C(e, s + 1) \in \{0, 1\}$. The oracle module enumerates numbers into $C^{[e]}$—in fact the set $\omega^{[e]}|s$—if and only if $C(e, s + 1) = 1$. The main module has a different strategy according to the value of $L(\sigma, s)$.

$L(\sigma, s) = 0$. In this case either $s + 1$ is the first stage at which the construction visits $\sigma$, or at the last visit (i.e., some stage $t \leq s$ such that $\sigma \subseteq \alpha_t$) the main module established $L(\sigma, t) = 0$. In this case the main module tries to put some $x \in \Phi^A_e - \Phi^A_0[s]$ by finding some such number $x$ with associated axiom $\langle x, D \rangle \in \Phi_e$ such that $D \cap \varepsilon(i, s + 1) = \emptyset$ for all $i \leq s$ (i.e., such that this action will interfere with no active $H$ restraints). We consider the two possible outcomes here.

(I) There is such an axiom $\langle x, D \rangle$. In this case the main module enumerates $D$ into $A_{s+1}$, so entailing that $x \in \Phi^A_e - \Phi^A_0[s]$, and records $L(\sigma, s + 1) = 0$ thus dictating that $\sigma^0$ is eligible to be processed next. The oracle module records $C(e, s + 1) = 0$ (i.e., no numbers are enumerated into $C^{[e]}$).

\footnote{Note that the $L$ module can enumerate numbers into $A^{[e]}$ before $H_e$ becomes active but that, by definition, $\varepsilon(e, t) = \emptyset$ at all such stages $t$ (i.e., if $t \leq e$).}
(II) There is no such axiom. In this case the main module sets \( L(\sigma, s + 1) = 1 \), so dictating that \( \sigma^\geq 1 \) is eligible to be processed after \( \sigma \). It then processes \( \sigma \) under the following assumptions.

(\#) For all \( i \leq c \) and all \( j < c \), neither requirement \( H_i \) nor requirement \( P_j \) will receive attention at any stage subsequent to \( s + 1 \). This means that for each \( i \leq c, \varepsilon(i, t) = \varepsilon(i, s + 1) \) at every stage \( t \geq s + 1 \).

(\#\#) For every \( c < i \leq l \), and stage \( t \geq s + 1 \), the restraint \( \varepsilon(i, t) \) associated with \( H_i \) will not be interfered with by any (higher priority) \( P \) requirements.

In other words, taken in conjunction with (\#), this is the assumption that if any requirement \( P_j \), such that \( c < j < i \), receives attention at a stage \( t + 1 \) subsequent to \( s + 1 \), then no numbers belonging to the the set \( \varepsilon(i, t) \) can be enumerated into \( A_{s+1} \) for the sake of \( P_j \). Accordingly any change to \( \varepsilon(i, t) \) can only come about as a result of \( H_i \) receiving attention (meaning that \( i \) enters \( \Phi^B \)).

Now, under assumptions (\#) and (\#\#) the fact that by definition of this case, for any axiom \((x, D) \in \Phi, \) such that \( x \notin \Phi^A \llbracket s \rrbracket, D \cap \bigcup \{ \varepsilon(i, s + 1) \mid i \leq s \} \neq \emptyset \) we are able to surmise (a)-(d) below.

**Notation.** Given any finite set \( D \) and stage \( s \) we use \( \gamma_{D,s} \) to denote the set of indices \( i \leq s \) such that that some number in \( D \) belongs to \( \varepsilon(i, s + 1) \), i.e.

\[
\gamma_{D,s} = \{ i \mid i \leq s \text{ & } D \cap \varepsilon(i, s + 1) \neq \emptyset \}.
\]

Also, if \( x \) enters \( \Phi^A \) due to the fact that \((x, D) \in \Phi, \) and \( D \subseteq A \), we say that \( x \) enters \( \Phi^A \) via \( D \).

(a) \( \Phi^A = \Phi^A \llbracket s \rrbracket \) unless there exists an axiom \((x, D) \) such that (i) \((x, D) \in \Phi, \) and \( x \notin \Phi^A \llbracket s \rrbracket \) and for which \((ii) \) the set \( \gamma_{D,s} \) is a subset of \( \{ j \mid e < j \leq s \} \) and also such that, \((iii) \) for every \( i \in \gamma_{D,s}, H_i \) receives attention at some stage \( t \) subsequent to \( s + 1 \), thus causing all of \( \omega^{[i]} \) (and in particular all of \( D^{[i]} \)) to enter \( A \).

(b) If there exists an axiom \((x, D) \) for which the conditions (i)-(iii) apply, and if also the construction carries out (iv) below, then \( x \) will necessarily enter \( \Phi^A \llbracket t \rrbracket \) at some subsequent stage \( t \).

(iv) All of \( D \cap \big( \omega - \bigcup \{ \varepsilon(i, s + 1) \mid i \leq s \} \big) \) is enumerated into \( A_{s+1} \).

(c) If the axiom \((x, D) \) satisfies (i)-(ii), then (iii) applies to the finite set \( D \) if and only if \( \gamma_{D,s} \subseteq K_B = \{ i \mid i \in \Phi^B \} \) by definition. Hence, assuming that an axiom \((x, D) \) satisfying (i)-(ii) exists, and that the relevant part of \( D \) is enumerated into \( A \) as specified by (iv), the oracle \( K_B \) knows whether or not \( x \) will enter \( \Phi^A \llbracket t \rrbracket \) via \( D \) at some subsequent stage \( t \).

(d) If there exist axioms \((x, D) \) and \((y, E) \) satisfying (i) and (ii) and such that \( \gamma_{D,s} = \gamma_{E,s} \) then, provided that (iv) is performed for both \( D \) and \( E \), we know that \( x \) subsequently enters \( \Phi^A \) via \( D \), if and only if \( y \) subsequently enters \( \Phi^A \) via \( E \), if and only if, for every \( i \in \gamma_{D,s}, H_i \) receives attention after stage \( s + 1 \). This is because, under the present assumptions the only way that any numbers can be enumerated into \( A^{[i]} \) for any \( c < i \leq s \) is due to \( H_i \) receiving attention, in which case all of \( \omega^{[i]} \) will eventually be enumerated into \( A \). Thus, for the oracle module to be able to establish for any \( \gamma \subseteq \{ i \mid e < i \leq s \} \) whether some number will enter \( \Phi^A \llbracket t \rrbracket - \Phi^A \llbracket s \rrbracket \) at some stage \( t \geq s + 1 \) due to the existence of
some \( \langle x, D \rangle \) such that \( \gamma_{p,x} = \gamma \) and such that \( \langle x, D \rangle \) satisfies (i)-(iii), the main
module only needs to find one single such witness \( \langle x, D \rangle \) such that \( \gamma = \gamma_{p,x} \) satisfying (i) and to enumerate the relevant part of \( D \) into \( A_{s+1} \) as prescribed in (iv).

In the light of the above comments the main module tests for every finite subset \( \gamma \subseteq \{ j \mid e < j \leq s \} \), whether there exists an axiom \( \langle x, D \rangle \) satisfying (i)-(ii) and such that \( \gamma_{p,x} = \gamma \). It then stores in a (temporary) parameter \( S(\sigma, s + 1) \) each \( \gamma \) for which this search is successful and, for each such \( \gamma \), chooses one axiom \( \langle x, D \rangle \) that witnesses this and enumerates the relevant part of \( D \) into \( A_{s+1} \) as dictated\(^5\) by (iv). It then passes every \( \gamma \in S(\sigma, s + 1) \) to the oracle module which tests whether \( \gamma \subseteq K_B \). In the case when this test fails for every \( \gamma \in S(\sigma, s + 1) \) the oracle module sets parameter \( \bar{C}(\sigma, s + 1) = 1 \) otherwise it sets \( \bar{C}(\sigma, s + 1) = 0 \) (i.e. \( \bar{C}(\sigma, s + 1) = 1 \) if and only if, for every \( \gamma \in S(\sigma, s + 1), \gamma \notin K_B \)). To understand the meaning of the parameter \( \bar{C}(\sigma, s + 1) \) consider a stage \( s_e \) such that the following assumption holds relative to \( s_e \).

(\dagger) No \( P \) or \( H \) requirement of higher priority than \( L_e \) receives attention at any stage \( t \geq s_e \). Moreover, at every such stage \( t \) the construction visits no node to the left of \( f_e \)—where \( f_e \) is the node of length \( e \) (i.e. in \( L_e \)) on the true path \( f \).

Now, if \( s + 1 \geq s_e \) and \( \sigma \) is on the true path (i.e. \( \sigma = f_e \)) both assumptions (* *) hold for \( \sigma \) at stage \( s + 1 \). (Why this happens will become clear during the discussion of the \( P \) module below). But in this case, from the above discussion we see that \( \bar{C}(\sigma, s + 1) = 1 \) if and only if \( \Phi^A_e = \Phi^A[s] \). However (still under the assumption \( s + 1 \geq s_e \)) the oracle module has not yet verified that \( \sigma \) is on the true path. To do this it now tests for every \( \tau \) such that \( \tau^{-1} \subseteq \sigma \) whether \( \bar{C}(\tau, s + 1) = 1 \). Only in the case when this test is positive for each such \( \tau \) does the oracle know that \( \sigma \) is on the true path and so sets \( C(\varepsilon, s + 1) = 1 \) (causing \( \omega^{[e]} \) to be enumerated into \( C_{s+1} \)). Otherwise, it knows that \( \sigma \) is not on the true path and sets \( C(\varepsilon, s + 1) = 0 \). Likewise it sets \( C(\varepsilon, s + 1) = 0 \) without undergoing this further test in the case when \( \bar{C}(\sigma, s + 1) = 0 \).

The main module meanwhile, on setting \( L(\sigma, s + 1) = 1 \) defines the index restraint \( \chi(\sigma, s + 1) = \{ j \mid e < j \leq s \} \). At subsequent stages \( t \) such that \( \sigma \subseteq \alpha_{t} \) or \( \sigma \ll_{L} \alpha_{e} \) and \( L(\sigma, t) = 1 \), the role of \( \chi(\sigma, t) \) is to protect at stage \( t + 1 \), for every \( i \in \chi(\sigma, t) \), the \( H \) restraints \( \varepsilon(i, t) \) from interference from any higher priority requirement \( P_{k} \) such that \( e \leq k < i \). (See 3.7 below.) Note that at stages \( r > s + 1 \), \( \chi(\sigma, r) = \chi(\sigma, s + 1) \) by automatic resetting for as long as \( L(\sigma, r) = 1 \) at each of the construction’s subsequent visits to \( \sigma \). However if at some later stage \( t \), the main module sets \( L(\sigma, t) = 0 \) then \( \chi(\sigma, t) \) is reset to \( \emptyset \).

\( L(\sigma, s) = 1 \). This means that the current assessment made by the main module relative to \( \sigma \) is that \( \Phi^A_e \) is finite. Suppose that that \( r + 1 < s + 1 \) is the stage at which this assessment was made as described in the case \( L(\sigma, s) = 0 \) above. Note that this means in effect that the main module inferred at stage \( r + 1 \) that \( \Phi^A_e = \Phi^A[r] \). The main module now tests whether \( \Phi^A[s] = \Phi[r] \).

\(^5\)Note therefore that in this case also the numbers \( n \) enumerated into \( A_{s+1} \) are such that \( n \notin \varepsilon(i, s + 1) \) for all \( i \leq s \). In other words the action taken here, once again, interferes with no active \( H \) requirement.
(The reader is reminded that $\Phi^A_t [t] \subseteq \Phi^A_t [t + 1]$ at any stage $t$.) If not it sets $L(\sigma, s + 1) = 0$ and $\chi(\sigma, s + 1) = 0$ whereas the oracle module sets $C(e, s + 1) = 0$ (and $\hat{C}(\sigma, s + 1) = 0$). If on the other hand it is the case that $\Phi^A_t [s] = \Phi^r [r]$, then the main module resets $L(\sigma, s + 1) = 1$ and $\chi(\sigma, s + 1) = \chi(\sigma, s)$ whereas the oracle module resets $\hat{C}(\sigma, s + 1) = \hat{C}(\sigma, s)$ and sets $C(e, s + 1) = 0$ if $\hat{C}(\sigma, s + 1) = 0$. Otherwise, in the case when $\hat{C}(\sigma, s + 1) = 1$, it retests for every $\tau$ such that $\tau^\omega \subseteq \sigma$ whether $\hat{C}(\tau, s + 1) = 1$ and, only if this is the case does it set $C(e, s + 1) = 1$; if not it sets $C(e, s + 1) = 0$

Note that this retesting is for the situation in which $\hat{C}(\tau, r + 1)$ was set to 1 but $C(e, r + 1)$ was set to 0 due to the fact$^6$ that $C(\tau, r + 1) = 0$ for some $\tau$ such that $\tau^\omega \subseteq \sigma$. Now, at stage $r + 1$, if $r + 1 < s_e$—where $s_e$ is the stage satisfying assumption (i)—it may happen that the construction visits $\tau^\omega$ at some stage $r + 1 < t < s + 1$ thus entailing that $\tau$’s parameters have been redefined, with the possible result that now $\hat{C}(\tau, s + 1) = 1$. In other words, if $r + 1 < s_e$ it may be the case that $\hat{C}(\tau, r + 1) = 1$ but $C(e, r + 1) = 0$, whereas it is in fact the case that $\Phi^A_e = \Phi^A_e [r]$. Accordingly, the above retesting procedure will eventually catch this error. (Note that this small complication is due the fact that there is no rightwards destruction/reinitialisation of nodes in this construction.)

Notice also that for any $\alpha \in \mathcal{L}_e$ and any $\alpha$-true stage $t + 1$ such that $t + 1 < s_e$, it may be the case that $C(e, t + 1) = 1$ whereas it turns out later that $\Phi^A_e \neq \Phi^A_e [t]$. However if the construction visits $\alpha$ infinitely often there will be a stage $p + 1$ at which the present case applies relative to $\alpha$, i.e. the main module sees that $\Phi^A_e [p] \neq \Phi^A_e [t]$ and outputs $L(\alpha, p + 1) = 0$ so forcing $\hat{C}(\alpha, p + 1) = 0$ (and $C(e, p + 1) = 0$). In other words in this case also earlier mistakes by the oracle module will eventually be caught and corrected.

We will see that the strategy described above satisfies requirement $L_e$, on the premise that the assumptions that we use are valid for any node $\alpha$ on the true path at every large enough $\alpha$-true stage$^7$. Indeed, consider the node $\sigma \in \mathcal{L}_e$ on the true path $f$ (i.e. $\sigma = f_e$). Then providing that there exists a stage $s_e$ after which assumptions (a) and (b) apply relative to $\sigma$, and supposing also that $i$ is such that $\sigma^\omega i$ is on the true path, if $i = 0$ then $\Phi^A_e$ is $\infty$ and if $i = 1$ then $\Phi^A_e$ is finite. Indeed if $i = 0$ but there are nevertheless infinitely many $\sigma^\omega i$-true stages it follows that at infinitely many $\sigma^\omega 0$-true stages the main module registers—in the manner described in the case $L(\sigma, s) = 1$ above—that the cardinality of $\Phi^A_e$ has increased. In the other case when $i = 0$ there exists a stage $p$ such that at every $\sigma$-true stage $t + 1 \geq p$ the construction ensures that some number $x$ enters $\Phi^A_e - \Phi^A_e [t]$ as described in the case $L(\sigma, s) = 1$ above$^8$. If $i = 1$ then there exists a least stage $r + 1$ such that at every $\sigma$-true stage $t + 1 \geq r + 1$, $L(\sigma, t + 1) = 1$. However, this means that the construction verifies that $\Phi^A_e [t] = \Phi^A_e [r]$ at every such stage, i.e. that $\Phi^A_e$

---

$^6$Notice that if $C(e, r + 1)$ was set to 0 because $\hat{C}(\sigma, r + 1) = 0$ then this means that enough numbers were enumerated into $A$ at stage $r + 1$ in the sense of (iv) above to ensure that there is indeed a number $x$ that enters $\Phi^A_e - \Phi^A_e [r]$ at a later stage. Thus no error can arise in this case.

$^7$That this is the case will become clear during our discussion of the $P$ module.

$^8$In fact a number $x$ can “enter” $\Phi^A_e - \Phi^A_e [t]$ in this manner more than once since $x$ only actually enters the approximation to $\Phi^A_e$ at some stage $r \geq t + 1$ at which the relevant axiom $(x, D)$ enters $\Phi_e [r]$. However action taken on behalf of $x$ in this sense can obviously only happen before stage $r + 1$, i.e. finitely often.
is finite. On the other hand, at any late enough stage \( s+1 \) in the construction the oracle module will correctly assess, whenever \( s+1 \) is \( \alpha^{-1} \)-true for some \( \alpha \in \mathcal{L}_e \), whether \( \alpha^{-1} \) is on or to the right of the true path. Accordingly this will mean that it is only when \( \sigma^{-1} \) is on the true path that \( C(e, s+1) = 1 \) for infinitely many \( s \)—and only in this case will the oracle module be able to enumerate \( \omega^{(e)} \) in \( C \) in the limit.

The Case \( P_e \). For as long as \( P_e \) has not received attention, the \( P \) module processes \( e \) at every stage \( s + 1 > e + 1 \) taking into account the finite tree of \( L \) nodes \( T_s \) constructed during the latter part of the previous stage \( s \). Indeed, supposing that \( \sigma \in \mathcal{L}_e \) (i.e. \( |\sigma| = e \)) and \( i \in \{0, 1\} \) are such that stage \( s \) was \( \sigma^{-i} \)-true (i.e. \( \sigma^{-i} \subseteq \alpha_s \)), then \( P \) processes \( e \) relative to the set \( \Omega(e, s+1) \) defined as follows.

\[
\Omega(e, s+1) = \bigcup \{ \varepsilon(i, s) \mid i \leq e \vee (\exists \tau) [\tau \in T_s[\sigma^{-i} \& i \in \chi(\tau, s)] \}
\]  

(3.7)

where \( T_s[\sigma^{-i}] =_\text{def} \{ \tau \mid \tau \in T_s \& (\tau <_L \sigma^{-i} \& \tau \subseteq \sigma) \} \). In other words \( \Omega(e, s+1) \) contains the union of \( H \) restraints \( \varepsilon(i, s) \) such that either \( H_i \) is of higher priority than \( P_e \) or such that interference with \( \varepsilon(i, s) \) might falsify the \( L \) oracle module’s assessment that \( \hat{C}(\tau, s) = 1 \) (associated with outcome \( L(\tau, s) = 1 \)) for some \( \tau \) to the left of, or above, \( \sigma^{-i} \) in the tree \( T_s \). To understand the latter part of this definition, consider any such \( \tau \) and stage \( r+1 \leq s \) such that the \( L \) module set \( L(\tau, r+1) = 1 \) in the manner described in the case \( L(\sigma, s+1) = 0 \) above. Then the oracle’s assessment \( \hat{C}(\tau, r+1) \) was informed in part by the assumption \( (*) \)—with the pair \( [\tau], r \) replacing \( e, s \) in its formulation (in conjunction with the assumption \( (*) \) that, for all \( i \leq |\tau|, \varepsilon(i, t) = \varepsilon(i, r+1) \) at every stage \( t \geq r+1 \)). So we see that, in general, for subsequent stages \( t+1 > r+1 \), the definition of \( \Omega(e, t+1) \) entails that this assumption is protected relative to the action taken on behalf of requirement \( P_e \) at stage \( t+1 \), for as long as the \( t \) stage path \( \alpha_t \) always either subsumes (i.e. \( \tau^{-1} \subseteq \alpha_t \)) or is to the right of \( \tau^{-1} \) (i.e. \( \tau^{-1} \leq \alpha_t \)).

The \( P \) module tries to diagonalise \( \mathcal{K} = \Phi^B_{e+\mathcal{A}} \) by searching for some \( x \in \mathcal{K} \) and axiom \( (x, D \oplus E) \in \Phi_e[x] \) such that \( D \subseteq B_x[s] \) and \( E \subseteq \omega - \Omega(e, s+1) \). If there exists such an axiom the \( P \) module will signal to the construction that \( P_e \) requires attention in order to enumerate \( E \) into \( A \). This will happen at stage \( s+1 \)—in which case we say that \( P_e \) receives attention—if no higher priority \( H \) or \( P \) requirement requires attention at this stage. Now, since every \( H \) and \( P \) requirement requires attention at most finitely often, there will be a stage \( q_e \) such that, if \( t+1 \geq q_e \) (and providing \( P_e \) has not already received attention), if such an axiom \( (x, D \oplus E) \) is found, then \( P_e \) will indeed receive attention. So, as \( \Phi^B_{e+\mathcal{A}}[x] \subseteq \Phi^B_{e+\mathcal{A}} \) for any stage \( r \), this means that \( x \in \Phi^B_{e+\mathcal{A}} = \mathcal{K} \) and so \( P_e \) is satisfied henceforth.

We are therefore able to deduce that, if indeed \( \mathcal{K} = \Phi^B_{e+\mathcal{A}} \), then \( P_e \) never receives attention, and accordingly we are able to specify a computable set \( \Omega(e) \) such that \( \Phi^B_{e+\mathcal{A}} = \Phi^B_{e+\mathcal{A}(\omega-\Omega(e))} \) in this case. To see this, suppose now that \( \sigma \in \mathcal{L}_e \) and \( i \in \{0, 1\} \) are such that \( \sigma^{-i} \subseteq f \). In other words \( \sigma^{-i} \) is the node of depth \( e+1 \) on the true path in \( T \). Let \( t_e > e \) be a stage such that the construction never visits any node \( \tau <_L \sigma^{-i} \) subsequent to \( t_e \). Then note that for any such \( \tau \), \( \chi(\tau, t) = \chi(\tau, t_e) \) for all stages \( t \geq t_e \) by automatic resetting. Moreover if \( \gamma \) (on the

---

\(^9\)If \( \tau \subseteq \sigma \), then \( \hat{C}(\tau, s) = 1 \) is the outcome registered by the \( L \) oracle module at stage \( s \). However, if \( \tau <_L \sigma^{-i} \) then this means, by automatic resetting, that the \( L \) oracle module registered \( \hat{C}(\tau, r) = 1 \) at the last stage \( r < s \) at which the construction visited \( \tau \).
true path) is such that $\gamma^{-1} \subseteq \sigma^{-i}$ then also $\chi(\gamma, t) = \chi(\gamma, t_e)$ for all $t \geq t_e$, partly by automatic resetting, but otherwise because at each $\sigma^{-i}$-true stage $r + 1 \geq t_e$, outcome $L(\gamma, r + 1) = 1$ and so $\chi(\gamma, r + 1)$ is reset to $\chi(\gamma, r)$ by definition. If however $\gamma^{-1} \not\subseteq \sigma^{-i}$ then $\chi(\gamma, r + 1) = \emptyset$ at every $\sigma^{-i}$-true stage $r + 1 \geq t_e$, since in this case $L(\gamma, r + 1) = 0$. Now supposing that $r_e$ is the first $\sigma^{-i}$-true stage at or after stage $t_e$, we therefore know that the set of indices

$$I_\Omega(e, s) = \{ i \mid i \leq e \vee (\exists r)[r \in T_e[\sigma^{-i} \& i \in \chi(\tau, s)] \}$$

(where $T_e[\sigma^{-i}$ is defined as in (3.7)) satisfies $I_\Omega(e, s) = I_\Omega(e, r_e)$ for every $s \geq t_e$ (or $r_e$) such that $s$ is $\sigma^{-i}$-true. Now since each $H$ and $P$ requirement only receives attention at most once we can choose a stage $p_e \geq r_e$ such that for every $i$ satisfying $i \leq e$ or $i \in I_\Omega(e, r_e)$, $H_i$ does not receive attention subsequent to stage $p_e$ and such that, for every $j \leq e$, $P_j$ does not receive attention subsequent to stage $r_e$. Moreover we can obviously also choose $p_e$ to be a $\sigma^{-i}$-true stage. Then it follows that $\Omega(e, s + 1) = \Omega(e, p_e + 1)$ for all stages $s + 1 \geq p_e + 1$ such that $s$ is $\sigma^{-i}$-true. This is because, since by definition $\Omega(e, s + 1) = \bigcup \{ \varepsilon(i, s) \mid i \in I_\Omega(e, s) \}$, the only way that this set can change at such a stage is due (by definition of $p_e$) to numbers being enumerated into $A$ by requirements $P_k$ such that $k > e$. Consider any such $k$ and stage $p + 1 \geq p_e + 1$. Then, supposing that $\alpha \in L_k$ and $j \in \{0, 1\}$ are such that stage $p$ is $\alpha^{-j}$-true, we can see by definition of $t_e$ ($\leq p_e$) that either $\sigma^{-i} \subseteq \alpha^{-j}$ or $\sigma^{-i} <_L \alpha^{-j}$. From this we can deduce that $\Omega(e, p_e + 1) \subseteq \Omega(k, p + 1)$. Now the $P$ module, at stage $p + 1$ can only enumerate numbers into $\omega - \Omega(k, p + 1)$ for the sake of $P_k$. It follows from this that (i) no $P$ requirement can enumerate numbers into $\Omega(e, p_e + 1)$ at subsequent stages and hence that $\Omega(e, q + 1) = \Omega(e, p_e + 1)$ for all $q \geq p_e$ such that $q$ is $\sigma^{-i}$-true and that (ii) $\Omega(e, p_e + 1) \subseteq T$.

By the above discussion, and defining $G_{\sigma^{-1}}$ to be the set of $\sigma^{-i}$-true stages, we can set

$$\Omega(e) =_{\text{def}} \Omega(e, p_e + 1) = \lim_{s \in G_{\sigma^{-1}}} \Omega(e, s + 1).$$

Since $\Omega(e) \subseteq T$ we know that $\Phi_{e_e, \oplus A}^{B_e} \subseteq \Phi_{e_e, \oplus (\omega - \Omega(e))}^{B_e}$, but $x \not\in \Phi_{e_e, \oplus A}^{B_e}$, then $x \in K$ by hypothesis and so at some large enough stage $s + 1$ such that $s$ is $\sigma^{-i}$-true, there will be an axiom $(x, D \oplus E) \in \Phi_e[s]$ such that $D \subseteq B_e[s]$ and $E \cap \Omega(e, s + 1) = \emptyset$ and so, without loss of generality, we can suppose that for one such axiom $(x, D \oplus E)$, $E$ will be enumerated into $A_{s + 1}$ entailing that $x \in \Phi_e[B_e, \oplus A] - K$ a contradiction. This shows that $\Phi_{e_e, \oplus A}^{B_e} = \Phi_{e_e, \oplus (\omega - \Omega(e))}$ as claimed above.

We therefore conclude that $P_e$ is satisfied. We can also infer from the discussion of this case (i.e. of the $P$ module) that if $\sigma \in L_e$ is on the true path $f$ in $T$ then there does indeed exist a stage $s_e$ such that for every $\sigma$-true stage $s \geq s_e$ both assumptions $(\ast)$ and $(\ast\ast)$ hold for $\sigma$. We therefore also conclude that $L_e$ is satisfied. Hence taking into consideration our earlier discussion of the $H_e$ module we see that the above outlines a successful strategy relative to the satisfaction of each individual requirement.

Now, in the light of the paragraph on page 17 preceding the above proof sketch we see that the existence of a low2 noncuppable degree is a special case of Theorem 3.4. Also the existence of a high noncuppable degree is another special case of this theorem when $b^\prime = 0_e^\prime$. Furthermore the fact that $b^\prime \leq a^\prime$ and $a^{\prime\prime} \leq b^\prime$
in Theorem 3.4 obviously implies that $a'' = b''$. From these facts and combining Lemma 1.11 with Theorem 3.4, we do indeed obtain the following result.

**Corollary 3.5 ([Harb]).** For every $n > m ≥ 0$ there exist noncuppable enumeration degrees $x, y ≤ 0''_e$ such that $x ∈ H_{m+1} - H_m$ and $y ∈ L_{n+1} - L_n$. There also exists noncuppable $z ≤ 0''_e$ such that $z ∈ I$.

Finally, the reader should also note, by Corollary 1.8, that the property of being noncuppable can be replaced by that of being *downwards properly $Σ^0_2$*, in Corollary 3.5.

**References**


E-mail address: harris.charles@gmail.com

URL: http://www.maths.leeds.ac.uk/~charlie