Math. Struct. in Comp. Science (2009), *vol.* 19, *pp.* 169–191.
doi:10.1017/S0960129508007378 Printed in the United Kingdom

Cupping Δ_2^0 enumeration degrees to $\mathbf{0}_e'$ †

MARIYA IVANOVA SOSKOVA‡ and GUOHUA WU§

‡*Department of Pure Mathematics, University of Leeds, Leeds LS2 9JT, United Kingdom*
Email: mariya@maths.leeds.ac.uk
§*School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore 639798, Republic of Singapore*
Email: guohua@ntu.edu.sg

Received 17 November 2007

In this paper we prove that every non-zero Δ_2^0 e-degree is cuppable to $\mathbf{0}_e'$ by a 1-generic Δ_2^0 e-degree (and is thus low and non-total), and that every non-zero ω-c.e. e-degree is cuppable to $\mathbf{0}_e'$ by an incomplete 3-c.e. e-degree.

1. Introduction

Intuitively, we say that a set A is *enumeration reducible* to a set B, denoted $A \leqslant_e B$, if there is an effective procedure to enumerate A given any enumeration of B. More formally, $A \leqslant_e B$ if there is a computably enumerable set W such that

$$A = \{x : (\exists u)[\langle x, u\rangle \in W \ \&\ D_u \subseteq B]\}$$

where D_u is the finite set with canonical index u. Therefore, every c.e. set gives rise to an operator, which is called an *enumeration operator*. We will identify an enumeration operator with the c.e. set that defines it. An enumeration operator is denoted by a capital Greek letter, and the elements of an enumeration operator are called axioms.

Let $\equiv_e$ denote the equivalence relation generated by $\leqslant_e$, and $[A]_e$ be the equivalence class of A, called the *enumeration degree* (*e-degree*) of A. The degree structure $\langle \mathscr{D}_e, \leqslant\rangle$ is defined by setting $\mathscr{D}_e = \{[A]_e : A \subseteq \omega\}$, and setting $[A]_e \leqslant [B]_e$ if and only if $A \leqslant_e B$. The operation of least upper bound is given by $[A]_e \cup [B]_e = [A \oplus B]_e$, where $A \oplus B = \{2x : x \in A\} \cup \{2x+1 : x \in B\}$. The structure $\mathscr{D}_e$ is an upper-semilattice with the least element $\mathbf{0}_e$, which is the collection of computably enumerable sets. Gutteridge (1971) proved that $\mathscr{D}_e$ does not have minimal degrees (see Cooper (1982)).

An important substructure of $\mathscr{D}_e$ is given by the Σ_2^0 e-degrees, that is, the e-degrees of Σ_2^0 sets. Cooper (1984) proved that Σ_2^0 e-degrees are exactly those e-degrees below $\mathbf{0}_e'$, which is the e-degree of $\overline{K}$. An e-degree is Δ_2^0 if it contains a Δ_2^0 set, which is a set A with a computable approximation f such that for every element x, $f(x, 0) = 0$ and $\lim_s f(x, s)$

† An extended abstract of this paper was first published as Soskova and Wu (2007).
‡ This author is supported by the Marie Curie Early Training grant MATHLOGAPS (MEST-CT-2004-504029).
§ This author is partially supported by a start-up grant No. M48110008 and a research grant No. RG58/06 from NTU.

exists and is equal to $A(x)$. Cooper and Copestake (1988) proved that there are e-degrees below $\mathbf{0}'_e$ that are not Δ^0_2, and these e-degrees are called *properly* Σ^0_2 e-degrees.

In this paper we are mainly concerned with the cupping property of Δ^0_2 e-degrees. An e-degree $\mathbf{a}$ is cuppable if there is an incomplete e-degree $\mathbf{c}$ such that $\mathbf{a} \cup \mathbf{c} = \mathbf{0}'_e$. Cooper, Sorbi and Yi proved that all the non-zero Δ^0_2 e-degrees are cuppable and that there are non-cuppable Σ^0_2 e-degrees (Cooper *et al.* 1996).

Theorem 1.1 (Cooper *et al.* 1996). Given a non-zero Δ^0_2 e-degree $\mathbf{a}$, there is an incomplete total Δ^0_2 e-degree $\mathbf{c}$ such that $\mathbf{a} \cup \mathbf{c} = \mathbf{0}'_e$, where an e-degree is total if it contains the graph of a total function. Meanwhile, non-cuppable Σ^0_2 e-degrees exist.

In this paper we first prove that each non-zero Δ^0_2 e-degree $\mathbf{a}$ is cuppable to $\mathbf{0}'_e$ by a non-total Δ^0_2 e-degree.

Theorem 1.2. Given a non-zero Δ^0_2 e-degree $\mathbf{a}$, there is a 1-generic Δ^0_2 e-degree $\mathbf{b}$ such that $\mathbf{a} \cup \mathbf{b} = \mathbf{0}'_e$. Since 1-generic e-degrees are quasi-minimal, and 1-generic Δ^0_2 e-degrees are low, $\mathbf{b}$ is non-total and low.

Here, a set A is *1-generic* if for every computably enumerable set S of $\{0,1\}$-valued strings, there is some initial segment σ of A such that either S contains σ, or S contains no extension of σ. An enumeration degree is 1-generic if it contains a 1-generic set. Obviously, no non-zero e-degree below a 1-generic e-degree contains a total function, and hence 1-generic e-degrees are quasi-minimal. Copestake proved that a 1-generic e-degree is low if and only if it is Δ^0_2 (see Copestake (1990)).

Our second result is concerned with cupping ω-c.e. e-degrees to $\mathbf{0}'_e$. A set A is n-c.e. if there is an effective function f such that for each x, we have $f(x,0) = 0$, $|\{s : f(x,s) \neq f(x,s+1)\}| \leqslant n$ and $A(x) = \lim_s f(x,s)$. A is is ω-c.e. if there are two computable functions $f(x,s), g(x)$ such that for all x, we have $f(x,0) = 0$, $|\{s : f(x,s) \neq f(x,s+1)\}| \leqslant g(x)$ and $A(x) = \lim_s f(x,s)$.

An enumeration degree is n-c.e. (ω-c.e.) if it contains an n-c.e. (ω-c.e.) set. It is easy to see that the 2-c.e. e-degrees are all total and coincide with the Π^0_1 e-degrees – see Cooper (1990). Cooper also proved the existence of a 3-c.e. non-total e-degree. As the construction presented in Cooper *et al.* (1996) actually proves that any non-zero n-c.e. e-degree can be cupped to $\mathbf{0}'_e$ by an $(n+1)$-c.e. e-degree, we will prove that any non-zero ω-c.e. e-degree is cuppable to $\mathbf{0}'_e$ by a 3-c.e. e-degree.

Theorem 1.3. Given a non-zero ω-c.e. e-degree $\mathbf{a}$, there is an incomplete 3-c.e. e-degree $\mathbf{b}$ such that $\mathbf{a} \cup \mathbf{b} = \mathbf{0}'_e$.

This is the strongest possible result. We can explain this as follows. Consider the standard embedding ι from $\mathscr{D}_T$ to $\mathscr{D}_e$ given by $\iota(deg_T(A)) = deg_e(\chi_A)$, where χ_A denotes the graph of the characteristic function of A. It is well known that ι is an order-preserving mapping, and that the Π^0_1 e-degrees are exactly the images of the Turing c.e degrees under ι. Now consider a non-cuppable c.e. degree $\mathbf{a}$. $\iota(\mathbf{a})$ is Π_1, and hence ω-c.e., and $\iota(\mathbf{a})$ is not cuppable by any Π^0_1 e-degree, as ι preserves the least upper bounds. Therefore, no 2-c.e. e-degree cups $\iota(\mathbf{a})$ to $\mathbf{0}'_e$.

The results in this paper were presented at the conference 'Computability in Europe 2007' and an extended abstract was published in Soskova and Wu (2007). Due to the limited space available in Soskova and Wu (2007), we were only able to present the basic ideas and a sketch of the verifications. In this paper, we provide complete constructions, together with detailed motivations and full verifications. At each stage in the constructions we perform the 'K-check' and the 'A-check' first, and the corresponding actions of this part can be out of the true path – a crucial feature of both constructions. Furthermore, we introduce the notion of 'the one axiom rule' – at any stage, at most one axiom in Γ can enumerate a number n into $\Gamma^{A,B}$. This is another crucial feature of the construction of Γ, and we hope that this rule can help explain and motivate the constructions of Γ clearly. In Section 3, we introduce the notion of 'pretargets' and 'targets' in the construction of the 1-generic set B to clarify the actions of the strategy. In the verifications, we define the true path as the limit of δ_s, rather than the liminf of δ_s (which is true, but not accurate), where δ_s is the current approximation of the true path at stage s. In Lemmas 3.1 and 4.1 we prove that the true paths are infinite. This was not specified in Soskova and Wu (2007).

Our notation is standard – see Cooper (2004) and Soare (1987) for reference.

2. Basic ideas of the Cooper–Sorbi–Yi cupping

In this section we describe the basic ideas of Cooper, Sorbi and Yi's construction given in Cooper *et al.* (1996). Let $\{A_s\}_{s\in\omega}$ be a Δ_2^0 approximation of the given Δ_2^0 set A, which is assumed to be not computably enumerable. We will construct two Δ_2^0 sets B and E (auxiliary) and an enumeration operator Γ such that the following requirements are satisfied:

$$S \ : \Gamma^{A,B} = \overline{K}$$
$$N_\Phi \ : E \neq \Phi^B.$$

The first requirement is a global requirement guaranteeing that the least upper bound of the e-degrees of A and B is $\mathbf{0}'_e$. Here $\Gamma^{A,B}$ denotes an enumeration operation relative to the enumerations of A and B.

The second group of requirements N_Φ, where Φ ranges over all enumeration operators, guarantee that the e-degree of B is incomplete, as the e-degree of E is not below that of B.

To satisfy the global requirement S, we will construct an enumeration operator Γ such that $\overline{K} = \Gamma^{A,B}$. That is, at stage s, we find the least $x < s$ (if any) such that $x \in \overline{K}_s$ but $x \notin \Gamma^{A,B}[s]$, the approximation of $\Gamma^{A,B}$ at stage s, and define two markers a_x (the bound of the A-part) and b_x (the bound of the B-part and $b_x \in B$) and enumerate x into $\Gamma^{A,B}$ by enumerating the axiom $\langle x, A_s \upharpoonright a_x + 1, B_s \upharpoonright b_x + 1\rangle$ into Γ. If x leaves $\overline{K}$ later, we will make this axiom invalid by extracting b_x from B, or by a change (from 1 to 0) of A on $A_s \upharpoonright a_x + 1$. We must use the A-part in the definition of Γ, since otherwise B would have a complete e-degree, contradicting the N-requirements. Since A is not in our control, if A does not provide such changes, we have to extract b_x from B. This process is called the *rectification* of $\Gamma^{A,B}$ at x. If at the end of the construction, $A_s \upharpoonright a_x + 1 \subset A$ and $B_s \upharpoonright b_x + 1 \subset B$, then $x \in \Gamma^{A,B}$. Our construction will ensure that $\Gamma^{A,B}(x) = \overline{K}(x)$.

https://doi.org/10.1017/S0960129508007378 Published online by Cambridge University Press

Note that after stage s, at a stage $t > s$ say, if $x \in \overline{K}_t$ but $A_s \upharpoonright a_x + 1 \not\subseteq A_t$ or $B_s \upharpoonright b_x + 1 \not\subseteq B_t$, then in order to put x into $\Gamma^{A,B}$ again, we need to enumerate another axiom into Γ. If such a procedure happens infinitely often, x is not in $\Gamma^{A,B}$ and we cannot make $\Gamma^{A,B}(x) = \overline{K}(x)$. To avoid this, in general (but not always, as we will see soon when the N-strategies are considered), at stage t, when we re-enumerate x into $\Gamma^{A,B}$, we keep a_x the same as before but let b_x be a bigger number. We put $b_x[t]$ into B and extract $b_x[s]$ from B (we do this because we want only one axiom enumerating x into $\Gamma^{A,B}$ to be valid). *Again, this is not always true when N-strategies are considered.* The crucial point is that at any stage in the construction there is just one axiom in $\Gamma^{A,B}(x)$ enumerating x. We call this the *one axiom rule*. Now, as a_x is fixed and A is Δ^0_2, there can be only finitely many changes in $A \upharpoonright a_x + 1$, and hence we will eventually stop enumerating axioms for x into Γ.

In general, we define $\Gamma^{A,B}(x)$ as follows:

1. Choose two markers a_x (the bound of the A-part) and b_x (the bound of the B-part and $b_x \in B$) and enumerate x into $\Gamma^{A,B}$ by enumerating the axiom $\langle x, A_s \upharpoonright a_x + 1, B_s \upharpoonright b_x + 1\rangle$ at a stage s, say, into Γ.
2. Check whether A or K changes first.
   2.1. If at a stage $t > s$ we have $A_s \upharpoonright a_x + 1 \not\subseteq A_t \upharpoonright a_x + 1$, then extract $b_x[s]$ from B, and go back to Step 1, but keep a_x the same.
   2.2. If at a stage $t > s$ we have that x leaves $\overline{K}$, then extract $b_x[s]$ from B.

Because $\overline{K}$ is Π^0_1, after reaching Step 2.2, we will do nothing further. On the other hand, as explained above, because a_x is fixed and A is Δ^0_2, we can only reselect b_x (go back to Step 1) finitely often. Therefore, if x remains in $\overline{K}$, Step 2.1 can only happen at most finitely often, and after a late enough stage, x will be enumerated into $\Gamma^{A,B}$ forever, which ensures that $\Gamma^{A,B}(x) = \overline{K}(x)$. Note that the extraction of each b_x from B at Step 2.1 is done to guarantee our *one axiom rule*.

Here, a_x and b_x are chosen bigger than any a_y, b_y if $y < x$. In general, when some y leaves $\Gamma^{A,B}$, we extract b_x from B to make sure that the axiom $\langle x, A_s \upharpoonright a_x + 1, B_s \upharpoonright b_x + 1\rangle$ can never be valid again. We will make a crucial modification to this when N-strategies are considered and we need to ensure that the associated enumeration will not be injured by the construction of $\Gamma^{A,B}$.

Now we consider how to satisfy one N_Φ-requirement. Here we shall see the necessity of modifying the way of defining $\Gamma^{A,B}(x)$ described above. An N_Φ-requirement is a variant of the Friedberg–Muchnik strategy. Namely, we select x as a witness, enumerate it into E and wait for $x \in \Phi^B$. If x never enters Φ^B, then N_Φ is satisfied. Otherwise, we will extract x from E, preserving $B \upharpoonright \varphi(x)$, where $\varphi(x)$ denotes the use of x in the enumeration of $\Phi^B(x)$.

However, as the S-strategy has the highest priority, it can rectify $\Gamma^{A,B}$ at any time of the construction, which may injure this N_Φ-strategy, possibly infinitely many times. To avoid this, before choosing x, this strategy will first choose a (big) number k, the threshold of this N_Φ-strategy. That is, whenever $\overline{K}$ changes below $k+1$ (correspondingly, we may need

https://doi.org/10.1017/S0960129508007378 Published online by Cambridge University Press

to change B to rectify $\Gamma^{A,B}$, and any effort this N_Φ-strategy has made can be injured) or $\Gamma^{A,B}$ changes below $k+1$ (because of the changes in A or B below the corresponding uses), we *reset* this N_Φ-strategy by cancelling all the associated parameters except this k. Since k is fixed and A is a Δ_2^0 set, such a *resetting* process can happen at most finitely many times, so we can assume that after a sufficiently late stage this N_Φ-strategy will never be reset again.

In the construction, k can enter K, and if it does, the threshold is moved automatically to the next least number in $\overline{K}$. Since $\overline{K}$ is infinite, the threshold will stop changing its value eventually. This will be the real threshold of the N_Φ-strategy.

To preserve some initial segment of B for the diagonalisation, N_Φ will first try to move all the markers b_n for $n \geqslant k$ above the restraint. A useful A-change will facilitate this.

An N_Φ-strategy works as follows:

Setup

Define a threshold k to be a big number. Choose a witness $x > k$ and enumerate x into E.

K-Check

If an element $n \leqslant k$ leaves $\overline{K}$ or $\Gamma^{A,B}$ (b_n is extracted from B for the Γ-rectification, or some elements below the corresponding A-part use leaves A, respectively), reset this strategy, by cancelling all the associated parameters except k.

If k leaves $\overline{K}$, redefine k as the least element in $\overline{K}$ bigger than the current value of k. This will happen at most finitely many times.

Attack

1. Wait for $x \in \Phi^B$.

   While waiting, at each stage s, we check to see whether a previous guess (the last one) of A defined at a stage $t < s$, $\hat{A}_t$, is not true. That is, t is the last stage at which we were at Step 2 when a guess of A, $\hat{A}_t$, was made and a_k was requested to be defined as a new number. If the answer is yes, go to Step 3. This will be one part of the A-Check module.

2. Suppose that x enters Φ^B at stage s. Then, at this stage, we:

   — Extract $b_k[s]$ from B to prevent the enumeration of $\Phi^B(x)$ from being injured by the S-strategy.

   — Cancel all the markers a_n and b_n for $n \geqslant k$.

   — Request that a_k be defined big – bigger than any element seen so far in the construction.

   — Go back to Step 1 and, simultaneously, wait for some element $\hat{A}_s = A_s \upharpoonright a_k[s]$ to leave A.

   (*Here we are guessing that $\hat{A}_s \subseteq A$, and if our guess is wrong, the corresponding A-change will undefine $\Gamma^{A,B}(n)$ for $n \geqslant k$, and hence the further construction of $\Gamma^{A,B}$ will not change the enumeration $\Phi^B(x)$.*)

https://doi.org/10.1017/S0960129508007378 Published online by Cambridge University Press

(★) *We want to preserve B up to this $\varphi(x)$, and at this stage, we only extract $b_k[s]$ from B, not the other b_n's for $n \geqslant k$. As we will ensure that at any later stage, either $b_k[s]$ is not in B or $\hat{A}_s \nsubseteq A$, every axiom for $\Gamma^{A,B}(n), n \geqslant k$, defined between stages s_0 and s, where s_0 is the last stage we define $\gamma(k)$, is invalid forever. That is, to extract n from $\Gamma^{A,B}$, we do not need to extract the corresponding b_n's from B – extracting $b_k[s]$ is enough. This is a crucial point in our one axiom rule.*

3. Extract x from E and put $b_k[s]$ back into B. We also remove $b_n[t]$, $n \geqslant k$, $t > s$, from B, and keep these numbers out of B forever. Thus, the axioms in Γ enumerating n into $\Gamma^{A,B}$ during this period will be invalid forever. This is another crucial feature of the *one axiom rule*. Note that extracting these numbers from B will not change the enumeration $\Phi^B(x)[s]$.
From now until the next stage s' when $\hat{A}_s \subseteq A_{s'}$ (at which point we will go to Step 4), this strategy will do nothing, as $\Phi^B(x)$ is recovered and this N_Φ-strategy is satisfied (temporarily maybe). As indicated in the last paragraph, those axioms enumerated into Γ between stages s and t are invalidated because the corresponding markers b_n are removed. This A-change lifts $\gamma(n)$ for $n \geqslant k$ to numbers bigger than $\varphi_s(x)$, even though $b_k[s]$ is now back in B. So the enumeration $\Phi^B(x)$ is preserved and kept from being injured by the further construction of Γ.

4. Wait for a later stage s' such that $\hat{A}_s \subseteq A_{s'}$. So the A-change we see at Step 3 is no longer valid. Then:
   - Enumerate x into E again.
   - Extract $b_k[s]$ from B.

     Again we do this because we want to prevent the enumeration of $\Phi^B(x)$ from being injured by the S-strategy.
   - For $n \geqslant k$, if n is enumerated into $\Gamma^{A,B}$ after Step 3 (*note that a_n, b_n have new definitions in this period*), extract the corresponding b_n from B.

     As these b_n are defined as big numbers, extracting these numbers from B will not injure the enumeration $\Phi^B(x)[s]$.
   - Go back to Step 1 and, simultaneously, wait for a stage s'' with $\hat{A}_s \nsubseteq A_{s''}$ until Step 1 is reached.

   As above, we also take $b_n[t']$, $n \geqslant k$, $t' \geqslant s'$, out of B, and keep these numbers out of B forever. Thus, the axioms in Γ enumerating n into $\Gamma^{A,B}$ in this period will be invalid forever.

If after a sufficiently late stage the strategy waits at Step 1 or 3 forever, this N_Φ-requirement is obviously satisfied. In the latter case,

$$\Phi^B(x) = 1 \neq 0 = E(x),$$

and the construction of Γ will never change the enumeration of $\Phi^B(x) = 1$ since all the γ-markers are lifted to bigger numbers by the changes of A (out) below $a_k[s]$.

We show below that this strategy will not go from Step 2 or 4 back to Step 1 infinitely often, so the strategy waits at Step 1 or 3 forever, and the corresponding N_Φ-requirement is satisfied.

If this were not the case, this strategy would go through Step 2 infinitely often, because A is assumed to be Δ_2^0 and for a fixed s, $a_k[s]$ is fixed. In the construction, for a fixed $a_k[s]$, the A-changes below $a_k[s]$ can happen only finitely often, giving us chances to go to Step 4 finitely often. We prove now that A is c.e. Let t_i be the stages at which Step 2 is reached. Then at each stage t_i, we know that $\hat{A}_{t_i} \subset A$, because otherwise, one element in $\hat{A}_{t_i}$ but not in A will allow us to go to Step 4 and stop there (N_Φ is satisfied by a diagonalisation) forever. By this property, A is computably enumerable because for each x, x is in A if and only if x is in $\hat{A}_{t_i}$ for some i. This contradicts our assumption on A.

Note that according to the actions at Step 2, for any $i < j$, we have $a_k[t_i] < a_k[t_j]$. It is possible that at a stage later than t_j, there is a number $m < a_k[t_i]$, and hence $m < a_k[t_j]$, leaving A, which gives us choices to put $b_k[t_i]$ or $b_k[t_j]$ back into B to realise a diagonalisation. In the construction, we always put $b_k[t_j]$, the bigger one, into B to recover the enumeration $\Phi^B(x)$ to $\Phi^B(x)[t_j]$ in order to diagonalise. If m never comes back to A, then N_Φ is satisfied.

We now show that this N_Φ-strategy is consistent with the definition of $\Gamma^{A,B}$. Again, k is the threshold of this strategy. First note that this N_Φ-strategy does not affect the definition of $\Gamma^{A,B}(n)$ when $n < k$. For $n \geqslant k$, this N_Φ-strategy can invalidate the axiom for n in $\Gamma^{A,B}$ at most finitely often at Step 2 or 4, and, eventually, after a_k has settled down (after which we will not go through Step 2), A can change below a_k at most finitely often (after which we will not go through Step 4), and, finally, once a new axiom enumerating n into $\Gamma^{A,B}$ is enumerated into Γ, it will not be invalidated by this N_Φ-strategy again.

3. Cupping by 1-generic degrees

In this section we prove Theorem 1.2. That is, given a non-c.e. Δ_2^0 set A, we will construct a Δ_2^0 1-generic set B and an enumeration operator Γ satisfying the following requirements:

$$S \ : \ \Gamma^{A,B} = \overline{K}$$
$$G_i \ : \ (\exists \lambda \subset B)[\lambda \in W_i \ \text{ or } \ (\forall \mu \supseteq \lambda)(\mu \notin W_i)]\,.$$

If all the requirements G_i, together with the global requirement S, are satisfied, then B will have the intended properties. It is well known that the e-degree of a 1-generic set cannot be complete.

The strategy for satisfying S is the same as that described in the last section. The idea of satisfying one G_i-requirement is an easy full-approximation argument: we select a string λ first, and wait for an extension of λ to enter W_i. While we are waiting for such an extension to appear, we let B extend λ, and once we find an extension μ of λ appearing in W_i, we let B extend μ. Here, letting B extend μ means that for any x, we put x into B if $\mu(x) = 1$, and keep x out of B if $\mu(x) = 0$. As we also need to code $\overline{K}$ into $A \oplus B$, the construction of Γ may prevent B from extending μ. Because of this, we call μ a target and λ a pretarget of G_i. We can either satisfy G_i through a pretarget λ ($\lambda \subset B$ and no extension of λ is in W_i) or a target μ ($\mu \subset B$ and μ is in W_i).

Now we consider the interaction between the S-strategy and a G_i-strategy. As mentioned above, it may happen that after we select a pretarget λ or a target μ, the construction of Γ will not allow B to extend it because S has the highest priority. We get around this difficulty by applying the threshold strategy. That is, we set k as the threshold first, and if $\overline{K}$ changes below $k+1$ or some $n \leqslant k$ leaves $\Gamma^{A,B}$ (the conflict situation described above can happen), we reset this G_i-strategy by giving up the selected pretargets and targets, and then select new ones that are consistent with the current construction of $\Gamma^{A,B} \upharpoonright (k+1)$. As k is fixed and A is Δ^0_2, this G_i-strategy can be reset only finitely often.

After we choose k, we wait for $\gamma(k)$ to be defined (that is, a_k and b_k are both defined), and then select a pretarget λ (we will make B extend λ from now on) with $\lambda(b_k) = 1$. If later we find a $\mu \supseteq \lambda$ in W_i, instead of making B extend μ immediately, we define $\hat{\mu}$ as a string the same as μ, except that $\hat{\mu}(b_k) = 0$, and make the current approximation of B extend $\hat{\mu}$. (*So b_k is taken out of B, which extracts all the numbers enumerated into $\Gamma^{A,B}$ from the stage λ is selected. We do this mainly because we want S to be happy with G_i's actions.*) We actually want B to extend μ, so once A changes below a_k (from 1 to 0), we can re-enumerate b_k (and nothing else, as in this way we make B extend $\hat{\mu}$) into B to satisfy G_i. If there are no A-changes below a_k, or a change in A is not permanent, we will ensure that b_k is out of B, so B extends $\hat{\mu}$. If $A \upharpoonright a_k$ recovers to its initial value (or we have not observed a change in $A \upharpoonright a_k$ at all), we select another pretarget λ', which is an extension of $\hat{\mu}$, and work on λ' in a similar way.

Again, to satisfy the S-requirement, the construction of Γ follows the *one axiom rule*, as explained in Section 2.

We are now ready to give the full construction of B. The construction will proceed on a binary tree T, where each node α on T is a G_i-strategy with $i = |\alpha|$. Each α works to satisfy $G_{|\alpha|}$ and has the following related parameters:

— k_α, the threshold of α;
— λ_α, a pretarget of α;
— μ_α, a target of α;
— $\hat{\mu}_\alpha$, a variant of μ_α;
— U_α, the collection of α's guesses of A, $\hat{A}_s, s \in \omega$.

Each node has two outcomes, 0, 1, with $0 <_L 1$.

As S is a global requirement, we do not put it on T.

Construction

The construction is as follows:

Stage 0**:**

Let $B = \varnothing$, $\Gamma = \varnothing$, $U_\alpha = \varnothing$ for all α, and let all the thresholds, pretargets and targets be undefined.

Stage $s+1$**:**

At stage $s+1$, we first perform two checks: the K-Check and the A-Check.

https://doi.org/10.1017/S0960129508007378 Published online by Cambridge University Press

K**-Check**

Suppose k_s leaves $\overline{K}$ at this stage. We determine which strategy is reset by this $\overline{K}$-change. Find a strategy α with the highest priority such that $k_\alpha \geqslant k_s$. Reset α by cancelling all the parameters of α, except the threshold k_α. *If $k_\alpha \geqslant k_s$, redefine k_α as the next number in $\overline{K}_{s+1}$. After this resetting, U_α becomes empty.* Initialise all the strategies with lower priority. If there is such a strategy, do nothing.

A**-Check**

Let a^s be the number such that $A(a^s)$ changes at stage $s+1$.

Find a strategy α with the highest priority such that $a^s < a_{k_\alpha}[s]$ and initialise all the strategies with lower priority. *If there is no such α, then for any n with $a^s < a_n[s]$, we extract $b_n[s]$ from B to remove n from $\Gamma^{A,B}$.* We assume in the following that there is such an α.
For $n > k_\alpha$, remove $b_n[s]$ from B to ensure that n is extracted from $\Gamma^{A,B}$.
If there is an $m < k_\alpha$ with $a^s < a_m[s]$, then reset α by cancelling all parameters of α except k_α. Again we remove $b_m[s]$ from B to ensure that m is extracted from $\Gamma^{A,B}$. (*When m is enumerated into $\Gamma^{A,B}$ again, we define a_m as $a_m[s]$. As A is Δ_2^0 and a_m is fixed, such a resetting procedure can happen at most finitely often.*)
Otherwise, extract $b_{k_\alpha}[s]$ from B to remove k_α from $\Gamma^{A,B}$. Consider the following two cases:

Case 1: a^s enters A.

If $\hat{A}_t \subseteq A_{s+1}$, where $\hat{A}_t$ is the largest one in U_α, we also extract the associated $b_{k_\alpha}[t]$ from B. *Note that between stages t and $s+1$, $b_{k_\alpha}[t]$ has been enumerated into B at some point (the last one), since $\hat{A}_t$ is not a subset of A from that point on. We have α is satisfied, temporarily, until stage $s+1$.* We also require that the new pretarget λ_α be defined as an extension of B_{s+1}. *In particular, as $b_{k_\alpha}[t]$ is removed, when we define λ_α again, we will define $\lambda_\alpha(b_{k_\alpha}[t])$ as* 0. We also require that both a_{k_α} and b_{k_α} be defined bigger.

If $\hat{A}_t \nsubseteq A_{s+1}$, we keep $b_{k_\alpha}[t]$ in B to make sure that α is still satisfied. In this case, as $b_{k_\alpha}[s]$ (if any) is removed from B, and will be kept out of B, the recent axiom enumerating k_α into $\Gamma^{A,B}$ is invalidated forever. We require that only b_{k_α} be defined bigger.

Case 2: a^s leaves A.

Check whether U_α contains a guess $\hat{A}_t$, the largest one such that $\hat{A}_t \subseteq A_s$ and that $a^s \in \hat{A}_t$.
If there is none, do nothing, but require that only b_{k_α} be defined bigger.
Otherwise, we know that $\hat{A}_t \nsubseteq A_{s+1}$ (because of a^s), and we enumerate the related $b_{k_\alpha}[t]$ into B, and extract the associated x_α from E. Then declare that α is satisfied until the next stage when $\hat{A}_t$ is contained in A again. *In this case, $b_{k_\alpha}[t]$ is enumerated into B and $b_{k_\alpha}[s]$ is removed from B (without loss of generality, we assume that they are different) at this stage, and $b_{k_\alpha}[s]$ will be kept outside B forever.* Again, we require that only b_{k_α} be defined bigger.

Here, we try to approximate A using a computable sequence $\{\hat{A}_t : t \in \omega\}$, where each $\hat{A}_t$ is defined at stage t as the set of elements already in A currently, and for any $t < t'$, if both $\hat{A}_t, \hat{A}_{t'}$ are defined, then $\hat{A}_t \subset \hat{A}_{t'}$. As A is Δ^0_2, if we can define $\hat{A}_t$ infinitely often, then $\bigcup_{t\in\omega} \hat{A}_t = A$, and A is computably enumerable, which is impossible.

After these two checks, we rectify $\Gamma^{A,B}$ as follows:

$\Gamma^{A,B}$-rectification module

Check for all elements $n < s$ to see whether there is some n such that $\Gamma^{A,B}(n) \neq \overline{K}(n)$. If there is no such n, do nothing. Otherwise, perform the following actions for the least such n:

$n \in \overline{K}$.

If n has not been enumerated into $\Gamma^{A,B}$ before (so a_n and b_n have not been defined), define both a_n and b_n as big numbers. Otherwise, let $s^- < s$ be the last stage when n is in $\Gamma^{A,B}$. *If both a_n and b_n are required to be defined as big numbers in the A-Check part, or n is a threshold of some G-strategy, and this G-strategy requires that a_n be redefined as a big number, we define a_n and b_n big. Otherwise, leave a_n the same as before and define b_n as a big number.*

In all cases, enumerate b_n into B and the axiom $\langle n, A_{s+1} \upharpoonright a_n + 1, \{b_m | m \leqslant n\}\rangle$ into Γ.

$n \notin \overline{K}$.

Find the valid axiom in Γ for n (if any), $\langle n, A \upharpoonright a_n + 1, M_n\rangle$ say, and extract the largest element of M_n from B. *This action invalidates the axiom $\langle n, A \upharpoonright a_n+1, M_n\rangle$, and n is removed from $\Gamma^{A,B}$.*

Now we construct a path through the tree T, δ_{s+1}, of length $\leqslant s$, as the approximation of the true path f at stage $s+1$. Each node $\alpha \subseteq \delta_{s+1}$ is said to be visited at stage $s+1$.

Construction of δ_{s+1}

We define $\delta_{s+1}(n)$ for $n < s+1$ by induction on n. When $n = s+1$, we stop stage $s+1$ and go to the next stage. Suppose $\delta_{s+1}(i-1)$ is defined. We let $\delta_{s+1} \upharpoonright i$ be α, a strategy working on the requirement G_i. When α is visited for the first time after being initialised, it starts from *Setup* to define k_α, the threshold of α, define $\delta_{s+1} = \alpha$, and go to the next stage. Otherwise, we go to the check part.

Setup

If a threshold k_α has not been defined or is cancelled, define it as a big number – bigger than any element that has appeared so far in the construction.

Attack

1. If $\gamma_s(k_\alpha)$ has not been defined yet, let $\delta_{s+1} = \alpha$. Go to the next stage. *We wait for $\gamma_s(k_\alpha)$ to be defined, which will be done at some later stage.*
2. If α is declared to be satisfied at (the most recent) stage $t < s$ through target μ_t, and $\hat{A}_t \not\subseteq A_{s+1}$, let the outcome of α be 0 *(satisfied)*. Go to substage $i+1$.

https://doi.org/10.1017/S0960129508007378 Published online by Cambridge University Press

3. If Cases 1 and 2 do not apply and the pretarget λ_α needs to be defined or redefined, define it as $B_{s+1} \upharpoonright b_{k_\alpha} + 1$. (*Note that* $\lambda_\alpha(b_{k_\alpha}) = 1$ *as* b_{k_α} *is currently in* B.) Let $\delta_{s+1} = \alpha$. Initialise all the strategies of lower priority and go to the next stage. *From now on, we will search for a target in* W_i *extending* λ_α.
4. If λ_α is defined and there is no $\mu \supseteq \lambda_\alpha$ in $W_{i,s+1}$ (so μ_α is not defined), let the outcome of α be 1 *(waiting)*. Go to substage $i+1$.
5. If λ_α is defined, μ_α is not defined and there is some $\mu \supseteq \lambda_\alpha$ in $W_{i,s+1}$, choose the least such μ, denote it by μ_α, and undefine λ_α. Enumerate in the guess list U_α a new guess $\hat{A}_s = A_s \upharpoonright (a_{k_\alpha} + 1)[s+1]$ as an approximation of $A \upharpoonright a_{k_\alpha}[s+1]$ at stage $s+1$. Extract $b_{k_\alpha}[s]$ from B. (*This extraction removes* n *from* $\Gamma^{A,B}$ *for each* $n \geqslant k_\alpha$.) *We have found a* μ_α *in* W_i *extending* λ_α, *and* μ_α *is our target, as we want* $\mu_\alpha \subset B$. *We need* α *to cooperate with the* S*-strategy, and if* $\hat{A}_s \not\subset A$, *we can make* B *extend* μ *by enumerating* $b_{k_\alpha}[s]$ *into* B *again.*
   Let $\hat{\mu}$ be a string that is the same as μ except at position $b_{k_\alpha}[s]$, where we have $\hat{\mu}(b_{k_\alpha}[s]) = 0$. Let $\hat{\mu} \subseteq B_{s+1}$. When α defines λ_α again, α defines it as an extension of $\hat{\mu}$. *Here, when we say that* B *extends* $\hat{\mu}$, *we mean that* $b_{k_\alpha}[s]$ *is moved from* B. *If later, after we see* $\hat{A}_s \not\subset A$, *we put* $b_{k_\alpha}[s]$ *into* B, *then* B *extends this* μ_α *immediately, which will satisfy* G_i.
   λ_α *is undefined, so we will choose another pretarget later, extending* B_{s+1}.
   Cancel all the markers a_n and b_n for $n \geqslant k_\alpha$, and request a_{k_α} and b_{k_α} be defined as big numbers. *We can do so because* $b_{k_\alpha}[s]$ *is removed from* B.
   Let $\delta_{s+1} = \alpha$. Initialise all the strategies of lower priority and go to the next stage.

This completes the construction of B.

Verification

We now verify that the B we have just constructed satisfies all the requirements. Define the true path $f \subset T$ as the limit of δ_s, $s \in \omega$. That is,

$$\forall n \exists s_n \forall s > s_n (f \upharpoonright n \subseteq \delta_s).$$

The following lemma ensures that f is well defined and infinite.

Lemma 3.1. For each n, let t_n be the last stage at which $f \upharpoonright n$ is initialised. The following are true:

(0) t_n exists.

(1) There is a stage $t_1(n) \geqslant t_n$ after which $f \upharpoonright n$ cannot be reset again. In particular, after stage $t_1(n)$, we will not cancel $U_{f \upharpoonright n}$ again.

(2) There is a stage $t_2(n) \geqslant t_1(n)$ after which $f \upharpoonright n$ does not act according to the A-changes in the A-Check part or in the *Attack* part. In particular, after stage $t_2(n)$, $f \upharpoonright n$ does not initialise lower priority strategies and does not affect the definition of Γ.

(3) $f(n)$ is defined. That is, after a stage $t_3(n) \geqslant t_2(n)$, for any s, we have $f \upharpoonright n^\frown \mathcal{O} \subseteq \delta_s$, where $\mathcal{O}$ is the true outcome of $f \upharpoonright n$.

https://doi.org/10.1017/S0960129508007378 Published online by Cambridge University Press

Proof. We prove the lemma by induction on n. Suppose that (0)–(3) are all true for $m < n$. We will prove that (0)–(3) are also true for n. Let $f \upharpoonright n = \alpha$.

It is easy to see that t_n exists from the assumption that (0)–(3) are true for $f \upharpoonright (n-1)$. In particular, by (3), after a stage large enough, whenever $f \upharpoonright (n-1)$ is visited, $f \upharpoonright n$ is also visited, and hence the construction never goes to the left of $f \upharpoonright n$. Thus (0) is true for n.

Let s_0 be the first α-stage after t_n. At this stage, α defines k_α, and will never redefine it again, modulo finitely many times of shifting. That is, if the current k_α leaves $\overline{K}$, we take the next element in $\overline{K}$ to be k_α automatically. As $\overline{K}$ is infinite, such a shifting process can happen at most finitely often. Let $s_1 \geqslant s_0$ be the last stage after which such a shifting never happens again.

We now show that α can be reset at most finitely often. Note that in the construction, only A-changes and $\overline{K}$-changes can reset α. First, after stage s_1, because k_α is fixed, α can be reset by the $\overline{K}$-changes only finitely often. Thus, there is a stage $s_2 \geqslant s_1$ after which $\overline{K}$ never changes below this k_α, and hence, α will never be reset by the $\overline{K}$-changes again.

As U_α can only be cancelled when α is initialised or reset by the $\overline{K}$-changes, we know that U_α can never be cancelled after stage s_2.

Since A is Δ^0_2, in order to prove that α can be reset by the A-changes only finitely often, we only need to prove by induction that after a sufficently late stage $s_3 \geqslant s_2$, for any $j < k_\alpha$, if $\gamma(j)$ is defined, then $a_j[s_3]$ is fixed, which is obviously true. Let $t_1(n) = s_3$. Then after stage $t_1(n)$ no A-changes can reset α again (*correspondingly, no b_j with $j < k_\alpha$ will be extracted from B to rectify $\Gamma^{A,B}$, to ensure that the construction of $\Gamma^{A,B}$ follows the one axiom rule*).

As U_α can only be cancelled when α is initialised or reset, we know that U_α can never be cancelled after stage t_1. Thus (1) is true for n.

We now prove (2). By the choice of s_3 above, we assume that for each $j < k_\alpha$, a_j is fixed, and A does not change below a_j again.

In order to show a contradiction, suppose that α acts infinitely many times. As A is Δ^0_2, for a particular $a_{k_\alpha}[s]$, A can change below it at most finitely often. Therefore, α reaches Case 5 in the *Attack* part infinitely often, and each time a guess $\hat{A}_s (= A_s \upharpoonright a_{k_\alpha}[s] + 1)$ of A for some s is put into U_α, and a_{k_α} is required to be defined bigger. Let

$$s'_1 < s'_2 < \cdots < s'_m < \cdots$$

be the list of these stages after stage s_3 at which α reaches Case 5 through $a_{k_\alpha}[s'_i]$, respectively.

We claim that for each i, $\hat{A}_{s'_i} \subset A$. Suppose this were not the case, and let y be in $\hat{A}_{s'_i}$, but not in A. Then this y is less than $a_{k_\alpha}[s'_i] + 1$, and can provide chances for α to do an action in the A-Check part, and, eventually, α will be satisfied forever through this y (shown in the next paragraph), and α will do no more actions in the construction, which contradicts our assumption.

We now show that this y enables us to satisfy α. Note that at stage s'_i, at Case 5, a target string μ_α is found in W_n, and $b_{k_\alpha}[s'_i]$ is extracted from B to remove all $l \geqslant k_\alpha$ from $\Gamma^{A,B}$. The process is that if before stage s'_{i+1}, some element, z (which can be different from y, or the same as y), in $\hat{A}_{s_i}$ leaves A, then $b_{k_\alpha}[s'_i]$ is put into B to make B extend μ_α. By our assumption, if s'_{i+1} exists, this z enters A again between s'_i and s'_{i+1}, and, as a consequence,

https://doi.org/10.1017/S0960129508007378 Published online by Cambridge University Press

$b_{k_\alpha}[s_i']$ is removed from B. As A is Δ_2^0, and we assume that y is not in A, there is a biggest j such that y is in $\hat{A}_{s'_{i+j}}$, and y leaves A after stage s'_{i+j}. As y is also in $\hat{A}_{s'_{i+j}}$ and y is less than $b_{k_\alpha}[s'_{i+j}]$, we get that $b_{k_\alpha}[s'_{i+j}]$ is enumerated into B to make B extend the current μ_α, and α is satisfied. After this, y remains out of A, and hence $\hat{A}_{s'_{i+j}} \not\subseteq A$ afterwards, and α is satisfied forever.

We need to mention here that the $\Gamma^{A,B}$-rectification procedure will not change B on $|\mu_\alpha|$. This is because at stage s'_{i+j}, only $b_{k_\alpha}[s'_{i+j}]$ is moved from B, and nothing else, and further construction of B extends $\hat{\mu}_\alpha$ until $\hat{A}_{s'_{i+j}} \not\subseteq A$ is found, and $b_{k_\alpha}[s'_{i+j}]$ is put back into B, so B extends μ_α. Also note that all the lower priority strategies are initialised at this stage, and the strings they select later will be extensions of μ_α, or $\hat{\mu}_\alpha$. This ensures that we can make B extend μ_α whenever we can. Note that in the construction we always ensure that either $\hat{A}_{s'_{i+j}} \not\subseteq A$ or $b_{k_\alpha}[s'_{i+j}]$ is not in B, so there is no conflict between α and the $\Gamma^{A,B}$-rectification procedure.

From the claim above, we can conclude that A is computably enumerable as follows: for each x, x is in A if and only if x is in $\hat{A}_{s_i'}$ for some i. This contradicts our assumption that A is not computably enumerable.

Therefore, α can act at most finitely often and (2) is true for α. Let $t_2(n)$ be the last stage at which α acts.

Now we can see that after $t_2(n)$, the *Attack* part is always at Case 2 (satisfied) or Case 4 (waiting). Correspondingly, α will always have outcome 0 or 1, respectively. If α remains at Case 4, then (the most recent version of) λ_α has no string μ in W_n extending λ (otherwise, we could have one more action later in Case 5, which is impossible by our assumption) and B extends λ_α in this case. If α stops at Case 2, α is satisfied through μ_α because μ_α is in W_n and B extends μ_α.

Let $\mathcal{O}$ be the true outcome of α. Then after stage $t_2(n)$, whenever α is visited, $\alpha^\frown\mathcal{O}$ is also visited, so (3) is proved.

This completes the proof of the lemma. □

Lemma 3.1 immediately gives us the following lemma.

Lemma 3.2. The true path f is well defined and infinite.

Now we prove that every G-requirement is satisfied.

Lemma 3.3. Every G_i-requirement is satisfied.

Proof. Fix i and let α be a G_i-strategy on f. By Lemma 3.1, there is a late enough stage, t, after which α cannot be initialised or reset again, so α will not act again in the remainder of the construction. Also, we can assume that α has true outcome $\mathcal{O}$, and after stage t, each stage is an $\alpha^\frown\mathcal{O}$-stage.

There are two cases:

— $\mathcal{O}$ is 1.

https://doi.org/10.1017/S0960129508007378 Published online by Cambridge University Press

After stage t in the construction, α is always in Case 4, which means that λ_α (the last version) has no extension in W_i since otherwise if a string extending λ_α appears in W_i after stage t, then α will be in Case 5 and α will initialise all strategies with lower priority, contradicting our choice of t, and if a string extending λ_α enters W_i before stage t, then according to the construction, λ_α should have been redefined as another string. Note that in this case, B extends λ_α, and G_i is satisfied.

— $\mathcal{O}$ is 0.

After stage t, α is always in Case 2, which means that μ_α is in W_i and that B extends μ_α, and again G_i is satisfied. □

The next lemma states that the S-requirement is satisfied.

Lemma 3.4. The S requirement is satisfied.

Proof. We need to prove that for each n, we have $\Gamma^{A,B}(n) = \overline{K}(n)$. Fix n.

Now find the G_n strategy α on the true path f, and let s be the last stage on which α acts. By Lemma 3.1, s exists, so we know $n < k_\alpha$ and:

(a) $\overline{K}_s \upharpoonright k_\alpha = \overline{K} \upharpoonright k_\alpha$.
(b) At stage s, for any $m < k_\alpha$, $m \in \overline{K}_s$ if and only if $m \in \Gamma^{A,B}[s]$.
(c) no $m < k_\alpha$ leaves $\Gamma^{A,B}$ after stage s.

If (a) were not true, α would be reset later, which, by our choice of s, cannot happen, so (a) is true. (b) and (c) are true because at stage s, $\gamma(k_\alpha)$ is defined and our $\Gamma^{A,B}$-rectification procedure ensures that for $l < k_\alpha$, if l is in $\overline{K}$, then l is also in $\Gamma^{A,B}[s]$. On the other hand, if l is not in $\overline{K}$, suppose that l leaves $\overline{K}$ at stage t, and, without loss of generality, suppose that l is in $\Gamma^{A,B}[t]$. Then, at the $\Gamma^{A,B}$-rectification part of stage t, $b_l[t]$ is extracted from B to remove l from $\Gamma^{A,B}$, and after stage t, l can never be enumerated into $\Gamma^{A,B}$ again. Note that after stage s, the $\Gamma^{A,B}$-rectification procedure will never define $\gamma(m)$ for those $m < k_\alpha$ again.

Now, as $n < k_\alpha$, from (a), (b) and (c), we know that

$$\Gamma^{A,B}(n) = \Gamma^{A,B}(n)[s] = \overline{K}_s(n) = \overline{K}(n),$$

which is the equality we want. □

Finally, we prove that the constructed B is a Δ^0_2 set, which completes the proof of Theorem 1.2.

Lemma 3.5. B is Δ^0_2.

Proof. We need to show that for each n, n can be enumerated and extracted from B at most finitely times. To see this, fix n, and again, as in the previous lemma, we consider the G_n-strategy α on the true path f. Let s be the last stage at which α acts. Then λ_α has length greater than n, and after stage s, B will never change on λ_α, and hence will not change on n. This means that $B(n)$ changes at most s times. □

This completes the proof of Theorem 1.2.

https://doi.org/10.1017/S0960129508007378 Published online by Cambridge University Press

4. Cupping by 3-c.e. degrees

In this section we give a proof of Theorem 1.3. Suppose that we are given an ω-c.e. set A that is not computably enumerable and has a change-bounding function g. We will modify the construction of B given in Section 2, to make it 3-c.e.. The following requirements will be satisfied:

$$S : \Gamma^{A,B} = \overline{K}$$
$$N_\Phi : E \neq \Phi^B.$$

We have explained how to satisfy the S-requirement in detail in the previous two sections. In particular, we have seen how to ensure that the constructed $\Gamma^{A,B}$ satisfies the one axiom rule. As we now want to make B 3-c.e., the construction of $\Gamma^{A,B}$ in this section will contain some new features. Again, for a fixed n, if we want to enumerate n into $\Gamma^{A,B}$, we will have one A-marker a_n, but instead of having just one B-marker b_n, we will have a block of B-markers B_n of size h_n, where $h_n = \sum_{x<a_n} g(x) + 1$, together with a counter c_n, which is a parameter telling us which element in this block can be extracted if needed. When $\gamma(n)$ is defined, we enumerate all elements in B_n into B. We can extract n from $\Gamma^{A,B}$ by extracting just one of B_n. We use b_{c_n} to denote the element in B_n at which c_n is pointing. In the construction, whenever we extract b_{c_n} from B_n, we also decrease c_n by 1, indicating that if we need to extract a number from B_n again, we will extract the next available number, which will be less than the previous one. As A is ω-c.e., and g is the change-bounding function, the size of B_n is large enough, and we can never run out of elements of B_n. This ensures that every element of B_n can be extracted from B at most once in the construction to satisfy N_Φ. Note that after being extracted from B, b_{c_n} can be enumerated into B again (the second time) when requested by the same N_Φ-strategy.

An N-strategy works as follows:

Setup

Define a threshold k to be a big number. Choose a witness $x > k$ and enumerate x into E.

Again, whenever an element $n \leqslant k$ leaves $\overline{K}$ or $\Gamma^{A,B}$ (b_n is extracted from B for the Γ-rectification or A changes below the corresponding A-part use, respectively), reset this strategy by cancelling all the associated parameters except k, and if k leaves $\overline{K}$, redefine k as the least element in $\overline{K}$ bigger than the current value of k.

Attack

1. Wait for $x \in \Phi^B$.

   While waiting at each stage s we check to see whether a previous enumeration guess (the last one) of A, $\hat{A}_t \in U_\alpha$, was defined at a stage $t < s$ such that $\hat{A}_t \not\subseteq A_s$. If it was, go to Step 3.

2. Suppose that x enters Φ^B at stage s. Also suppose that at this stage k is in $\Gamma^{A,B}$ with a_k, B_k and c_k defined. Then, at this stage we extract $b_{c_k}[s]$ from B, to remove $n \geqslant k$ from $\Gamma^{A,B}$. Also, cancel all the markers a_n and B_n for $n \geqslant k$, and request that a_k and elements of B_k be defined big (of course, a_n and elements of B_n, $n > k$, are also

automatically defined to be big.). Go back to Step 1 and, simultaneously, wait for $\hat{A}_s \nsubseteq A$.

Again, the corresponding A-change will allow us to invalidate the current axiom for n in $\Gamma^{A,B}(n)$ *and lift its marker block* B_n, $n \geqslant k$, *so that the further construction of* $\Gamma^{A,B}$ *will not change the enumeration* $\Phi^B(x) = 1$. *Here we want to preserve B up to this* $\varphi(x)$, *and at this stage we only extract* $b_{c_k}[s]$ *from B, and not the other* b_n*'s for* $n \geqslant k$. *This is to ensure that our N-strategy is consistent with the* $\Gamma^{A,B}$*-construction. Again, the one axiom rule is a crucial point.*

3. Extract x from E and put $b_{c_k}[s]$ back into B. We also take elements in $B_n[t]$, $n \geqslant k$, $t > s$, out of B if these elements have not been extracted from B already, and keep these numbers out of B forever to satisfy the one axiom rule. We do this to invalidate the axioms in Γ, which enumerates n into $\Gamma^{A,B}$ during this period. Note that extracting these numbers from B will not change the enumeration $\Phi^B(x)[s]$, and putting $b_{c_k}[s]$ back into B recovers $\Phi^B(x)$ to $\Phi^B(x)[s]$.

   Decrease c_k *by one.*

   From now until the next stage s' *when* $\hat{A}_s \subseteq A_{s'}$ *(we will go to Step 4 when this is the case), this strategy will do nothing since* $\Phi^B(x)$ *is recovered and this* N_Φ*-strategy is satisfied (temporarily maybe).*

4. Wait for a later stage s' such that $\hat{A}_s \subseteq A_{s'}$. So the A-change we saw at Step 3 is no longer there. We now do as follows:

   — Enumerate x into E again.

   — Extract $b_{c_k}[s']$ from B. *Again, this extraction is to prevent the enumeration of* $\Phi^B(x)$ *from being injured by the S-strategy. Note that* $b_{c_k}[s]$ *and* $b_{c_k}[s']$ *are different.*

   — For $n \geqslant k$, if n is enumerated into $\Gamma^{A,B}$ after Step 3 (*note that* a_n, B_n *have new definitions in this period*), extract elements in B_n from B, provided these elements have not been extracted from B already. *As these elements are defined as big numbers, extracting them from B will not injure the enumeration* $\Phi^B(x)[s]$.

   — Request that a_k and elements in B_k be defined as big numbers.

   — Go back to Step 1 and, simultaneously, wait for $\hat{A}_s \nsubseteq A$ until Step 2 is reached.

This N_Φ-requirement is obviously satisfied if after a late enough stage the strategy waits at Step 1 or 3 forever. In the latter case,

$$\Phi^B(x) = 1 \neq 0 = E(x),$$

and the construction of $\Gamma^{A,B}$ will never change the enumeration of $\Phi^B(x) = 1$ since all the γ-markers are lifted to bigger numbers by the changes of A (out) below $a_k[s]$.

As in Section 2, we can show that this strategy will not go from Step 2 or 4 back to Step 1 infinitely often. Therefore, the strategy waits at Step 1 or 3 forever, and the corresponding N_Φ-requirement is satisfied.

We now show that this N_Φ-strategy is consistent with the definition of $\Gamma^{A,B}$. Again, k is the threshold of this strategy. First note that this N_Φ-strategy does not affect the definition of $\Gamma^{A,B}(n)$ when $n < k$. This N_Φ-strategy can undefine $\Gamma^{A,B}(n)$ for $n \geqslant k$ at most

https://doi.org/10.1017/S0960129508007378 Published online by Cambridge University Press

finitely often at Step 2 or Step 4, and, eventually, after a_k has settled down (after which we will not reach Step 2 again), A can change below a_k at most finitely often (after which we will not reach Step 4), and, finally, once $\Gamma^{A,B}(k)$ is defined, it will not be undefined by this N_Φ-strategy again.

The crucial point here is that the block B_k contains enough elements for us to extract at Step 4, because we know the change-bounding function g in advance. It is possible that after Step 4 we define a_k and B_k afresh (so the elements in B_k are bigger than the use of the enumeration we see at Step 2) and enumerate k into $\Gamma^{A,B}$ (the numbers in this new B_k are enumerated into B), and later we have an A-change (out) below $a_k[s]$. We go to Step 4 by enumerating $b_{c_k}[s]$ into B to recover the enumeration. If so, as specified at Step 4, we also remove elements from the new B_k forever to make sure that these axioms can never enumerate k into $\Gamma^{A,B}$. If k is enumerated into $\Gamma^{A,B}$, it should be enumerated by other (new) axioms. Again, it is a significant point of our *one axiom rule*.

We now describe the construction of Γ and B. As in Section 2, the construction proceeds on a binary tree, and each node α is a strategy working on the N_{Φ_i}-requirement where $i = |\alpha|$. Parameters k_α, U_α, $\hat{A}_i$ are exactly the same as those in Section 3. Some modifications are made to make B 3-c.e.

Construction

Stage 0**:**

Let $B = \emptyset$, $\Gamma = \emptyset$, $U_\alpha = \emptyset$ for all α, and let all the thresholds be undefined.

Stage $s+1$**:**

At stage $s+1$, we first perform two checks: the K-Check and A-Check.

K-Check

Suppose that k_s leaves $\overline{K}$ at this stage and determine which strategy is reset by this $\overline{K}$-change. Find a strategy α with the highest priority such that $k_\alpha \geqslant k_s$. Reset α by cancelling all the parameters of α, except the threshold k_α. *If $k_\alpha \geqslant k_s$, then also redefine k_α as the next number in $\overline{K}_{s+1}$. After this resetting, U_α becomes empty.* Initialise all the strategies with lower priority. If there is no such a strategy, do nothing.

A-Check

Let a^s be the number such that $A(a^s)$ changes at stage $s+1$.

Find a strategy α with the highest priority such that $a^s < a_{k_\alpha}[s]$ and initialise all the strategies with lower priority. *If there is no such α, then for any n with $a^s < a_n[s]$, we extract all elements in $B_n[s]$ from B, provided these elements have not been extracted from B already, in order to remove n from $\Gamma^{A,B}$.* We assume below that such an α exists.

For $n > k_\alpha$, remove elements in $B_n[s]$ from B, provided these elements have not been extracted from B already, to ensure that n is extracted from $\Gamma^{A,B}$.

If there is an $m < k_\alpha$ with $a^s < a_m[s]$, reset α by cancelling all the parameters of α except k_α. Again, we remove elements of $B_m[s]$ from B, if these elements have not been extracted from B already, to ensure that m is extracted from $\Gamma^{A,B}$. (*When m*

is enumerated into $\Gamma^{A,B}$ *again, we define* a_m *as* $a_m[s]$, *but with* B_m *new. As* A *is* Δ^0_2 *and* a_m *is fixed, such a resetting procedure can happen at most finitely often.*)
Otherwise, extract $b_{c_{k_\alpha}}[s]$ from B to remove k_α from $\Gamma^{A,B}$. Consider the following two cases:

Case 1: a^s enters A.

If $\hat{A}_t \subseteq A_{s+1}$, where $\hat{A}_t$ is the largest one in U_α, we also extract the associated $b_{c_{k_\alpha}}[t]$ from B, and put the corresponding x_α back into E. *Note that* $b_{c_{k_\alpha}}[t]$ *has been enumerated into* B *at some point (the last one) between stages* t *and* $s+1$ *since* $\hat{A}_t$ *is not a subset of* A *from that point on. We have* α *is satisfied, temporarily, until stage* $s+1$. We also require that both a_{k_α} and B_{k_α} be defined bigger.

If $\hat{A}_t \not\subseteq A_{s+1}$, *we keep* $b_{c_{k_\alpha}}[t]$ *in* B *to make sure that* α *is still satisfied. In this case, as* $b_{c_{k_\alpha}}[s]$ *(if any) is removed from* B *and will be kept out of* B, *the recent axiom enumerating* k_α *into* $\Gamma^{A,B}$ *is invalidated forever.* We require that only B_{k_α} be defined bigger.

Case 2: a^s leaves A.

Check whether U_α contains a guess $\hat{A}_t$, the biggest one, such that $\hat{A}_t \subseteq A_s$ and $a^s \in \hat{A}_t$.
If it does, do nothing, but require that only B_{k_α} be defined bigger.
Otherwise, we know that $\hat{A}_t \not\subseteq A_{s+1}$ (because of a^s), and we enumerate the related $b_{c_{k_\alpha}}[t]$ into B and extract the associated x_α from E. Declare that α is satisfied until the next stage when $\hat{A}_t$ is contained in A again. *In this case,* $b_{c_{k_\alpha}}[t]$ *is enumerated into* B *and* $b_{c_{k_\alpha}}[s]$ *(from the new block) is removed from* B *(without loss of generality, we can assume that they are different) at this stage, and* $b_{c_{k_\alpha}}[s]$ *will be kept outside* B *forever.* Again we require that only B_{k_α} be defined bigger.

After these two checks, we rectify $\Gamma^{A,B}$ as follows:

$\Gamma^{A,B}$-rectification module.

For all elements $n < s$, check whether there is some n such that $\Gamma^{A,B}(n) \neq \overline{K}(n)$. If there is no such n, do nothing. Otherwise, perform the following actions for the least such n:

— $n \in \overline{K}$.

If n has not been enumerated into $\Gamma^{A,B}$ before (so a_n and B_n (also c_n) have not been defined), define both a_n and the elements of B_n as big numbers. We let B_n have size $g_n + 1$, where $g_n = \sum_{x<a_n} g(x)$. Let $c_n = g_n + 1$.

Otherwise, let $s^- < s$ be the last stage when n was in $\Gamma^{A,B}$.

We define both a_n *and elements of* B_n *as big numbers if between stages* s^- *and* s there is some $l \leqslant n$ such that l leaves $\Gamma^{A,B}$ or n is a threshold of some N-strategy and this N-strategy requires that a_n and B_n be redefined big. Again we let B_n have size $g_n + 1$, where $g_n = \sum_{x<a_n} g(x)$. Set $c_n = g_n + 1$. In the former case we

https://doi.org/10.1017/S0960129508007378 Published online by Cambridge University Press

extract all elements $\leqslant b_{c_m}[s+1]$ in B_m for $m \geqslant n$ from B to remove m from $\Gamma^{A,B}$.

If there is no such l or N-strategy, redefine a_n as before and B_n as a new block consisting of big elements. We let B_n have size $g_n + 1$, where $g_n = \sum_{x<a_n} g(x)$. Set $c_n = g_n + 1$.

In all cases, enumerate all numbers from B_n into B and the axiom

$$\langle n, A_{s+1} \upharpoonright a_n + 1, \bigcup_{m \leqslant n} B_m[s+1]\rangle$$

into Γ.

— $n \notin \overline{K}$.
Find the valid axiom in Γ for n (if any), $\langle n, A_t \upharpoonright a_n + 1, M_n\rangle$ say, and let $t < s+1$ be the stage at which $\langle n, A_t \upharpoonright a_n + 1, M_n\rangle$ is enumerated into Γ. Extract all elements $\leqslant b_{c_m}[s+1]$ in $B_m, m \geqslant n$, from B to remove m from $\Gamma^{A,B}$. *This action removes all $m \geqslant n$ from $\Gamma^{A,B}$.*

Now we construct a path δ_{s+1} of length $\leqslant s$ through the tree T. Each node $\alpha \subseteq \delta_{s+1}$ is said to be visited at stage $s+1$.

Construction of δ_{s+1}

We will define $\delta_{s+1}(n)$ for $n < s+1$ by induction on n. If $n = s+1$, we stop stage $s+1$ and go to the next stage. Suppose $\delta_{s+1}(i-1)$ is defined. We let $\delta_{s+1} \upharpoonright i$ be α, a strategy working on the requirement N_{Φ_i}. When α is visited for the first time after being initialised, it starts from *Setup* to define k_α, the threshold of α, and we define $\delta_{s+1} = \alpha$ and go to the next stage. Otherwise, we go to the *Attack* part.

Setup

If a threshold k_α has not been defined or is cancelled, we define it as a big number – bigger than any element that has appeared so far in the construction.

Attack

1 If x_α has no definition at stage s, define x_α as a big number and let $\delta_{s+1} = \alpha$. Go to the next stage.
2 If $\gamma_s(k_\alpha)$ has not been defined yet, let $\delta_{s+1} = \alpha$. Go to the next stage.
3 If α is declared to be satisfied at (the last one) stage $t < s$ and the associated guess $\hat{A}_t$ is not contained in A_{s+1}, let the outcome of α be 0 *(satisfied)*. Go to the next substage.
4 If none of 1–3 applies and x_α is not in $\Phi_\alpha^B[s+1]$, let the outcome of α be 1 *(waiting)*. Go to the next substage.
5 If none of 1–3 applies and x_α is in $\Phi_\alpha^B[s+1]$, enumerate into U_α a new guess $\hat{A}_{s+1} = A_{s+1} \upharpoonright a_{k_\alpha}[s+1]$. Extract $b_{c_{k_\alpha}}[s+1]$ from B. Decrease $c_{k(\alpha)}$ by one. *(This action moves n out of $\Gamma^{A,B}$ for each $n \geqslant k_\alpha$.)*
Cancel all the markers a_n and B_n for $n \geqslant k_\alpha$, and request that a_{k_α} and B_{k_α} be defined as big numbers.
Let $\delta_{s+1} = \alpha$. Initialise all the strategies of lower priority and go to stage $s+1$.

This completes the construction of B.

https://doi.org/10.1017/S0960129508007378 Published online by Cambridge University Press

Verification

As in Section 3, we can now verify that the constructed B satisfies all the requirements. Define the true path $f \in [T]$ as the limit of δ_s, $s \in \omega$, as in Section 3. We can now prove the following crucial lemma in a similar way.

Lemma 4.1. For each n, let t_n be the last stage at which $f \upharpoonright n$ is initialised. The following are true:

(0) t_n exists.
(1) There is a stage $t_1(n) \geqslant t_n$ after which $f \upharpoonright n$ cannot be reset again.
(2) There is a stage $t_2(n) \geqslant t_1(n)$ after which $f \upharpoonright n$ does not act according to the A-changes in the A-Check part or in the *Attack* part. In particular, after stage $t_2(n)$, $f \upharpoonright n$ does not initialise lower priority strategies and does not affect the definition of Γ.
(3) $f(n)$ is defined. That is, after a stage $t_3(n) \geqslant t_2(n)$ for any $s > t_3(n)$, we have $f \upharpoonright n^\frown \mathcal{O} \subseteq \delta_s$, where $\mathcal{O}$ is the true outcome of $f \upharpoonright n$.

Proof. We prove the lemma by induction on n. Suppose (0)–(3) are all true for $m < n$. We will prove that (0)–(3) are also true for n. Let $f \upharpoonright n = \alpha$. It is easy to see that t_n exists from the assumption that (0)–(3) are true for $f \upharpoonright (n-1)$. In particular, by (3), after a late enough stage, whenever $f \upharpoonright (n-1)$ is visited, $f \upharpoonright n$ is also visited, so the construction never goes to the left of $f \upharpoonright n$ again. Thus (0) is true for n.

Let s_0 be the first α-stage after t_n. At this stage α defines k_α and will never redefine it again, modulo finitely many occurrences of shifting when the selected k_α enters K. As $\overline{K}$ is infinite, such a shifting process can happen at most finitely often. Let $s_1 \geqslant s_0$ be the last stage after which such a shifting never happens again.

We now show that α can be reset at most finitely often. In the construction, only A-changes and $\overline{K}$-changes can reset α. First, after stage s_1, α can be reset by the $\overline{K}$-changes only finitely often as k_α is fixed. So there is a stage $s_2 \geqslant s_1$ after which $\overline{K}$ never changes below this k_α, and thus α will never again be reset by the $\overline{K}$-changes. As U_α can only be cancelled when α is initialised or reset by the $\overline{K}$-changes, U_α can never be cancelled after stage s_2.

To prove that α can be reset by the A-changes only finitely often, as A is Δ^0_2, we only need to prove by induction that after a late enough stage $s_3 \geqslant s_2$, for any $j < k_\alpha$, if $\gamma(j)$ is defined, then $a_j[s_3]$ is fixed, which is obviously true. Let $t_1(n) = s_3$. Then after stage $t_1(n)$ no A-changes can reset α again (*correspondingly, to ensure that the construction of* $\Gamma^{A,B}$ *follows the one axiom rule, no* b_j *with* $j < k_\alpha$ *will be extracted from* B *to rectify* $\Gamma^{A,B}$).

As U_α can only be cancelled when α is initialised or reset, we know that U_α can never be cancelled after stage t_1. Thus (1) is true for n.

We now prove (2). By the choice of s_3 above, we assume that for each $j < k_\alpha$,we have a_j is fixed and A does not change below a_j again.

In order to show a contradiction, suppose that α acts infinitely many times. As A is Δ^0_2, for a particular $a_{k_\alpha}[s]$, A can change below it at most finitely often. Therefore, α reaches Case 5 in the *Attack* part infinitely often, each time a guess $\hat{A}_s (= A_s \upharpoonright a_{k_\alpha}[s] + 1)$ of A for

https://doi.org/10.1017/S0960129508007378 Published online by Cambridge University Press

some s is put into U_α, and a_{k_α} is required to be defined bigger. Let

$$s'_1 < s'_2 < \cdots < s'_m < \cdots$$

be the list of these stages after stage s_3 at which α reaches Case 5 through $a_{k_\alpha}[s'_i]$, respectively.

We claim that for each i, $\hat{A}_{s'_i} \subset A$. Assume this is not the case, and let y be in $\hat{A}_{s'_i}$, but not in A. Then this y is less than $a_{k_\alpha}[s'_i] + 1$ and can provide chances for α to do an action in the A-Check part, so, eventually, α will be satisfied forever through this y (see next paragraph), and α will do no more actions in the construction, which contradicts our assumption.

We now show that this y enables us to satisfy α. Note that at stage s'_i, at Case 5, x_α is enumerated into Φ^B_n by an enumeration, and the action we did then was to extract $b_{c_{k_\alpha}}[s'_i]$ from B to remove all $l \geqslant k_\alpha$ from $\Gamma^{A,B}$. If before stage s'_{i+1} some element, z, in $\hat{A}_{s_i}$ leaves A, then $b_{k_\alpha}[s'_i]$ is put into B to re-enumerate x_α into Φ^B_n. By our choice of s'_{i+1} (by our assumption, it exists), this z must enter A again between s'_i and s'_{i+1}, and, as a consequence, $b_{c_{k_\alpha}}[s'_{i+1}]$, which is less than $b_{c_{k_\alpha}}[s'_i]$, is removed from B. As A is Δ^0_2, and we assume that y is not in A, there is a (biggest) j such that y is in $\hat{A}_{s'_{i+j}}$ and y leaves A after stage s'_{i+j}. As y is also in $A_{s'_{i+j}}$ and y is less than $b_{c_{k_\alpha}}[s'_{i+j}]$, when y leaves A again, $b_{c_{k_\alpha}}[s'_{i+j}]$ is enumerated into B, and, as a consequence, x_α is enumerated into Φ^B_n again. After this, y remains out of A, so $\hat{A}_{s'_{i+j}} \not\subset A$ afterwards, and α is satisfies forever.

We can conclude from the above claim that A is computably enumerable as follows: for each x, x is in A if and only if x is in $\hat{A}_{s'_i}$ for some i. This contradicts our assumption that A is not computably enumerable.

Therefore, α can act at most finitely often and (2) is true for α. Let $t_2(n)$ be the last stage at which α acts.

Now we can see that after $t_2(n)$ the *Attack* part is always at Case 3 (satisfied) or Case 4 (waiting). Correspondingly, α will always have outcome 0 or 1, respectively. If α remains at Case 4, then after stage $t_2(n)$, x_α can never be enumerated into Φ^B_n. If α stops at Case 3, stage $t_2(n)$ is the last stage at which α acts, and at this stage a number y, as mentioned above, is found and never enters A afterwards.

Let $\mathcal{O}$ be the true outcome of α. Then after stage $t_2(n)$, whenever α is visited, $\alpha^\frown\mathcal{O}$ is also visited and (3) is proved.

This completes the proof of the lemma. □

Following Lemma 4.1, we get the following two lemmas immediately.

Lemma 4.2. f is well defined and infinite.

Now we show that the constructed B is a 3-c.e. set.

Lemma 4.3. B is a 3-c.e., as required.

Proof. Note that in the construction whenever we extract a number from B to lift the γ uses, we can always do so because we can select a_n first, and then using the change-bounding function, g, we select B_n with enough elements. We only need to show that every number is extracted from B at most once, and hence B is 3-c.e.

https://doi.org/10.1017/S0960129508007378 Published online by Cambridge University Press

Fix n. We consider a γ-use, $\gamma(n)$.

If n is not a threshold of some N-strategy, then for any a_n, B_n, selected in the construction, there are two cases:

- One case is when we either keep all elements of B_n in B because n is in $\overline{K}$ and A does not have changes below a_n, or we are not allowed to remove elements from B because some strategy α with $k_\alpha < n$ goes to Case 5 in the *Attack* part. In this case some element from B_{k_α} is removed from B, and moves n out of $\Gamma^{A,B}$.
- The other case is when A changes below a_n later and we are allowed to move all elements of B_n out. In this case we just do it, and define B_n later as a set of bigger numbers. After this stage, no element in this old B_n can be enumerated into B again.

If n is a threshold of some N-strategy α, k_α say, there are three cases. The first two cases are exactly the same as the cases discussed in the last paragraph. We now consider the third case: α attacks at Case 5 of the *Attack* part by extracting $b_{c_{k_\alpha}}$, m say, from B when we see that some $\hat{A}_t \not\subset A$. Note that c_{k_α} is decreased by one, and the new $b_{c_{k_\alpha}}$ is different from the previous one, m. If after this stage we never have Case 2 for this particular α, we can never re-enumerate this m into B. Otherwise, m is re-enumerated into B during the A-check, and from that point on we wait for $\hat{A}_t \not\subset A$ to happen again or for a new B_{k_α} to be defined. If $\hat{A}_t \not\subset A$ happens first, we extract $b_{c_{k_\alpha}}$, the new one, from B. Otherwise, we have a new B_{k_α}, and we will take $b_{c_{k_\alpha}}$ from this new B_{k_α}. Of course, m is not in this new B_{k_α}. In any case, m can be removed from B at most once. □

The next lemma states that all the N-requirements are satisfied.

Lemma 4.4. Every N-requirement is satisfied.

Proof. Fix n and let α be a N_n-strategy on f. By Lemma 4.1, there is a late enough stage, t, after which α cannot be initialised or reset again, α will not act again in the remainder of the construction. Also, we can assume that α has true outcome $\mathcal{O}$, and after stage t, each stage is an $\alpha^\frown\mathcal{O}$-stage.

There are two cases:

- $\mathcal{O}$ is 1.
  Then after stage t in the construction, α is always in Case 4, which means that no axiom will enumerate x_α into Φ_n^B, so $E(x_\alpha) = 1 \neq 0 = \Phi_n^B(x_\alpha)$, and N_n is satisfied.
- $\mathcal{O}$ is 0.
  Then α is always in Case 3 after stage t, which means that at the last stage when α acts, x_α is actually enumerated into Φ_n^B and extracted from E. As described in Lemma 4.1, the enumeration of $\Phi_n^B(x_\alpha)$ is clear of the γ-uses, and hence it is preserved. Again, N_n is satisfied. □

Exactly the same argument in the proof of Lemma 3.4 shows that the S is satisfied.

Lemma 4.5. The S requirement is satisfied. That is, for any n, we have $\Gamma^{A,B}(n) = \overline{K}(n)$.

This completes the proof of Theorem 1.3.

References

Cooper, S. B. (1982) Partial degrees and the density problem. *J. Symb. Log.* **47** 854–859.

Cooper, S. B. (1984) Partial Degrees and the density problem, part 2: the enumeration degrees of the Σ_2 sets are dense. *J. Symb. Log.* **49** 503–513.

Cooper, S. B. (1990) Enumeration reducibility, nondeterministic computations and relative computability of partial functions. In Recursion Theory Week, Oberwolfach 1989. *Springer-Verlag Lecture Notes in Mathematics* **1432** 57–110.

Cooper, S. B. (2004) *Computability Theory*, Chapman and Hall/CRC Mathematics.

Cooper, S. B. and Copestake, C. S. (1988) Properly Σ_2 enumeration degrees. *Zeits. f. Math. Logik. u. Grundl. der Math.* **34** 491–522.

Cooper, S. B., Sorbi, A. and Yi, X. (1996) Cupping and noncupping in the enumeration degrees of Σ^0_2 sets. *Ann. Pure Appl. Logic* **82** 317–342.

Copestake, K. (1988) 1-Genericity enumeration Degrees. *J. Symb. Log.* **53** 878–887.

Copestake, K. (1990) 1-Genericity in the enumeration degrees below $\mathbf{0}'_e$. In: Petkov, P. P. (ed.) *Mathematical Logic*, Plenum Press 257–265.

Gutteridge, L. (1971) *Some Results on Enumeration Reducibility*, Ph.D. thesis, Simon Fraser University.

Soare, R. I. (1987) *Recursively enumerable sets and degrees*, Springer-Verlag.

Soskova, M. and Wu, G. (2007) Cupping Δ^0_2 enumeration degrees to $\mathbf{0}'_e$ (extended abstract). In: Computability in Europe 2007. *Springer-Verlag Lecture Notes in Computer Science* **4497** 727–738.