MULTIPLE GENERICITY: A NEW TRANSFINITE HIERARCHY OF GENERICITY NOTIONS

MICHAEL MCINERNEY AND KENG MENG NG

Abstract. We introduce a transfinite hierarchy of genericity notions stronger than 1-genericity and weaker than 2-genericity. There are many connections with Downey and Greenberg's hierarchy of totally α-c.a. degrees [8]. We give several theorems concerning the strength required to compute multiply generic degrees, and show that some of the levels in the hierarchy can be separated, and that these separations can be witnessed by a Δ_2^0 degree. Finally, we consider downward density for these classes.

1. Introduction

The notions of measure and category on the real line give rise to different ways in which we might think of a real number as being typical. In computability theory, we are interested in studying these classical notions in the effective setting. With respect to effective measure, the typical reals are the algorithmically random reals, where a huge body of recent research is devoted to studying the different properties of algorithmic randomness. Two books [23, 9] have recently appeared collecting some of this work. In particular there have been various papers exploring the interactions between classical Turing degrees and algorithmic randomness [1, 2, 22, 10, 7], and the different structural properties possessed by the Turing degrees of random reals.

The concern of this paper is to follow up on the more neglected concept of typicality: effective category. The reals with are typical with respect to category are known as Cohen generic, or simply generic. Intuitively speaking, a (effectively) generic real is one that is constructed step by step, where at each step we specify more of a finite initial segment of the real by meeting the next requirement. The requirements are usually described by a sequence of effective topological descriptions, and since *any* sufficiently generic real (instead of the one explicitly constructed) will also meet all the requirements, this class of constructions became known as the effective analogue of Cohen's construction.

The most commonly studied notions of genericity are the classes of n-generic sets, for $n \geqslant 1$. Though originally formulated in terms of forcing n-quantifier arithmetic sentences, we briefly recall the more common definition given by Jockusch in [15]. Let S be a set of finite binary strings, and A a subset of natural numbers, which we think of as an infinite binary sequence. We say that A meets S if there is some $\sigma \prec A$ such that $\sigma \in S$, and A avoids S if there is some $\sigma \prec S$ such that no extension of σ is in S. Then A is said to be n-generic if for every Σ_n^0 set of strings S, A either meets or avoids S.

Both authors are partially supported by MOE2015-T2-2-055.

These classes form a proper hierarchy: for all $n \geqslant 1$, the n-generic sets properly contain the $n+1$-generic sets. This was later refined by Kurtz [18] who introduced the weakly n-generic sets. We say that a set of strings S is dense if every finite binary string has an extension in S. Then a set is weakly n-generic if it meets every dense Σ^0_n set of string. Kurtz [18] showed that for all $n \geqslant 1$, the n-generic sets properly contain the weakly $n+1$-generic sets, which properly contain the $n+1$-generic sets. These proper containments even hold for Turing degrees, where a Turing degree is (weakly) n-generic if it contains a (weakly) n-generic set.

Most of the interest in genericity occurs at the levels $n = 1$ and $n = 2$. Even at these low levels, there is a large difference in the behaviour of 1-generic and 2-generic sets. For example, the class of 1-generic sets has measure 1, whereas the class of 2-generic sets has measure 0. As another example, the 2-generic sets are downward dense [1] below 2-generic sets, whereas this fails in general for 1-generics. Intuitively, typical behaviour seems to start with 2-genericity, but can sometimes fail at the level of 1-genericity. Many such results for genericity, as well as randomness, are given in [1], and the survey article [2].

Given this, it would seem interesting to develop notions of genericity which are stronger than 1-genericity, but weaker than 2-genericity. Several such notions already exist, though they have not received much attention. The most well-known is pb-genericity, introduced by Downey, Jockusch, and Stob in [12]. Consider a total function $f : 2^{<\omega} \to 2^{<\omega}$ for which there is a total computable function $g : 2^{<\omega} \times \omega \to 2^{<\omega}$ and primitive recursive function $p : \omega \to \omega$ such that

(1) $\lim_s g(\sigma, s) = f(\sigma)$,
(2) $g(\sigma, s) \succcurlyeq \sigma$, and
(3) $|\{s \in \omega : g(\sigma, s+1) \neq g(\sigma, s)\}| < p(\sigma)$

for all $\sigma \in 2^{<\omega}$ and $s \in \omega$. We say that the set A is pb-generic if it meets the range of all such functions f.

Suppose that we are given functions f, g, and p as above, and that we are trying to construct a real A to meet the range of f. As usual, we construct A as the limit of a computable sequence of approximations A_s for $s \in \omega$. Say that we have decided by stage s_0 that we would like σ to be an initial segment of A. Then for A to meet the range of f, at all stages $s \geqslant s_0$, we would simply let A_s extend $g(\sigma, s)$. Although we do not know what $g(\sigma, s)$ may be (except that it must extend σ), we do know that after stage s_0, we need to change our approximation to A at most $p(\sigma)$ many times in order to meet the range of f. The crucial feature here is that p is primitive recursive. We can think of the bound $p(\sigma)$ as being given "in advance" – we do not need to perform an unbounded search in order to compute $p(\sigma)$, and so we can, for instance, organise permission before the construction begins. Indeed, it is shown in [12] that every array noncomputable degree computes a pb-generic set.

The first way in which we may generalise pb-genericity is by replacing the primitive recursive function p with a total computable function h. This notion was called c-genericity by Schaeffer [24], though we will refer to it as weak ω-change genericity. Here, the bound is no longer given in advance, but can be thought of as given during the construction: the bound is only declared at the stage s at which $h_s(\sigma)\downarrow$. As we show in Theorem 4.16, there is an array noncomputable degree which cannot compute a weakly ω-change generic degree. In order for a degree to compute a weakly ω-change generic degree, it must be not ω-c.a. dominated, which

[1] see Section 6 for definitions and more results along these lines

can be thought of as a non-uniform version of array noncomputability. We give the definition for this and other notions related to domination in Section 4.

Note that by properties (1) and (2) and the fact that f is total, f must have dense range. Another way in which we may generalise pb-genericity is to consider how the function f might be partial, so that its range is not necessarily a dense set of strings. We must somehow still track the changes in the approximation for those inputs where f is actually defined. We can do this by instead requiring the functions f and g as above to be total, but allowing the bounding function h to be partial computable. We still insist on properties (1) and (2), and that $h(\sigma)$ bounds the number of changes in the approximation to $f(\sigma)$ for those strings σ with $\sigma \in \operatorname{dom} h$. Then given such functions, we can consider the "range" of the triple $\langle f, g, h\rangle$ to be the set of all strings $f(\sigma)$ where $\sigma \in \operatorname{dom} h$. We say that a set is ω-change generic if it either meets or avoids the range (in this sense) of every such triple.

There is no reason for us to restrict ourselves to a computably bounded number of mind-changes. In [8], Downey and Greenberg extend the notion of an ω-c.a. function, that is, a function which can be computably approximated with a computable number of mind-changes, to α-c.a. functions, for ordinals $\alpha \leqslant \varepsilon_0$. Though there are many details which are needed to properly define this concept, roughly speaking, the approximation is equipped with a sequence of functions $o_s : \omega \to \alpha$ such that if the approximation for input n changes at stage s, then we have $o_s(n) < o_{s-1}(n)$. This allows us to track the changes in the approximation. We make use of this idea for the functions g which approximate the functions f above. This leads to a transfinite hierarchy of genericity notions. We will show that pb-genericity coincides with the first notion in our hierarchy that is stronger than 1-genericity.

Downey and Greenberg [8] also introduce some domination properties in order to generalise the definition of array computability. We show that there are many deep and intricate connections between these domination properties and the hierarchy of genericity notions. These greatly extend the fact that every degree bounds a pb-generic if and only if it is array noncomputable. Our results rely on a fine analysis of the forcing and permitting constructions which can be carried out below a degree with some such domination property. We explore this in Sections 4 and 5. Together with the results from [21], the sequel to the current article, we show that our results are as tight as possible with respect to these notions, which gives a separation of each of the genericity notions in our hierarchy.

2. Background and definitions

We first cover the necessary material from [8] to work with a large collection of ordinals α. We refer the reader to Chapter 2 of [8] for more details.

Recall that every ordinal α can be uniquely expressed as the sum

$$\omega^{\alpha_1} n_1 + \omega^{\alpha_2} n_2 + \cdots + \omega^{\alpha_k} n_k$$

where $n_i < \omega$ are nonzero and $\alpha_1 > \alpha_2 > \cdots > \alpha_k$ are ordinals. This is the Cantor normal form of α. Recall also that

$$\varepsilon_0 = \sup \left\{\omega, \omega^{\omega}, \omega^{\omega^{\omega}}, \omega^{\omega^{\omega^{\omega}}}, \dots\right\}$$

is the least ordinal γ such that $\omega^\gamma = \gamma$, so for all $\alpha < \varepsilon_0$, every ordinal appearing in the Cantor normal form of α is strictly smaller than α.

Let $\mathcal{R} = (R, \leqslant_{\mathcal{R}})$ be a computable well-ordering of a computable set R, and let $|\cdot|: R \to \mathrm{otp}(\mathcal{R})$ be the unique isomorphism between $\mathcal{R}$ and its order-type. The pullback to $\mathcal{R}$ of the Cantor normal form function is the function $\mathtt{nf}_{\mathcal{R}}$ whose domain is R and is defined by letting

$$\mathtt{nf}_{\mathcal{R}}(z) = \langle (z_1, n_1), (z_2, n_2), \ldots, (z_k, n_k) \rangle$$

where $n_i < \omega$ are nonzero, $z_i \in R$, $z_1 >_{\mathcal{R}} z_2 >_{\mathcal{R}} \cdots >_{\mathcal{R}} z_k$, and

$$|z| = \omega^{|z_1|} n_1 + \omega^{|z_2|} n_2 + \cdots + \omega^{|z_k|} n_k.$$

Definition 2.1 ([8])**.** A computable well-ordering $\mathcal{R}$ is canonical if its associated Cantor normal form function $\mathtt{nf}_{\mathcal{R}}$ is also computable.

It is shown in [8] that for every ordinal $\alpha \leqslant \varepsilon_0$ there is a canonical computable well-ordering of order-type α, and further that any two canonical computable well-orderings of order-type α are computably isomorphic. We therefore identify each ordinal $\alpha \leqslant \varepsilon_0$ with a canonical computable well-ordering of order-type α.

With all this in place, we are now able to give the main definitions of this paper.

Definition 2.2. Let $\alpha \leqslant \varepsilon_0$, and let ∞ denote the greatest element of the linear ordering $\alpha + 1$. An *α-change test* is a sequence $\langle f_s, o_s \rangle_{s<\omega}$ of pairs of uniformly computable functions $f_s : 2^{<\omega} \to 2^{<\omega}$ and $o_s : 2^{<\omega} \to \alpha + 1$ such that for all $\sigma \in 2^{<\omega}$ and $s \in \omega$,

- $f_s(\sigma) \succcurlyeq \sigma$,
- $o_{s+1}(\sigma) \leqslant o_s(\sigma)$, and
- if $f_{s+1}(\sigma) \neq f_s(\sigma)$, then $o_{s+1}(\sigma) < o_s(\sigma)$.

For a test $\langle f_s, o_s \rangle_{s<\omega}$, we let the range of the test be

$$\mathrm{range}\ \langle f_s, o_s \rangle_{s<\omega} = \{\lim_s f_s(\sigma) : \lim_s o_s(\sigma) < \infty\}.$$

We will usually write range f for this set.

We use the term *test* in analogy with randomness tests. We may occasionally refer to $f_s(\sigma)$ as the *arrow* for σ at stage s in the test $\langle f_s, o_s \rangle_{s<\omega}$. In α-change tests that we construct, we may say that we *update the arrow for σ at stage s* if $f_s(\sigma) \neq f_{s-1}(\sigma)$.

Proposition 2.3. *There is an effective list of all α-change tests.*

Proof. We follow the proof of proposition 1.7 from [8]. There exists an effective list $\langle h_s, m_s \rangle_{s<\omega}$ of all pairs of partial computable functions where for all $s \in \omega$, $h_s : 2^{<\omega} \to 2^{<\omega}$ and $m_s : 2^{<\omega} \to \alpha$, and furthermore, that if $h_s(\sigma)\downarrow$, then $h_s(\sigma) \succcurlyeq \sigma$. We define an effective list $\langle f_s, o_s \rangle_{s<\omega}$ as follows. Let $\sigma \in 2^{<\omega}$. We let $f_0(\sigma) = \sigma$ and $o_0(\sigma) = \infty$. Now let $s > 0$. We let $t_s(\sigma)$ be the greatest $t \leqslant s$ such that for all $r \leqslant t$,

- at stage s we see that $h_r(\sigma)\downarrow$ and $m_r(\sigma)\downarrow$,
- $m_r(\sigma) < \infty$, and if $r > 0$, then $m_r(\sigma) \leqslant m_{r-1}(\sigma)$,
- if $r > 0$ and $h_r(\sigma) \neq h_{r-1}(\sigma)$ then $m_r(\sigma) < m_{r-1}(\sigma)$.

If there is no such t, then we leave $t_s(\sigma)$ undefined. If $t_s(\sigma)$ is defined, then we let $f_s(\sigma) = h_{t_s(\sigma)}(\sigma)$ and $o_s(\sigma) = m_{t_s(\sigma)}(\sigma)$. If $t_s(\sigma)$ is not defined, then we let $f_s(\sigma) = \sigma$ and $o_s(\sigma) = \infty$.

□

Definition 2.4. Let $\alpha \leqslant \varepsilon_0$ and let $\mathbf{a}$ be a Turing degree. We say that $\mathbf{a}$ is *α-change generic* if there is a set $A \in \mathbf{a}$ which meets or avoids the range of all α-change tests.

For $\alpha = \omega$, this notion is equivalent to the definition of ω-change genericity given in the introduction. We now work towards defining the weak version of this notion.

Definition 2.5. Let $\langle f_s, o_s \rangle_{s<\omega}$ be an α-change test. We say that $\langle f_s, o_s \rangle_{s<\omega}$ is *total* if for all $\sigma \in 2^{<\omega}$, $\lim_s o_s(\sigma) < \infty$.

Proposition 2.6. *Let $\alpha \leqslant \varepsilon_0$. Then a set A meets the range of every total α-change test if and only if it meets the range of every α-change test with dense range.*

Proof. For the backwards direction, suppose A meets the range of every α-change test with dense range, and let $\langle f_s, o_s \rangle$ be a total α-change test. Then for all $\sigma \in 2^{<\omega}$, $\lim_s f_s(\sigma)$ is in the range of the test, and $\lim_s f_s(\sigma) \succcurlyeq \sigma$. So the range is dense.

For the forwards direction, suppose A meets the range of every total α-change test, and let $\langle f_s, o_s \rangle$ be an α-change test with dense range. We define a test $\langle g_s, p_s \rangle$ as follows. Let $g_0(\sigma) = f_0(\sigma)$ and $p_0(\sigma) = o_0(\sigma)$ for all σ. Given $g_s(\sigma)$ and $p_s(\sigma)$, if $p_s(\sigma) < \infty$ and at some previous stage $t > 0$ we set $g_{t+1}(\sigma) = f_{t+1}(\tau)$ for some τ, then we let $g_{s+1}(\sigma) = f_{s+1}(\tau)$ and $p_{s+1}(\sigma) = o_{s+1}(\tau)$. Otherwise, if there is some $\tau \succcurlyeq \sigma$ such that $o_{s+1}(\tau) < \infty$, then we choose the least such τ, and let $g_{s+1}(\sigma) = f_{s+1}(\tau)$ and $p_{s+1}(\sigma) = o_{s+1}(\tau)$. Then because $\langle f_s, o_s \rangle$ has dense range, $\langle g_s, p_s \rangle$ will be a total test. Therefore A meets the range of $\langle g_s, p_s \rangle$, and by construction, A meets the range of $\langle f_s, o_s \rangle$. ☐

Definition 2.7. We say that $\mathbf{a}$ is *weakly α-change generic* if there is a set $A \in \mathbf{a}$ which meets the range of all total α-change tests.

We would now like to see whether we can give a definition of pb-genericity along these lines, and if possible, extend it to all ordinals $\alpha \leqslant \varepsilon_0$. We follow the approach taken by Downey and Greenberg in their definition of the uniformly totally α-c.a. degrees (see Section 3.3 of [8]).

Definition 2.8. Let $h : 2^{<\omega} \to \alpha$ be a total computable function. We say that the α-change test $\langle f_s, o_s \rangle_{s<\omega}$ is *h-bounded* if for all $\sigma \in 2^{<\omega}$ and $s \in \omega$, if $o_s(\sigma) < \infty$, then $o_s(\sigma) \leqslant h(\sigma)$.

Definition 2.9. Let $\leqslant_{\mathrm{L}}$ denote the usual length-lexicographic ordering on $2^{<\omega}$. We say that $h : 2^{<\omega} \to \alpha$ is an *α-order function* if h is total and computable with range cofinal in α, and such that if $\sigma \leqslant_{\mathrm{L}} \tau$, then $h(\sigma) \leqslant h(\tau)$.

Proposition 2.10. *Let $\alpha \leqslant \varepsilon_0$ and let $h_1, h_2 : 2^{<\omega} \to \alpha$ be α-order functions. Then a Turing degree contains a set which meets all total h_1-bounded α-change tests if and only if it contains a set which meets all total h_2-bounded α-change tests.*

Proof. It suffices to show that given some set A which meets all total h_1-bounded α-change tests, there is some $B \leqslant_{\mathrm{T}} A$ which meets all total h_2-bounded α-change tests. We construct a Turing functional Γ such that $\Gamma(A)$ is the desired set B.

We in fact construct a truth table functional Γ. We define the function $h_2^{-1} : \omega \to \omega$ as follows. For any $n \in \omega$, let $h_2^{-1}(n)$ be the least $m \in \omega$ such that

$$\min\{\, h_1(\sigma) : |\sigma| = m \,\} \geqslant \max\{\, h_2(\tau) : |\tau| = n \,\}.$$

Then h_2^{-1} is total and computable because h_1 and h_2 are α-order functions. Because h_2 has range cofinal in α, h_2^{-1} will have range cofinal in ω, and so we may assume that h_2^{-1} is strictly increasing. We may also assume that $h_1(\lambda) \geqslant h_2(\lambda)$.

We define Γ recursively as follows. Let $\Gamma(\alpha) = \lambda$ for all strings α of length $h_2^{-1}(0)$, where λ is the empty string. Now assume that we have defined $\Gamma(\alpha)$ for all strings α of length $h_2^{-1}(n)$. Then for any string α' of length $h_2^{-1}(n+1)$, let α be the initial segment of α' of length $h_2^{-1}(n)$. If $\alpha' \succcurlyeq \alpha\,\hat{}\,0$, then we let $\Gamma(\alpha') = \Gamma(\alpha)\,\hat{}\,0$, and if $\alpha' \succcurlyeq \alpha\,\hat{}\,1$, then we let $\Gamma(\alpha') = \Gamma(\alpha)\,\hat{}\,1$. It is immediate that Γ is consistent and onto $2^{<\omega}$. We extend the domain of Γ consistently so that its domain is $2^{<\omega}$.

We now show that $\Gamma(B)$ meets all total h_2-bounded α-change tests. Suppose $\langle f_s, o_s\rangle_{s<\omega}$ is such a test. We define a total h_1-bounded α-change test $\langle g_s, p_s\rangle_{s<\omega}$ such that if A meets range g, then $\Gamma(A)$ meets range f.

For any σ, we set $g_s(\sigma) = \sigma$ and $p_s(\sigma) = \infty$ until we see a stage t where $o_t(\Gamma(\sigma)) < \infty$. We then let $g_t(\sigma)$ be some string extending σ such that $\Gamma(g_t(\sigma)) = f_t(\Gamma(\sigma))$ and $p_t(\sigma) = o_t(\Gamma(\sigma))$. If at some later stage t' we see that $f_{t'}(\Gamma(\sigma)) \neq f_{t'-1}(\Gamma(\sigma))$, then we set $g_{t'}(\sigma)$ to be some string extending σ such that $\Gamma(g_{t'}(\sigma)) = f_{t'}(\Gamma(\sigma))$ and $p_{t'}(\sigma) = o_{t'}(\Gamma(\sigma))$. By the definition of h_2^{-1} and Γ, $\langle g_s, p_s\rangle_{s<\omega}$ is an h_1-bounded test, and it is total because $\langle f_s, o_s\rangle_{s<\omega}$ is total.

□

Given the previous proposition, we are able to make the following definition.

Definition 2.11. We say that $\mathbf{a}$ is *uniformly α-change generic* if for all (some) α-order functions h, there is a set $A \in \mathbf{a}$ such that A meets the range of all total h-bounded α-change tests.

We show that the notions of uniform ω-change genericity and pb-genericity are equivalent up to Turing degree in Theorem 3.2.

It is worthwhile to note the following simple proposition.

Proposition 2.12. *Let $\alpha \leqslant \varepsilon_0$, let $\nu \in 2^{<\omega}$, and let $\langle f_s, o_s\rangle_{s<\omega}$ be an α-change test. Then if there is no $\tau \succcurlyeq \nu$ and no $s \in \omega$ such that $f_s(\tau) \succcurlyeq \nu$, then any set $A \succ \nu$ avoids* range f.

Proof. We have $\lim_s f_s(\sigma) \succcurlyeq \sigma$ for all σ such that $\lim_s o_s(\sigma) < \infty$. Then by assumption, the only strings in range f which extend ν are those of the form $\lim_s f_s(\rho)$ for some $\rho \preccurlyeq \nu$. As there are only finitely many such strings, any set A such that $A \succ \nu$ avoids range f. □

3. A hierarchy of genericity notions

It is clear that for any $\alpha \leqslant \varepsilon_0$ and any Turing degree $\mathbf{a}$ that α-change generic implies weakly α-change generic implies uniformly α-change generic. We show that there is a transfinite hierarchy of these genericity notions, as in [8].

$$\begin{array}{c} \text{weakly 2-generic} \\ \Downarrow \\ \vdots \\ \Downarrow \\ \omega^2\text{-change generic} \\ \Downarrow \\ \text{weakly } \omega^2\text{-change generic} \\ \Downarrow \\ \text{uniformly } \omega^2\text{-change generic} \end{array}$$

$$\Downarrow$$

ω-change generic

$$\Downarrow$$

weakly ω-change generic

$$\Downarrow$$

uniformly ω-change generic $\equiv$ pb-generic

$$\Downarrow$$

1-change generic $\equiv$ 1-generic

$$\Downarrow$$

weakly 1-generic

We will show that the hierarchy collapses to the levels above, that the series of implications holds, and furthermore, that each implication is strict and can be witnessed by a Δ^0_2 Turing degree. More precisely, we have the following theorem.

Theorem 3.1. *Let $\alpha \leqslant \varepsilon_0$.*

(i) If α is not a power of ω, and β is such that $\alpha \in (\omega^\beta, \omega^{\beta+1})$, then the classes of degrees which are ω^β-change generic, uniformly α-change generic, weakly α-change generic, and α-change generic, are all equal.

(ii) If α is a power of ω, then if $\alpha = \omega^\beta$, the class of degrees which are uniformly ω^β-change generic is a proper subclass of the class of degrees which are weakly ω^β-change generic, which is a proper subclass of the degrees which are ω^β-change generic, which is a proper subclass of the degrees which are uniformly $\omega^{\beta+1}$-change generic. Moreover, these proper inclusions can be witnessed by a Δ^0_2 Turing degree.

We will refer generally to the notions of uniform α-change genericity, weak α-change genericity, and α-change genericity as notions of multiple genericity.

We first show that our definitions align with the definitions of 1-genericity and pb-genericity.

Theorem 3.2. *A Turing degree is 1-change generic if and only if it contains a 1-generic set, and a Turing degree is uniformly ω-change generic if and only if it contains a pb-generic set.*

Proof. It is easy to see that 1-change tests are in correspondence with c.e. sets of strings, and so the first statement holds.

Now suppose that the degree $\mathbf{a}$ is uniformly ω-change generic. Let $h : \omega \to \omega$ be some computable order function which dominates all primitive recursive functions, and let $A \in \mathbf{a}$ be such that A meets the range of all h-bounded ω-change tests. We show that A is pb-generic. Let the functions f, g, and p be as in the discussion of pb-genericity in the introduction. Then because h dominates p, we can produce an h-bounded ω-change test whose range is equal to the range of f. Then as A meets the range of this ω-change test, it meets the range of f.

For the opposite direction, suppose that the set A is pb-generic. The function $p : 2^{<\omega} \to \omega$ with $p(\sigma) = |\sigma|$ is a primitive recursive ω-order function. Let $\langle f_s, o_s \rangle_{s<\omega}$ be a p-bounded ω-change test. We can present the range of this test as the range of a function that can be approximated with mind-change function bounded by p.

Then as A is pb-generic, it meets the range of this function, and so meets the range of this test. ☐

The following lemma will be used used to prove Part (i) of Theorem 3.1. It is a straightforward adaptation of Lemma 2.2 from [8] to α-change tests.

Lemma 3.3. *Suppose $\gamma \leqslant \varepsilon_0$. Then for all $m \in \omega$, if a set meets or avoids all γ-change tests, it meets or avoids all γm-change tests.*

Proof. Suppose A meets or avoids all γ-change tests and that $\langle f_s, o_s\rangle$ is a γm-change test. We break the test $\langle f_s, o_s\rangle$ up into m many γ-tests.

For every σ and s there is some unique $k < m$ such that $o_s(\sigma) \in [\gamma \cdot k, \gamma \cdot (k+1))$; we denote this k by $k_s(\sigma)$. We have $o_s(\sigma) = \gamma \cdot k_s(\sigma) + \beta_s(\sigma)$ for some $\beta_s(\sigma) < \gamma$. For each $k < m$, we define a γ-change test $\langle g_{k,s}, p_{k,s}\rangle_{s<\omega}$. We let $g_{k,s}(\sigma) = \sigma$ and $p_{k,s}(\sigma) = \infty$ until we see a stage t where $k_t(\sigma) = k$. Then we define $g_{k,t}(\sigma) = f_t(\sigma)$ and $p_{k,t}(\sigma) = \beta_t(\sigma)$. If at some later stage u we see $f_u(\sigma) \neq f_{u-1}(\sigma)$ and $k_u(\sigma) = k$, then we define $g_{k,u}(\sigma) = f_u(\sigma)$ and $p_{k,u}(\sigma) = \beta_u(\sigma)$.

Let k^* be least such that for some σ and s, we have $p_{k^*,s}(\sigma) < \infty$. We know that A meets or avoids $\langle g_{k^*,s}, p_{k^*,s}\rangle$. By the choice of k^*, A must meet or avoid $\langle f_s, o_s\rangle$. ☐

It is immediate that the previous proposition holds with total tests and/or tests which are h-bounded for some order function h. Part (i) of Theorem 3.1 now follows: there is some $m \in \omega$ such that $\alpha \leqslant \omega^\beta m$. Then if a set A is ω^β-change generic, it is $\omega^\beta m$-change generic, and so α-change generic. We noted before that each α-change generic degree is weakly α-change generic and uniformly α-change generic.

Proposition 3.4. *Let $\alpha \leqslant \varepsilon_0$. If a Turing degree is uniformly $(\alpha+1)$-change generic, then it is α-change generic.*

Proof. Suppose that $\mathbf{a}$ is uniformly $(\alpha+1)$-change generic and let $h : 2^{<\omega} \to \alpha + 1$ be the $(\alpha+1)$-order function with $h(\sigma) = \alpha$ for all $\sigma \in 2^{<\omega}$. Let $A \in \mathbf{a}$ be such that A meets the range of all total h-bounded $(\alpha+1)$-change tests. Let $\langle f_s, o_s\rangle_{s<\omega}$ be an α-change test. Then it is easy to check that $\langle f_s, o_s\rangle_{s<\omega}$ is in fact a total h-bounded $(\alpha+1)$-change test. Therefore A witnesses that $\mathbf{a}$ is α-change generic. ☐

With the previous proposition, we now see that the (non-proper) inclusions in part (ii) of Theorem 3.1 hold. The fact that the inclusions are proper and can be witnessed by Δ^0_2 Turing degrees will be shown in Theorem 4.16, Theorem 5.4, and in the main theorem of [21].

Finally, we show that the topmost implication in the diagram above holds.

Proposition 3.5. *Let X be weakly 2-generic. Then for any $\alpha \leqslant \varepsilon_0$, X is α-change generic.*

Proof. Suppose $\langle f_s, o_s\rangle_{s<\omega}$ is an α-change test. We define a dense Σ^0_2 set S as follows. For any string σ, we ask $\varnothing'$ whether there exists an s and a $\tau \succcurlyeq \sigma$ such that $o_s(\tau) < \infty$. There are two cases.

Case 1: the answer is no. Then we enumerate every string $\tau \succcurlyeq \sigma$ into S.

Case 2: the answer is yes. Then let s be the least such, and let τ be the least such for this s. Ask $\varnothing'$ whether there is a $s_1 > s$ such that $o_{s_1}(\tau) < o_s(\tau)$. If the answer is no, then we enumerate $f_s(\tau)$ into S. If the answer is yes, we ask $\varnothing'$

whether there is a $s_2 > s_1$ such that $o_{s_2}(\tau) < o_{s_1}(\tau)$, and act as we did before. We eventually come to some stage where we enumerate some string into S which is equal to $\lim_t f_t(\tau)$ for some τ.

It is easy to see that S is dense. As X is weakly 2-generic, X meets S. Suppose X extends the string $\chi \in S$. If χ was enumerated into S via Case 1, then there is no extension of χ in range f, and so X avoids range $\langle f_s, o_s\rangle_{s<\omega}$. If χ was enumerated into S via Case 2, then as noted above, χ is equal to $\lim_t f_t(\tau)$ for some τ, and so X meets range $\langle f_s, o_s\rangle_{s<\omega}$.

□

4. Domination properties and multiple genericity

In this section we begin to investigate the strength required to compute multiply generic degrees. Our first result is the following simple but important theorem. As no weakly 2-generic set can be Δ^0_2, it shows that there is a degree separation of each notion of multiple genericity from weak 2-genericity.

Theorem 4.1. *For any $\alpha \leqslant \varepsilon_0$, $\varnothing'$ computes an α-change generic degree.*

Proof. Let $\langle\langle f_{i,s}, o_{i,s}\rangle_{s<\omega}\rangle_{i<\omega}$ be an effective list of all α-change tests. We build a set A by finite extension, computably in $\varnothing'$.

Construction

Stage 0: Let $A_0 = \lambda$, the empty string.

Stage s, $s \geqslant 1$: Given A_{s-1}, we ask $\varnothing'$ whether there is some $\tau \succcurlyeq A_{s-1}$ and $t \in \omega$ such that $o_{s-1,t}(\tau) < \infty$. There are two cases

Case 1: the answer is no. Then we let $A_s = A_{s-1}$ and proceed to the next stage.

Case 2: the answer is yes. Let τ be the least such, and let t be the least such for this τ. We ask $\varnothing'$ whether there is a $t_1 > t$ such that $o_{s-1,t_1}(\tau) < o_{s-1,t}(\tau)$. If the answer is no, then we let $A_s = f_{s-1,t}(\tau)$ and proceed to the next stage. If the answer is yes, we ask $\varnothing'$ whether there is a $t_2 > t_1$ such that $o_{s-1,t_2}(\tau) < o_{s-1,t_1}(\tau)$, and act as we did before.

End of construction

If we act in Case 2 at stage s, then we will eventually define $A_s = f_{s-1,t'}(\tau)$ for some t' such that $\lim_u f_{s-1,u}(\tau) = f_{s-1,t'}(\tau)$, and proceed to the next stage. This is because the number of $t \in \omega$ such that $f_{s-1,t}(\tau) \neq f_{s-1,t-1}(\tau)$ is finite, as the sequence $\langle o_{s-1,t}(\tau)\rangle_{t<\omega}$ is non-increasing in the ordinal α, and if $f_{s-1,t}(\tau) \neq f_{s-1,t-1}(\tau)$, then $o_{s-1,t}(\tau) < o_{s-1,t-1}(\tau)$.

Let $A = \bigcup_{s\in\omega} A_s$. It is clear that A either meets or avoids the range of each α-change test, and so the degree of A is α-change generic.

□

Using the approach in the proof of the previous theorem, it is now straightforward to modify the usual proof of the Friedberg jump inversion theorem (for example, the one given in Section 2.16 of [9]) to show that jump inversion holds for multiply generic degrees.

Theorem 4.2. *For any $\alpha \leqslant \varepsilon_0$, if $C \geqslant_{\mathrm{T}} \varnothing'$, then there is an α-change generic degree $\mathbf{a}$ such that $\mathbf{a}' \equiv_{\mathrm{T}} C$.*

For finer results than Theorem 4.1, we look at the domination properties introduced in [8]. We give the required definitions, and again refer the reader to [8] for details.

Let $\mathcal{R} = (R, \leqslant_{\mathcal{R}})$ be a computable well-ordering of a computable set R. An $\mathcal{R}$-*computable approximation* of a function f is a computable approximation $\langle f_s \rangle_{s<\omega}$ of f, equipped with a uniformly computable sequence $\langle o_s \rangle_{s<\omega}$ of functions from ω to R such that for all x and s:

- $o_{s+1}(x) \leqslant_{\mathcal{R}} o_s(x)$, and
- if $f_{s+1}(x) \neq f_s(x)$, then $o_{s+1}(x) <_{\mathcal{R}} o_s(x)$

Definition 4.3 ([8])**.** A function $f : \omega \to \omega$ is $\mathcal{R}$-computably approximable (or $\mathcal{R}$-c.a.) if it has an $\mathcal{R}$-computable approximation.

Definition 4.4 ([8])**.** Let $\alpha \leqslant \varepsilon_0$. A function f is α-c.a. if it is $\mathcal{R}$-c.a. for some (all) canonical computable well-ordering $\mathcal{R}$ of order-type α.

We will have occasional use for the following definition as well.

Definition 4.5 ([8])**.** Let $\mathcal{R}$ be a computable well-ordering. An $(\mathcal{R}+1)$-computable approximation $\langle f_s, o_s \rangle$ is *tidy* if:

- for all n, $f_0(n) = 0$, and
- for all n and s, if $o_s(n+1) \in R$ then $o_s(n) \in R$.

Recall that if $\mathcal{C}$ is a class of functions from ω to ω, then a Turing degree $\mathbf{a}$ is $\mathcal{C}$-*dominated* if every function $g \in \mathbf{a}$ (or equivalently $g \leqslant_{\mathrm{T}} \mathbf{a}$) is dominated by some function $f \in \mathcal{C}$. We say that a Turing degree $\mathbf{a}$ is *uniformly* $\mathcal{C}$-*dominated* if there is some function $f \in \mathcal{C}$ such that every function $g \in \mathbf{a}$ is dominated by f.

Definition 4.6 ([8])**.** A Turing degree is *α-c.a. dominated* if it is $\mathcal{C}$-dominated, and *uniformly α-c.a. dominated* if it is uniformly $\mathcal{C}$-dominated, where $\mathcal{C}$ is the class of all α-c.a. functions.

Note that a $\mathbf{0}$-dominated degree is also called *hyperimmune-free*.

These notions form a hierarchy as follows. It is easy to see that if $\alpha < \beta$, then every α-c.a. function is β-c.a., and that for any class $\mathcal{C}$, if a degree is uniformly $\mathcal{C}$-dominated, then it is $\mathcal{C}$-dominated. Again, it is only possible for these notions to differ at powers of ω. As our results refer to degrees which are *not* dominated in some sense, we present the hierarchy as follows.

$$\begin{gathered}
\vdots \\
\Downarrow \\
\text{not } \omega^2\text{-c.a. dominated} \\
\Downarrow \\
\text{not uniformly } \omega^2\text{-c.a. dominated} \\
\Downarrow \\
\text{not } \omega\text{-c.a. dominated} \\
\Downarrow \\
\text{not uniformly } \omega\text{-c.a. dominated} \\
\Downarrow \\
\text{not } \mathbf{0}\text{-dominated}
\end{gathered}$$

Downey and Greenberg show that this hierarchy is proper, and in fact the separation of each level can be witnessed by a c.e. degree (see Section 3.5 of [8]).

There seems to be a close connection between notions of genericity which involve meeting dense sets of strings, and domination properties. For a Turing degree $\mathbf{b}$, we say that a Turing degree $\mathbf{a}$ is $\mathbf{b}$-*dominated* if it is $\mathcal{C}$-dominated, for $\mathcal{C}$ the class of all functions computable in $\mathbf{b}$. The first result in this direction is the following.

Theorem 4.7 ([18, 19]). *For all $n \geqslant 1$, every weakly $(n+1)$-generic degree is not $\mathbf{0^{(n)}}$-dominated.*

In fact, Kurtz showed ([18], see also Section 2.24 of [9]) that a degree is weakly 1-generic if and only if it is not $\mathbf{0}$-dominated.

The notion of array noncomputability was originally defined only for c.e. degrees in [11], but was extended in [12] to the general degrees using a domination property. In our terminology, a degree is array noncomputable if and only if it is not uniformly ω-c.a. dominated.

Theorem 4.8 ([12]). *Every array noncomputable degree computes a pb-generic set, and every pb-generic set is of array noncomputable degree.*

Stated another way, the previous theorem says that the upward closure (in the Turing degrees) of the pb-generic sets is exactly the set of array noncomputable degrees. We give analogous results for the uniformly α-change generic and weakly α-change generic degrees.

Proposition 4.9. *Suppose the Turing degree* $\mathbf{a}$ *computes a weakly α-change generic degree. Then* $\mathbf{a}$ *is not α-c.a. dominated.*

Proof. Suppose that the set A meets the range of all total α-change tests. We show that the degree of A is not α-c.a. dominated. Then because the property of being not α-c.a. dominated is upwards closed in the Turing degrees, the proposition will hold.

Let $\langle\langle f_{i,s}, o_{i,s}\rangle_{s<\omega}\rangle_{i<\omega}$ be an effective list of all tidy $(\alpha+1)$-computable approximations whose limits $f_i = \lim_s f_{i,s}$ consist of all α-c.a. functions. Let p_A be the principal function of A. We show that if i is such that f_i is a total function, then there is some $n \in \omega$ such that $p_A(n) \geqslant f_i(n)$.

For all $i \in \omega$, we define the α-change test $\langle g_{i,s}, p_{i,s}\rangle_{s<\omega}$ as follows. For all $s \in \omega$ and $n \in \omega$, if $o_{i,s}(m) < \infty$ for all $m \leqslant n$, then we define $g_{i,s}(\sigma) = \sigma\,\hat{}\,0^{f_{i,s}(n)}$ and $p_{i,s}(\sigma) = o_{i,s}(n)$ for all strings σ of length n, and otherwise we define $g_{i,s}(\sigma) = \sigma$ and $p_{i,s}(\sigma) = \infty$ for all strings σ of length n.

Let i be such that f_i is a total function. Then using the fact that $\langle f_{i,s}, o_{i,s}\rangle$ is a tidy $(\alpha+1)$-computable approximation, it is straightforward to verify that $\langle g_{i,s}, p_{i,s}\rangle_{s<\omega}$ is a total α-change test. Let σ be such that $A \succ \lim_s g_{i,s}(\sigma)$. If σ has length n, then we have $\lim_s g_{i,s}(\sigma) = \sigma\,\hat{}\,0^{f_i(n)}$, and so $p_A(n) \geqslant f_i(n)$. □

Proposition 4.10. *Suppose the Turing degree* $\mathbf{a}$ *computes a uniformly α-change generic degree. Then* $\mathbf{a}$ *is not uniformly α-c.a. dominated.*

Proof. Let $h : \omega \to \alpha$ be some computable α-order function, and let $\langle\langle f_{i,s}, o_{i,s}\rangle_{s<\omega}\rangle_{i<\omega}$ be an effective list of all tidy $(h+1)$-computable approximations whose limits $f_i = \lim_s f_{i,s}$ consist of all h-c.a. functions. The rest of the proof follows the proof of the previous proposition. □

To show that every degree with a certain domination property is able to compute a multiply generic degree of a certain kind, we use forcing. These constructions can be seen as refinements of the construction of Jockusch and Posner [16] who showed that every degree which is not generalised low_2 computes a 1-generic set.

Theorem 4.11. *Every not α-c.a. dominated degree computes a weakly α-change generic degree.*

Proof. Let $\langle\langle f_{i,s}, o_{i,s}\rangle_{s<\omega}\rangle_{i<\omega}$ be an effective list of all α-change tests. For $t \in \omega$, let

$$\operatorname{range} f_{i,t} = \{f_{i,t}(\sigma) : o_{i,t}(\sigma) < \infty\}.$$

Let $\mathbf{a}$ be a not α-c.a. dominated degree, and let $g \leqslant_{\mathrm{T}} \mathbf{a}$ be a function which is not dominated by any α-c.a. function. We may assume that g is strictly increasing. We build a set G by finite extensions $\sigma_0 \preccurlyeq \sigma_1 \preccurlyeq \cdots$. We say that e *requires attention* at stage s if σ_{s-1} does not meet $\operatorname{range} f_{e,g(s)}$, but there is some τ such that $o_{e,g(s)}(\tau) < \infty$ and $f_{e,g(s)}(\tau) \succ \sigma_{s-1}$.

Construction

Stage 0: let σ_0 be the empty string, and proceed to the next stage.

Stage s, $s \geqslant 1$:

Case 1: there is some $e \leqslant s$ which requires attention at stage s. We choose the least such e, and the least such τ for this e. We let σ_s be the initial segment of $f_{e,g(s)}(\tau)$ of length s. We say that we *act for e* at stage s. We proceed to the next stage.

Case 2: otherwise. We let $\sigma_s = \sigma_{s-1}\hat{\ }0$ and proceed to the next stage.

End of construction

The construction is carried out computably in g, and so $G \leqslant_{\mathrm{T}} \mathbf{a}$.

Lemma 4.12. *G meets the range of every total α-change test.*

Proof. Suppose that $\langle f_{e,s}, o_{e,s}\rangle_{s<\omega}$ is a total α-change test. Assume by induction that we do not act for any $d < e$ after stage s^*.

We define a total function $p : 2^{<\omega} \to \omega$ as follows. Let $p(\sigma)$ be the least t such that for all $s \geqslant t$, $o_{e,s}(\sigma) = o_{e,t}(\sigma)$. We claim that p is α-computably approximable. Let t_σ be the least stage t such that $o_{e,t}(\sigma) < \infty$. Let

$$p_s(\sigma) = \begin{cases} t_\sigma & \text{if } s < t_\sigma \\ (\mu t)(o_{e,t}(\sigma) = o_{e,s}(\sigma)) & \text{if } s \geqslant t_\sigma \end{cases}$$

and let

$$u_s(\sigma) = \begin{cases} o_{e,t_\sigma}(\sigma) & \text{if } s < t_\sigma \\ o_{e,s}(\sigma) & \text{if } s \geqslant t_\sigma. \end{cases}$$

Then $\langle p_s, u_s\rangle_{s<\omega}$ is an α-computable approximation for p.

Let $p'(n) = \max\{\, p(\sigma) : \sigma \in 2^n \,\}$. Then p' is α-computably approximable too. For $p'_s(\sigma) = \max\{\, p_s(\sigma) : \sigma \in 2^n \,\}$ and $u'_s(\sigma) = \bigoplus\{\, u_s(\sigma) : \sigma \in 2^n \,\}$, we have that $\langle p'_s, u'_s\rangle_{s<\omega}$ is an α-computable approximation for p'.

Therefore, the function g escapes domination by p'. Let $t > s^*$ be least such that $g(t) > p'(t)$. Note that for all $s \in \omega$, $|\sigma_s| = s$. At stage t, we will act for e, and we will continue to act for e at all subsequent stages until some stage $t' \geqslant t$ where $\sigma_{t'}$ meets $\operatorname{range} f_e$.

$\square$

□

Theorem 4.13. *Every Turing degree computes a weakly α-change generic if and only if it is not α-c.a. dominated.*

Proof. By Proposition 4.9 and Theorem 4.11. □

Theorem 4.14. *Every not uniformly α-c.a. dominated degree computes a uniformly α-change generic degree.*

Proof. Let $h : 2^{<\omega} \to \alpha$ be a computable α-order function, and let $\langle\langle f_{i,s}, o_{i,s}\rangle_{s<\omega}\rangle_{i<\omega}$ be an effective list of all total h-bounded α-change tests.

We define a total function $r : \omega \times 2^{<\omega} \to \omega$ as follows. Let $r(e, \sigma)$ be the least t such that for all $s \geqslant t$, $o_{e,s}(\sigma) = o_{e,t}(\sigma)$. Then r is h-computably approximable and so certainly α-computably approximable. Let $r'(n) = \max\{\, r(e, \sigma) : e \leqslant n, |\sigma| \leqslant n \,\}$. Then r' is α-computably approximable too.

Let $\mathbf{a}$ be a not uniformly α-c.a. dominated degree and let $g \leqslant_{\mathrm{T}} \mathbf{a}$ be a function which escapes domination by r'. Then as before, we run the construction from the previous theorem to construct a set $G \leqslant \mathbf{a}$. The proof that G meets the range of every total h-bounded α-change test is straightforward.

□

Theorem 4.15. *Every Turing degree computes a uniformly α-change generic if and only if it is not uniformly α-c.a. dominated.*

Proof. By Proposition 4.10 and Theorem 4.14. □

We are now able to give the first of our theorems which separate levels in the hierarchy of multiple genericity notions.

Theorem 4.16. *Let $\alpha \leqslant \varepsilon_0$ be a power of ω. Then there is a Δ^0_2 Turing degree which is uniformly α-change generic but not weakly α-change generic.*

Proof. By Theorem 3.5(2) of [8], there is a c.e. degree $\mathbf{a}$ which is totally α-c.a. but not uniformly totally α-c.a., and by Theorem 5.2 and Theorem 5.4 of [8], $\mathbf{a}$ is α-c.a. dominated but not uniformly α-c.a. dominated. Then by Theorem 4.14, $\mathbf{a}$ computes a uniformly α-change generic degree $\mathbf{b}$. As $\mathbf{a}$ is α-c.a. dominated and $\mathbf{b} \leqslant_{\mathrm{T}} \mathbf{a}$, $\mathbf{b}$ is α-c.a. dominated. Then by Proposition 4.9, $\mathbf{b}$ cannot be weakly α-change generic. □

5. C.e. degrees computing multiply generics

In terms of domination, the best result for computing 1-generics is that every array noncomputable degree computes a 1-generic set. However, every noncomputable c.e. degree computes a 1-generic set. Therefore, the assumption that the degree is c.e. provides extra strength. A similar situation occurs for the notions of multiple genericity. If we assume that the degree is c.e., then we can improve Theorem 4.11 to compute an α-change generic degree.

Theorem 5.1. *Every not α-c.a. dominated c.e. degree computes an α-change generic degree.*

Proof. Let $\mathbf{a}$ be a not ω-c.a. dominated c.e. degree. Because $\mathbf{a}$ is a c.e. degree, by Theorem 5.2 of [8], $\mathbf{a}$ is not totally α-c.a. Recall that this means that there is some function $g \leqslant_{\mathrm{T}} \mathbf{a}$ which is not α-c.a. Let $A \in \mathbf{a}$ be a c.e. set and let Γ be a Turing functional such that $\Gamma(A) = g$.

We first fix some technicalities. Suppose $\langle \Gamma_s \rangle_{s<\omega}$ is a computable enumeration of Γ and $\langle A_s \rangle_{s<\omega}$ is a computable enumeration of A. We assume that if $\Gamma_s(A_s, n)\downarrow$, then $\Gamma_s(A_s, m)\downarrow$ for all $m < n$. From the computable enumerations of Γ and A, we can produce a Δ^0_2-approximation to g as follows. For all $s, n \in \omega$, if $\Gamma_s(A_s, n)\downarrow$, then we set $g_s(n) = \Gamma_s(A_s, n)$, and otherwise we set $g_s(n) = 0$. If $\Gamma_s(A_s, n)\downarrow$, then as usual, we let $\gamma_s(n)$ be the length of the least $\alpha \prec A_s$ such that $\Gamma_s(\alpha, n)\downarrow$. If $\Gamma_s(A_s, n)\uparrow$, then we let $\gamma_s(n) = \gamma_s(m)$ for $m < n$ greatest such that $\Gamma_s(A_s, m)\downarrow$, or $\gamma_s(n) = 0$ if there is no such m.

Let $\langle f_e \rangle_{e<\omega} = \langle \langle f_{e,s}, o_{e,s} \rangle_{s<\omega} \rangle_{e<\omega}$ be an effective list of all α-change tests. We build a Turing functional Δ and meet for every $e \in \omega$ the requirement

$$R_e : \Delta(A) \text{ either meets or avoids range}\, f_e.$$

We begin by considering the strategy to satisfy the requirement R_e in isolation, and in the simplified case where for all $\sigma \in 2^{<\omega}$ and $s \in \omega$, if $o_{e,s}(\sigma) < \infty$, then $o_{e,s}(\sigma) = 0$. That is, range f_e is simply a c.e. set of strings. At every stage of the construction, we will have a finite sequence of natural numbers which we call *lengths*. If l_i is the i^{th} length in our sequence at stage s, then the i^{th} substrategy for R_e will begin by looking for a string τ with $\tau \succ \Delta_{s-1}(A_s) \upharpoonright l_i$ and such that $o_{e,s}(\tau) < \infty$. If we find such a τ (we say that l_i is *realised*), then we would like to define Δ such that $\Delta(A) \succ f_{e,s}(\tau)$, but may require changes in the approximation to A in order to allow us to consistently make such a definition for Δ. We know that A computes the function g which is not α-c.a. function, and so we look for changes in A which are used to compute the various approximations $g_t(n)$ to $g(n)$.

More precisely, whenever a new length is defined, we assign to it a *permitting number*; in this instance we assign l_i the permitting number i. We will then grant permission for l_i at stage s if we see that $g_s(i) \neq g_{s-1}(i)$, and hence $A_s \upharpoonright \gamma_{s-1}(i) \neq A_{s-1} \upharpoonright \gamma_{s-1}(i)$. If permission is granted, then since $\langle A_s \rangle_{s<\omega}$ is a computable enumeration, for all $t < s$ and all $s' \geqslant s$, we have that $A_{s'} \upharpoonright \gamma_{s-1}(i) \neq A_t \upharpoonright \gamma_{s-1}(i)$; that is, the permission can never be retracted. Therefore, we are able to consistently define $\Delta_s(A_s \upharpoonright \gamma_{s-1}(i)) = f_{e,s}(\tau)$, and R_e will be permanently satisfied. As we may never receive permission on any of the lengths already in our sequence, we choose a fresh large number to be a new length, and assign it a permitting number. If there are infinitely many realised lengths, none of which receive permission, then we derive a contradiction to g being not α-c.a. as follows. Suppose l_i is realised at stage s_i. As we do not receive permission for l_i after stage s_i, we know that the approximation to $g(i)$ cannot change past this stage. Thus we can computably bound the number of times the approximation to $g(i)$ can change, which is a contradiction. In fact, in this simplified case, we have that $A \upharpoonright \gamma_{s_i}(i) = A_{s_i} \upharpoonright \gamma_{s_i}(i)$ for all i, which contradicts A being noncomputable.

We now consider the general case, where we no longer necessarily have that $o_{e,s}(\sigma) < \infty$ implies $o_{e,s}(\sigma) = 0$. This will mean that a single length may require multiple permissions, as we now discuss. We begin as above, with the length l_0 with permitting number 0. Suppose at stage s, l_0 is realized when we see that there is $\tau \succ \Delta_{s-1}(A_s) \upharpoonright l_0$ with $o_{e,s}(\tau) < \infty$. We associate the string τ with the length l_0 by letting $\tau_0 = \tau$. As above, while we wait to receive permission for l_0, we will choose a new length l_1 with permitting number 1. If we were to ever receive permission for l_0 at some later stage t, then we would define Δ_t such that

$\Delta_t(A_t) > f_{e,t}(\tau_0)$. In the simplified case from before, this action would have been enough to permanently satisfy R_e. Here, in the general case, at some later stage u we may have that $f_{e,u}(\tau_0) \neq f_{e,t}(\tau_0)$, in which case R_e would no longer appear to be satisfied at stage u. In order to consistently define Δ so that $\Delta(A) > f_{e,u}(\tau_0)$, we may require another permission for l_0. If there are infinitely many realised lengths, none of which receive as many permissions as they require, then we show that we can build an α-computable approximation for g, which is a contradiction.

Suppose that we do see the stage u as in the previous paragraph. We must decide how to proceed. We request another permission for l_0, and since l_0 still has permitting number 0, l_0 will receive permission if we see a change in the approximation to $g(0)$ at some later stage. While waiting for l_0 to receive permission, we must define a new length, and assign it a permitting number. The first candidate we might consider for the new length is the old length l_1, which was defined earlier at stage s. We would like to be able to argue that if the new length is never realised, then $\Delta(A)$ avoids range f_e. However, we may have $|f_{e,t}(\tau_0)| > l_1$, so if we use l_1 for the new length, we cannot make this argument. Therefore, we choose to *clear* l_1 from the sequence of lengths when l_0 receives permission, and at stage u we will define the new length to be a fresh large number. We write this now indexed by the stage number as $l_{1,u}$.

Now we must assign $l_{1,u}$ a permitting number. Again, the first candidate for the permitting number of $l_{1,u}$ is the permitting number of the original l_1, namely 1. Recall that in order to show that we do eventually receive enough permissions for some length, we must define an α-computable approximation, call it $\langle h_s, q_s\rangle_{s<\omega}$, for the function g. The values of $q_s(n)$ will depend on the values of $o_{e,s}(\tau_i)$, where τ_i is the string found at the stage when l_i is realised. In particular, if the original length l_1 is realised at a stage t' *before* we receive the first permission for l_0, then if τ_1 is the string found at t', we will define $q_{t'}(1) = o_{e,t'}(\tau_0) \oplus o_{e,t'}(\tau_1)$. However, if $l_{1,u}$ is realised at some stage $v > u$ when we find the string $\tau_{1,v}$, we may not be able to redefine $q_v(1)$ to incorporate the value $o_{e,v}(\tau_{1,v})$, because the sequence $\langle q_s(n)\rangle_{s<\omega}$ must be non-increasing. We therefore must assign $l_{1,u}$ the permitting number 2.

There is one last complication. At stage v, we have the lengths l_0 and $l_{1,u}$, both waiting for permission, with permitting numbers 0 and 2, respectively. What should we do if we see at some later stage w that $g_w(1) \neq g_{w-1}(1)$? We would like to use changes in the approximations to the values of $g(n)$, and so changes in A, as much as possible. So it would be wasteful to not allow l_0 to receive permission when we see such a change in A. If we think in terms of the approximation $\langle h_s, q_s\rangle_{s<\omega}$, we must have $\lim_s h_s(n) = g(n)$ for *all* $n \in \omega$. Our solution is to simply allow l_0 to receive permission when we see a change in the approximation to either $g(0)$ or $g(1)$ after stage t. In effect, the permitting number has become a *permitting interval*. We ensure that if the natural number n is in the permitting interval of some length at some stage, then at all later stages, it is in the permitting interval of some (possibly different) length. Then the approximation $\langle h_s, q_s\rangle_{s<\omega}$ will correctly approximate g. We now turn to the formal details for the construction.

At every stage $s \in \omega$ and for every requirement R_e, we may define the natural number $l_{e,i,s}$ for some $i \in \omega$. If $l_{e,i,s}$ is defined, then we say that $l_{e,i,s}$ is a *length* for R_e at stage s. If $l_{e,i,s}$ is defined, then we will also have $l_{e,j,s}$ defined for all $j < i$. If $l_{e,i,s}$ is defined, then we may also define a string $\tau_{e,i,s}$. If we say at stage s that we *clear* a length $l_{e,i,s-1}$, then we let $l_{e,i,s}$ and $\tau_{e,i,s}$ be undefined. The requirement R_e is initialised at stage s by clearing all lengths for R_e.

To every length $l_{e,i,s}$ we associate a *permitting interval* $I_s(l_{e,i,s})$. This is an interval of natural numbers; its left end is fixed from when $l_{e,i,s}$ is first defined, but its right end may grow with time (but only finitely often). We say that $l_{e,i,s-1}$ is *permitted* at stage s if $l_{e,i,s-1}$ is waiting for permission at stage s, and for some $n \in I_{s-1}(l_{e,i,s-1})$ we have $g_s(n) \neq g_{s-1}(n)$.

We use the following priority ordering among the R-requirements:

$$R_0 < R_1 < R_2 < \cdots .$$

We say that R_e *requires attention at stage* s if either

(1) $l_{e,0,s-1}$ is undefined,
(2) there is some length $l_{e,i,s-1}$ which is permitted at stage s,
(3) (a) we acted in Case 2 for R_e for some $l_{e,i,t}$ at some previous stage t,
 (b) $l_{e,i,s-1} = l_{e,i,t}$ and $\tau_{e,i,s-1} = \tau_{e,i,t}$, but
 (c) $f_{e,s}(\tau_{e,i,s-1}) \neq f_{e,t}(\tau_{e,i,s-1})$, or
(4) for some i such that $l_{e,i,s-1}$ is not realised at the beginning of stage s, there is some τ such that $\tau \succcurlyeq \Delta_{s-1}(A_s) \upharpoonright l_{e,i,s-1}$ and $o_{e,s}(\tau) < \infty$.

We note that the i as in Case 3 and Case 4 must be greatest such that $l_{e,i,s-1}$ is defined.

Construction

Stage 0: for every $n \in \omega$, we let $\Delta_0(A_0 \upharpoonright \gamma_0(n)) = 0^n$. We proceed to the next stage.

Stage s, $s \geqslant 1$: we follow the instructions in Step 1 and Step 2, and then proceed to the next stage.

Step 1

We choose the least e such that R_e requires attention at stage s, act according to the cases below, initialise all requirements of weaker priority than R_e, and then proceed to Step 2.

Case 1: R_e requires attention via (1) at stage s. We define $l_{e,0,s}$ to be a fresh large number and let $I_s(l_{e,0,s}) = \{0\}$.

Case 2: R_e requires attention via (2) at stage s. We choose the least i as in the definition of requires attention. Let n be the greatest element of any permitting interval for any length for R_e at the beginning of stage s. We set $\Delta_s(A_s \upharpoonright \gamma_s(n)) = f_{e,s}(\tau_{e,i,s-1})$. We let $l_{e,i,s} = l_{e,i,s-1}$, and set $I_s(l_{e,i,s}) = [\min I_{s-1}(l_{e,i,s}), n]$. We clear all lengths $l_{e,j,s-1}$ for all $j > i$. We do not say that $l_{e,i,s}$ is waiting for permission at stage $s+1$. We say that we act in Case 2 for R_e via $l_{e,i,s}$ at stage s.

Case 3: R_e requires attention via (3) at stage s. Let i be as in the definition of requires attention. For all $j \leqslant i$, we let $l_{e,j,s} = l_{e,j,s-1}$ and let $I_s(l_{e,j,s}) = I_{s-1}(l_{e,j,s-1})$. We say that $l_{e,i,s}$ is waiting for permission at stage $s+1$. We define $l_{e,i+1,s}$ to be some fresh large number, and let $I_s(l_{e,i+1,s}) = \{n\}$, where n is the least natural number which is not in any $I_s(l_{e,j,s})$ for $j \leqslant i$. We say that we act in Case 3 for R_e for $l_{e,i,s}$ at stage s.

Case 4: R_e requires attention via (4) at stage s. Let i be as in the definition of requires attention. We choose the least such τ as in the definition of requires attention, and define $\tau_{e,i,s} = \tau$. For all $j \leqslant i$, we let $l_{e,j,s} = l_{e,j,s-1}$ and let

$I_s(l_{e,j,s}) = I_{s-1}(l_{e,j,s-1})$. We say that $l_{e,i,s}$ has been realised at stage s. We say that $l_{e,i,s}$ is waiting for permission at stage $s+1$. We define $l_{e,i+1,s}$ to be some fresh large number, and let $I_s(l_{e,i+1,s}) = \{n\}$, where n is the least natural number which is not in any $I_s(l_{e,j,s})$ for $j \leqslant i$. We say that we act in Case 4 for R_e for $l_{e,i,s}$ at stage s.

Step 2

If we did not act in Case 2 in Step 1, then for every $n \in \omega$, if $A_s \upharpoonright \gamma_s(n) \notin \operatorname{dom} \Delta_{s-1}$ and β is such that $\Delta_{s-1}(A_{s-1} \upharpoonright \gamma_{s-1}(n)) = \beta$, we set $\Delta_s(A_s \upharpoonright \gamma_s(n)) = \beta$.

If we did act in Case 2 in Step 1, then for n as in Case 2 and all $m \geqslant 1$, we set $\Delta_s(A_s \upharpoonright \gamma_s(n+m)) = f_{e,s}(\tau_{e,i,s-1})\hat{\ }0^m$.

End of construction

By the consistency of the functional Γ, if $g_{s-1}(n) \neq g_s(n)$, then we must have $A_s \upharpoonright \gamma_{s-1}(n) \neq A_{s-1} \upharpoonright \gamma_{s-1}(n)$. Using this fact, a straightforward induction shows that the functional Δ is consistent.

Lemma 5.2. *Each requirement is met.*

Proof. Assume by induction that R_e is initialised for the last time at stage s^*. Suppose for contradiction that we act for R_e at infinitely many stages. We build a tidy $(\alpha+1)$-computable approximation $\langle h_s, q_s \rangle_{s<\omega}$ for g. We will in addition show that $\langle h_s, q_s \rangle_{s<\omega}$ is eventually α-computable. That is, for all $n \in \omega$, there is some $s \in \omega$ such that $q_s(n) < \infty$. By Lemma 1.6 of [8], this shows that g is α-c.a.

For all $n \in \omega$ and all $s \leqslant s^*$, we set $h_s(n) = 0$ and $q_s(n) = \infty$. For all $s > s^*$ and $n \in \omega$, if we act in Case 3 or Case 4 for R_e for some $l_{e,i,s}$ at stage s such that $n \in I_s(l_{e,i,s})$, then we set $h_s(n) = g_s(n)$ and

$$q_s(n) = \bigoplus_{j \leqslant i} o_{e,s}(\tau_{e,j,s}) \oplus (i+1).$$

We say that we update the approximation to $h(n)$ at stage s (though we may define $h_s(n) = h_{s-1}(n)$). Otherwise, we set $h_s(n) = h_{s-1}(n)$ and $q_s(n) = q_{s-1}(n)$.

Suppose that i and s_0 are such that $l_{e,i,s_0} = l_{e,i}$ is never cleared after stage s_0. We show that there is some stage $s_1 \geqslant s_0$ at which we act in Case 4 for R_e for $l_{e,i}$. By our assumption on i and s_0, for all $t \geqslant s_0$, we cannot act in Case 2 for R_e for any $l_{e,j,t}$ at stage t where $j < i$. A straightforward induction shows that if $l_{e,j,s}$ is defined but there is some $k > j$ such that $l_{e,k,s}$ is defined, then $l_{e,j,s}$ must be waiting for permission at stage s. Therefore, for all $t \geqslant s_0$, we cannot act in Case 3 for R_e for any $l_{e,j,s}$ at stage t where $j < i$. As l_{e,i,s_0} is defined, we cannot act in Case 1 for R_e after stage s_0. As we must act for R_e at infinitely many stages after stage s^*, we must then act in Case 4 for R_e for $l_{e,i}$ at some stage $s_1 \geqslant s_0$.

Now suppose that i and s_0 are such that $l_{e,i,s_0} = l_{e,i}$ is never cleared after stage s_0, and that $l_{e,i}$ was realised at some stage $s_1 \geqslant s_0$. Further suppose that we act in Case 2 for R_e for $l_{e,i}$ at some stage $s_2 > s_1$. We show that we must act in Case 3 for R_e for $l_{e,i}$ at some stage $s_3 > s_2$. Suppose for contradiction that this is not the case. As in the proof in the previous paragraph, we cannot act in Case 2 or Case 3 for R_e for any $l_{e,j,t}$ with $j < i$ at any stage $t \geqslant s_2$. At stage s_2 we cleared l_{e,j,s_2-1} for all $j > i$. Therefore, by assumption, we cannot act in any case for R_e for any $l_{e,j,t}$ with $j > i$ at any stage $t \geqslant s_2$. We also say at stage s_2 that $l_{e,i}$ is not waiting for permission at stage s_2+1, and because we do not act in Case 3 for R_e for $l_{e,i}$ at any later stage s_3, we cannot act again in Case 2 for R_e for $l_{e,i}$ at any stage $t > s_2$. Therefore, we do not act for R_e after stage s_2, which is a contradiction.

We now show that $\langle h_s, q_s\rangle_{s<\omega}$ is eventually α-computable. By the definition of $q_s(n)$ above and the definition of the permitting intervals $I_s(l_{e,i,s})$ in the construction, it suffices to show that there are infinitely many stages at which we act in either Case 3 or Case 4 for R_e. Suppose for contradiction we do not act in Case 3 or Case 4 after some stage $s > s^*$. As R_e is not initialised after stage s^*, there is some i such that $l_{e,i,s}$ is never cleared after stage s. Let i the greatest such, and let $l_{e,i} = l_{e,i,s}$. There are three possibilities to consider. The first is that $l_{e,i}$ has not been realised by stage s. Then as shown above, we must act in Case 4 for R_e for $l_{e,i}$ at some stage $s_1 > s$, which is a contradiction. The second is that $l_{e,i}$ has been realised by stage s, and $l_{e,i}$ is waiting for permission at stage s. Then it must be the case that there is some $j > i$ such that $l_{e,j,s}$ is defined. By the choice of i, there is some stage $t > s$ at which $l_{e,j,t-1}$ is cleared, and at the least such stage t, we must act in Case 2 for R_e for $l_{e,i}$ at stage t. Then as shown above, we must act in Case 3 for R_e for $l_{e,i}$ at some stage $s_2 > s$, which is a contradiction. The last possibility is that $l_{e,i}$ has been realised by stage s, but $l_{e,i}$ is not waiting for permission at stage s. Then again by the above, we must act in Case 3 for R_e for $l_{e,i}$ at some stage $s_2 > s$, which is a contradiction.

It follows from the fact that the sequence $\langle o_{e,s}(n)\rangle_{s<\omega}$ is non-increasing for all $n \in \omega$ and the definition of $I_s(l_{e,i,s})$ in Case 2 of the construction that $\langle q_s(n)\rangle_{s<\omega}$ is non-increasing for all $n \in \omega$. We must now show that for all $n \in \omega$ and all $s > 0$ that

(1) if we update the approximation to $h(n)$ at stage s, then $q_s(n) < q_{s-1}(n)$.

Fix some $n \in \omega$. Let $s_0 < s_1 < \cdots$ be the potentially infinite sequence of all stages at which we update the approximation to $h(n)$. We have $q_0(n) = \infty$ and $q_{s_0}(n) < \infty$, so (1) holds at stage s_0. Suppose by induction that (1) holds at stage s_k, and that s_{k+1} is defined. We show that (1) holds at stage s_{k+1}. Suppose that i and j are such that we act via l_{e,i,s_k} at stage s_k and via $l_{e,j,s_{k+1}}$ at stage s_{k+1}. If $j < i$, then the inclusion of the last term in the definition of $q_{s_{k+1}}(n)$ is enough to ensure that (1) holds at stage s_{k+1}. Now suppose $j = i$. We must have $l_{e,i,t} = l_{e,i,s_k}$ and $\tau_{e,i,t} = \tau_{e,i,s_k}$ for all t with $s_k < t \leqslant s_{k+1}$. As we act in Case 3 for l_{e,i,s_k} at stage s_k and stage s_{k+1}, we must act in Case 2 for l_{e,i,s_k} at some stage t with $s_k < t < s_{k+1}$. At stage t we defined Δ_t such that $\Delta_t(A_t) \succ f_{e,t}(\tau_{e,i,t})$, and as we act in Case 3 for l_{e,i,s_k} at stage s_{k+1}, we must have $f_{e,s_{k+1}}(\tau_{e,i,s_k}) \neq f_{e,t}(\tau_{e,i,s_k})$, and so $o_{e,s_{k+1}}(\tau_{e,i,s_k}) < o_{e,t}(\tau_{e,i,s_k}) \leqslant o_{e,s_k}(\tau_{e,i,s_k})$. This is enough to ensure that (1) holds at stage s_{k+1}.

Finally, we show that $\langle h_s, q_s\rangle_{s<\omega}$ is an approximation for g. That is, for all $n \in \omega$, $\lim_s h_s(n) = g(n)$. Fix some $n \in \omega$. The fact that (1) holds at all stages together with the well-foundedness of the ordinal α shows that we update the approximation to $h(n)$ at only finitely many stages. Let t be the last such stage. We claim that $g(n) = h_t(n)$. At stage t we set $h_t(n) = g_t(n)$, so it suffices to show that $g_u(n) = g_t(n)$ for all $u > t$. Suppose for contradiction that $g_u(n) \neq g_t(n)$ for some $u > t$, and let u be the least such. Let i be such that $n \in I_t(l_{e,i,t})$. By the choice of u, we have $l_{e,i,u} = l_{e,i,t}$ and $n \in I_u(l_{e,i,u})$. Then for all $j \leqslant i$, $l_{e,j,u}$ is waiting for permission at stage u. At all stages s, if m is the greatest element of any permitting interval for any length for R_e at stage s, then every $l \leqslant m$ is in some permitting interval for some length for R_e at stage s. Therefore, for some $j \leqslant i$, $l_{e,j,u-1}$ is permitted at stage u, and we act in Case 2 for R_e for $l_{e,j,u}$ at stage u. As R_e is not initialised after stage s^*, there is some $k \leqslant i$ and some v such that $n \in I_w(l_{e,k,w})$ for all $w \geqslant v$. Let v be the least such. By assumption, at any stage y with $t < y \leqslant v$, we cannot act in Case 3 for R_e for any length $l_{e,j,y}$ with $n \in I_y(l_{e,j,y})$. By the choice of k, $l_{e,k,v} = l_{e,k}$ is never cleared. Then as shown above, we must act in Case 3 for R_e for $l_{e,k}$ at some stage $z > v$. We have $n \in I_z(l_{e,k})$, so we will update the approximation to $h(n)$ at stage z. This is a contradiction. This completes the proof that we act for R_e at finitely many stages.

We are now in a position to show that $\Delta(A)$ either meets or avoids range f_e. Let t be the last stage at which we act for R_e, and let i be greatest such that $l_{e,i,t}$ is defined. First suppose that we act in Case 1, Case 3, or Case 4 for R_e at stage t. Then $l_{e,i,t}$ is not realised at stage t, and because we do not act for R_e after stage t, $l_{e,i,t}$ is never realised. We initialise all requirements of weaker priority than R_e at stage t, and we do not act for R_e after stage t, so the construction will ensure that $\Delta_t(A_t) \restriction l_{e,i,t} \prec \Delta_u(A_u)$ for all $u > t$. Then because $l_{e,i,t}$ is never realised, there is no τ such that $\tau \succcurlyeq \Delta_t(A_t) \restriction l_{e,i,t}$ and $o_{e,u}(\tau) < \infty$. Therefore, by Proposition 2.12, $\Delta(A)$ avoids range f_e. Finally, suppose that we act in Case 2 for R_e at stage t. We define Δ_t such that $\Delta_t(A_t) \succ f_{e,t}(\tau_{e,i,t})$. We initialise all requirements of weaker priority than R_e at stage t, and we do not act for R_e after stage t, so the construction will ensure that $\Delta_u(A_u) \succ f_{e,t}(\tau_{e,i,t})$ for all $u > t$. Then because we do not act in Case 3 for R_e for $l_{e,i,u}$ at any stage $u > t$, we have $f_{e,u}(\tau_{e,i,t}) = f_{e,t}(\tau_{e,i,t})$ for all $u > t$, and so $\Delta(A)$ meets range f_e.

□

□

We now have the following characterisation.

Theorem 5.3. *A c.e. degree computes an α-change generic degree if and only if it is not totally α-c.a.*

Proof. Theorem 5.2 of [8] says that a c.e. degree is totally α-c.a. if and only if it is α-c.a. dominated. Suppose that $\mathbf{a}$ is a c.e. degree which computes an α-change generic degree $\mathbf{b}$. Then $\mathbf{b}$ is weakly α-change generic, and so by Theorem 4.11, $\mathbf{a}$ is not α-c.a. dominated. The other direction follows from the previous theorem. □

Theorem 5.1 allows us to separate further levels in the hierarchy of multiple genericity notions.

Theorem 5.4. *Let $\alpha \leqslant \varepsilon_0$ be a power of ω and let β be such that $\alpha = \omega^\beta$. Then there is a Δ^0_2 degree which is ω^β-change generic but not uniformly $\omega^{\beta+1}$-change generic.*

Proof. By Theorem 3.5(1) of [8], there is a c.e. degree $\mathbf{a}$ which is uniformly totally $\omega^{\beta+1}$-c.a., but not totally ω^β-c.a., and by Theorem 5.2 and Theorem 5.4 of [8], $\mathbf{a}$ is uniformly $\omega^{\beta+1}$-c.a dominated and not ω^β-c.a. dominated. Then by Theorem 5.1, $\mathbf{a}$ computes an ω^β-change generic degree $\mathbf{b}$. As $\mathbf{b} \leqslant_{\mathrm{T}} \mathbf{a}$ and $\mathbf{a}$ is uniformly $\omega^{\beta+1}$-c.a. dominated, $\mathbf{b}$ is uniformly $\omega^{\beta+1}$-c.a. dominated. Now by Proposition 4.10, $\mathbf{b}$ cannot be uniformly $\omega^{\beta+1}$-change generic. □

6. Downward density

We say that a class of degrees $\mathcal{D}$ is *downwards dense* below a degree $\mathbf{a}$ if for every noncomputable degree $\mathbf{b} \leqslant_{\mathrm{T}} \mathbf{a}$, there is some $\mathbf{c} \in \mathcal{D}$ with $\mathbf{c} \leqslant_{\mathrm{T}} \mathbf{b}$. There are several results regarding downward density and genericity. Martin showed (see Theorem 4.1 in [15]) that for all $n \geqslant 2$, the n-generic degrees are downwards dense below n-generic degrees. The case for $n = 1$ is more involved though. Chong and Jockusch [5] showed that the 1-generic sets are downwards dense below Δ^0_2 1-generic sets. However, it was later shown by Chong and Downey [4], and independently Kumabe [17], that there is a 1-generic degree below $\varnothing''$ which bounds a minimal degree, and so downward density of 1-generics below 1-generic degrees fails in general.

Downey and Nandakumar [13] have recently shown that there is a weakly 2-generic set which bounds a minimal degree, and so downward density cannot hold in general for any notion of multiple genericity. We can still ask whether it holds below $\varnothing'$, as it does for 1-genericity. Schaeffer [24] has shown that there is a pb-generic below $\varnothing'$ which bounds a noncomputable superlow degree. By Proposition 6.3 of [24], no superlow degree can bound a pb-generic degree, and so pb-generics are not downwards dense below pb-generics below $\varnothing'$. We show that downwards density does not hold below $\varnothing'$ at any uniform or weak level in the hierarchy of multiple genericity notions.

Theorem 6.1. *Let $\alpha \leqslant \varepsilon_0$ be a power of ω. Then there is a weakly α-change generic degree $\mathbf{a} \leqslant_{\mathrm{T}} \varnothing'$ which bounds a noncomputable degree $\mathbf{b}$ which does not bound a uniformly ω-change generic degree. Therefore the weakly α-change generic degrees are not downward dense below weakly α-change generic degrees below $\varnothing'$, and similarly for the uniformly α-change generic degrees.*

Proof. We construct a set A and a Turing functional Φ such that $\mathbf{a} = \deg_{\mathrm{T}}(A)$ and $\mathbf{b} = \deg_{\mathrm{T}}(\Phi(A))$ are as required. Let $\langle\langle f_{e,s}, o_{e,s}\rangle_{s<\omega}\rangle_{e<\omega}$ be an effective list of all α-change tests. So that $\mathbf{a}$ is of weakly α-change generic degree, we meet for every $i \in \omega$ the requirement

$$P_i : \text{if } f_i = \langle f_{i,s}, o_{i,s}\rangle_{s<\omega} \text{ is a total } \alpha\text{-change test, then } A \text{ meets range } f_i.$$

We construct Φ such that $\Phi(A)$ is total, and in order to make $\Phi(A)$ noncomputable, we meet for every $i \in \omega$ the requirement

$$N_i : \Phi(A) \neq \varphi_i$$

where $\langle \varphi_i\rangle_{i<\omega}$ is an effective list of all partial computable functions.

Let $h : 2^{<\omega} \to \alpha$ be some computable α-order function with $h(\sigma) > 0$ for all $\sigma \in 2^{<\omega}$. So that $\mathbf{b}$ does not bound any uniformly α-change generic degree, we meet for every $e \in \omega$ the requirement

$$Q_e : \text{if } \Gamma_e(\Phi(A)) \text{ is total, then it does not meet the range of every } h\text{-change test}$$

where $\langle \Gamma_e\rangle_{e<\omega}$ is an effective list of all Turing functionals. To meet Q_e, we build an h-change test $t_e = \langle g_{e,s}, p_{e,s}\rangle_{s<\omega}$ such that if $\Gamma_e(\Phi(A))$ is total, then $\Gamma_e(\Phi(A))$ does not meet range t_e.

The basic strategy for a P-requirement P_i is straightfoward. We pick a follower σ for P_i and wait until a stage s where we see that $o_{i,s}(\sigma) < \infty$. At such a stage s, we say that P_i is *realised*, and we let $A_s > f_{i,s}(\sigma)$. If at some later stage t we see that $f_{i,t}(\sigma) \not\prec A_{t-1}$, then we define A_t to extend $f_{i,t}(\sigma)$. Since f_i is an α-change test, this strategy is finitary.

The basic strategy for an N-requirement N_i is also straightforward. We begin at some stage s by defining some string $\tau_{i,s}$ for N_i. We will have defined only finitely many axioms for Φ by stage s, so $\Phi_{s-1}(A_{s-1})$ is some finite string, say, β. We define $\Phi(\tau_{i,s}) = \beta$. We also define $\Phi(\tau_{i,s}\,\hat{}\,0) = \beta\,\hat{}\,0$. We then wait for some stage t where for all $n \leqslant |\beta| + 1$, $\varphi_{i,t}(n)\downarrow$ and $\Phi_{t-1}(A_{t-1}) = \varphi_i(n)$. At such a stage t, we *act for* N_i by letting A_t extend $\tau_{i,s}\,\hat{}\,1$, and defining $\Phi(\tau_{i,s}\,\hat{}\,1) = \beta\,\hat{}\,1$.

Now suppose that we having one Q-requirement Q_e together with all the N-requirements, with the priority ordering $Q_e < N_0 < N_1 < \cdots$. The action of the N-requirements will ensure that $\Phi(A)$ is total. Suppose that e is such that $\Gamma_e(\Phi(A))$ is total. We will need to build an h-change test t_e such that $\Gamma_e(\Phi(A))$ does not meet the range of t_e.

Suppose that we begin the strategy for N_i at stage s. We keep A above $\tau_{i,s}\,\hat{}\,0$ so that $\Phi(A)$ extends $\beta\,\hat{}\,0$. Suppose we see at some later stage t that there is some $\gamma \in \operatorname{dom}\Gamma_{e,t}$ with $\gamma \preccurlyeq \Phi_{t-1}(A_{t-1})$. It seems as though we should look to update arrows in the test t_e. Therefore, for every string μ such that $\mu \prec \Gamma_e(\gamma)$, we define $g_{e,t}(\mu)$ to be some string properly extending μ that is incomparable with $\Gamma_e(\gamma)$.

If $\gamma \prec \Phi_u(A_u)$ for all $u \geqslant t$, then $\Gamma_e(\gamma) \prec \Gamma_{e,u}(\Phi_u(A_u))$ for all $u \geqslant t$, and so we will never again need to update the arrow for any such μ. Suppose though that at some later stage u, we see that we need to act for N_i. We let $A_u \succ \tau_{i,s}\,\hat{}\,1$ and define $\Phi(\tau\,\hat{}\,1) = \beta\,\hat{}\,1$. However, if $\gamma \not\prec \Phi_u(A_u)$, the opponent is now free to define axioms in Γ_e in such a way that $\Gamma_{e,u}(\Phi_u(A_u)) \succ g_{e,t}(\nu)$, which means that we will again need to update the arrow for ν. Moreover, action for any N-requirement of stronger priority than N_i may also force us to update the arrow for μ. Since t_e must be an h-change test, we must be careful that we do not update the arrow for μ too many times.

In this restricted scenario, it is quite easy to manage this; each N-requirement N_i is able to act at most once and force us to change $\Phi(A)$, requiring at most one update to the arrow for μ due to N_i. In light of this, we revise our strategy for defining the test t_e.

Suppose that we begin our strategy for N_i at some stage s by defining some string $\tau_{i,s}$ and defining $\Phi(\tau_{i,s}) = \beta$, as above. Suppose that there are n many requirements of stronger priority than N_i for which we have not already acted. We seek a string $\nu \preccurlyeq \Phi(A)$ such that $\nu \in \operatorname{dom}\Gamma_e$ and $h(|\nu|) \geqslant n + 1$. Since $\Gamma_e(\Phi(A))$ is total and h is an α-order function, we will eventually see such a stage. Suppose we see such a string ν at stage t.

First suppose that N_i is never initialised after stage t. We would like to restart the strategy N_i so that we may keep $\Phi(A)$ above the string ν. If we can do so, then acting for N_i will cause a change in $\Phi(A)$ above ν, and $\Gamma_e(\Phi(A))$ must remain above $\Gamma_e(\nu)$, which is sufficiently large to accommodate any update to the arrows in the test t_e as a result of action for N_i.

We restart N_i at stage t as follows. We let $A_t = A_{t-1}$, and define $\tau_{i,t}$ to be some intial segment of A_t of a fresh large length. For $\beta = \Phi_{t-1}(A_t)$, we enumerate $\langle \tau_{i,t}, \beta \rangle$ and $\langle \tau_{i,t}\,\hat{}\,0, \beta\,\hat{}\,0 \rangle$ into Φ. Having done this, we then act for N_i at some later stage u if for all $n \leqslant |\beta| + 1$, $\varphi_{i,u}(n)\downarrow$ and $\Phi_{u-1}(A_{u-1}) = \varphi_i(n)$. While waiting to act for N_i, we look at stage u for some string $\gamma \in \operatorname{dom}\Gamma_{e,u}$ with $\beta\,\hat{}\,0 \preccurlyeq \gamma \preccurlyeq \Phi_{u-1}(A_{u-1})$. Only once we see such a string γ do we update arrows in t_e.

In effect, the requirement Q_e is broken into infinitely many Q_e-subrequirements Q_{e,N_i} for each N-requirement N_i. The subrequirement Q_{e,N_i} will look for some string ν as above (waiting for *initial* convergence) and then will look for a string γ as above (waiting for *further* convergence). Only once further convergence has been found will we look to update arrows in t_e.

If at any stage we act for some N-requiement N_j of stronger priority than N_i, we initialise N_i, and we no longer update the arrow for any μ as part of action for Q_{e,N_i}; this responsibility then falls to Q_{e,N_j}.

Now consider introducing the P-requirement P_i, with the priority ordering

$$P_i < \{N_0, Q_{e,N_0}\} < \{N_1, Q_{e,N_1}\} < \cdots$$

and where N_i and Q_{e,N_i} are of equally strong priority for all $i \in \omega$. We begin the strategy for P_e by choosing a follower σ. We proceed with the strategy above, acting for many N-requirements, and updating many arrows in the test t_e. Only after this do we see at some stage s that $o_{e,s}(\sigma) < \infty$.

The problem is readily apparent. We may have updated the arrow for some string μ for which $o_{e,s}(\sigma) > h(|\mu|)$. Acting for P_i is then able to force A to move through a very large number of strings, and because of the action taken for N-requirements, this will force $\Phi(A)$ to move through a very large number of strings. No matter our definition of the arrow for μ in the test t_e, the opponent is able to manoeuvre $\Phi(A)$ and define axioms for Γ_e in such a way that $\Gamma_e(\Phi(A))$ extends the arrow for μ.

Note however that the opponent has revealed to us the value $o_{e,s}(\sigma)$, which gives us some upper bound on the number of times this can occur. Our strategy for Q_{e,P_i} is reminiscent of the strategy we adopted above. We seek a string $\nu \preccurlyeq \Phi(A)$ such that $\nu \in \operatorname{dom}\Gamma_e$ and $h(|\nu|) > o_{e,s}(\sigma)$. Since $\Gamma_e(\Phi(A))$ is total and h is an α-order function, we will eventually see such a stage. Suppose we see such a string ν at stage t.

We would like to restart the strategy for P_i in a way that allows us to keep $\Phi(A)$ above ν. However, if we were to simply follow our current strategy for P_i, we must always let A_u extend $f_{i,u}(\sigma)$, but doing so may force us to move $\Phi(A)$ away from ν. Note that we are only interested in meeting the range of f_i if f_i is a *total* test. Therefore, we are free to define a set $S_i = \{\sigma_0, \sigma_1\}$ of incomparable strings, and wait until we see o_i converge on all strings in S_i. Suppose that we see $o_{i,s}(\sigma) < \infty$ for all $\sigma \in S_i$ at some stage s. We say that P_i is *realised* at stage s. We let A_s extend $f_{i,s}(\sigma_0)$, and will let A_u extend $f_{i,u}(\sigma_0)$ at subsequent stages u.

We slightly modify the strategy for Q_{e,P_i} by now requiring that the string ν is such that $h(|\nu|) > o_{e,s}(\sigma_0) \oplus o_{e,s}(\sigma_1)$. Suppose we see such a string in the domain of Γ_e at stage t. We may now restart P_i by letting A_t extend $f_{i,t}(\sigma_1)$, and will let A_u extend $f_{i,u}(\sigma_1)$ at subsequent stages u. It is straightforward to organise the construction in a way that allows us to then keep $\Phi(A)$ above ν.

The introduction of further P- and Q-requirements poses no significant challenge. We proceed to the formal argument.

Definitions and conventions for the construction

At each stage $s \in \omega$, and for each $i \in \omega$, we define for a set $S_{i,s} \subset 2^{<\omega}$. The requirement P_i is initialised a stage s by setting $S_{e,s} = \varnothing$.

At each stage $s \in \omega$, and for each $i \in \omega$, we may define a string $\tau_{i,s}$ and a natural number $l_{i,s}$. The requirement N_i is initialised at stage s by letting $\tau_{i,s}$ and $l_{i,s}$ be undefined.

For each $e \in \omega$ and each i with $e \leqslant i$, we have the Q_e-subrequirements Q_{e,P_i} and Q_{e,N_i}. At each stage $s \in \omega$, for each $e \in \omega$, and for each P- or N-requirement R, we may define a natural number $c_{e,R,s}$, and strings $\nu_{e,R,s}$ and $\gamma_{e,R,s}$. We may say that *we are waiting for initial convergence for $Q_{e,R}$ at stage s*, or that *we are waiting for further convergence for $Q_{e,R}$ above the string β at stage s* for some string β. The subrequirement $Q_{e,R}$ is initialised at stage s by letting $c_{e,R,s}$, $\nu_{e,R,s}$, and $\gamma_{e,R,s}$ be undefined. Additionally, we do not say that we are waiting for initial or further convergence for $Q_{e,R}$ at stage $s+1$.

As usual, all definitions and statements made at stage s will apply at stage $s+1$ unless otherwise specified.

We use the convention that a commutative ordinal sum over an empty set is equal to zero.

We say that P_i *requires attention at stage s* if either

(1) $S_{i,s-1} = \varnothing$,
(2) $S_{i,s-1} \neq \varnothing$, P_i has not been realised by the beginning of stage s, and for all $\sigma \in S_{i,s-1}$, $o_{i,s}(\sigma) < \infty$, or
(3) P_i has been realised by the beginning of stage s, and there is some $\sigma \in S_{i,s-1}$ such that $\sigma \prec A_{s-1}$ but $f_{i,s}(\sigma) \nprec A_{s-1}$.

We say that N_i *requires attention at stage s* if either

(1) $\tau_{i,s-1}$ is not defined,
(2) $\tau_{i,s-1}$ is defined, and for all $n \leqslant l_{i,s-1}$, $\varphi_{i,s}(n)\downarrow$ and $\Phi_{s-1}(A_{s-1}, n) = \varphi_i(n)$.

We say that $Q_{e,R}$ *requires attention at stage s* if either

(1) we are waiting for initial convergence for $Q_{e,R}$ at stage s, and there is some $\nu \in \operatorname{dom}\Gamma_{e,s}$ such that $\nu \preccurlyeq \Phi_{s-1}(A_{s-1})$ and $|\Gamma_{e,s}(\nu)| \geqslant c_{e,R,s-1}$, or
(2) we are waiting for further convergence for $Q_{e,R}$ above the string β at stage s, and there is some $\gamma \in \operatorname{dom}\Gamma_{e,s}$ such that $\beta \preccurlyeq \gamma \preccurlyeq \Phi_{s-1}(A_{s-1})$.

We use the following priority ordering on the P- and N-requirements and the Q-subrequirements:

$$\{P_0, Q_{0,P_0}\} < \{N_0, Q_{0,N_0}\} < \{P_1, Q_{0,P_1}, Q_{1,P_1}\} < \{N_1, Q_{0,N_1}, Q_{1,N_1}\} \\ < \{P_2, Q_{0,P_2}, Q_{1,P_2}, Q_{2,P_2}\} < \cdots$$

where the requirements and subrequirements in the same set are considered to be of equally strong priority.

Construction

Stage 0: we set $A_0 = 0^\omega$. We let $\Phi_0 = \varnothing$. For all $e \in \omega$ and all $\mu \in 2^{<\omega}$, we let $g_{e,0}(\mu) = \mu$ and $p_{e,0}(\mu) = h(\mu)$. We initialise all requirements and subrequirements and proceed to the next stage.

Stage s, $s \geqslant 1$:

$s \equiv 1$ mod 3

Step 1

Let R be the P- or N-requirement of strongest priority which requires attention at stage s. We act according to the cases below, and then initialise all requirements and subrequirements of weaker priority than R. We say that we act for R at stage s. If we act in subcase 1b or subcase 2a, then we proceed to step 2, and otherwise, we proceed to the next stage.

Case 1: $R = P_i$ for some i. We act according to the subcases below.

Subcase 1a: P_i requires attention at stage s via (1). Let $\tau \prec A_{s-1}$ be of a fresh large length, and let $S_{i,s}$ consist of $i+2$ many pairwise incomparable strings which extend τ. We choose some string $\sigma \in S_{i,s}$ and let $A_s = \sigma\,\hat{}\,0^\omega$.

Subcase 1b: P_i requires attention at stage s via (2). We say that P_i is realised at stage s.

Subcase 1c: P_i requires attention at stage s via (3). Let σ be as in the definition of requires attention. We set $A_s = f_{i,s}(\sigma)\,\hat{}\,0^\omega$.

For all $e \leqslant i$, if we are not waiting for initial convergence for Q_{e,P_i} at stage s, then we let $\gamma_{e,P_i,s}$ be undefined, and we say that we are waiting for futher convergence for Q_{e,P_i} above the string $\Phi_{s-1}(f_{i,s}(\sigma))$ at stage $s+1$.

Case 2: $R = N_i$ for some i. We act according to the subcases below.

Subcase 2a: N_i requires attention at stage s via (1). We let $A_s = A_{s-1}$. Define $\tau_{i,s}$ to be some inital segment of A_s of a fresh large length. Let $\beta = \Phi_{s-1}(A_s)$. We enumerate $\langle \tau_{i,s}, \beta\rangle$ and $\langle \tau_{i,s}\,\hat{}\,0, \beta\,\hat{}\,0\rangle$ into Φ.

Subcase 2b: N_i requires attention at stage s via (2). Define ρ to be some string extending $\tau_{i,s-1}$ of a fresh large length, and such that there is no string $\alpha \in \operatorname{dom}\Phi_{s-1}$ such that $\tau_{i,s-1} \prec \alpha \prec \rho$. We set $A_s = \rho\,\hat{}\,0^\omega$. Let $\beta = \Phi_{s-1}(\tau_{i,s-1})$. We enumerate $\langle \rho, \beta\,\hat{}\,1\rangle$ into Φ.

For all $e \leqslant i$, we do the following. We do not say that we are waiting for initial convergence for Q_{e,N_i} at stage $s+1$. If we are not waiting for initial convergence for Q_{e,N_i} at stage s, then we say that we are waiting for further convergence for Q_{e,N_i} above the string $\beta\,\hat{}\,1$ at stage $s+1$.

<u>*Step 2*</u>

Let R be the requirement for which we acted in step 1, and let i be such that $R = P_i$ or $R = N_i$. For all $e \leqslant i$, we do the following.

If we are not waiting for initial or further convergence for any Q_e-subrequirement of stronger priority than $Q_{e,R}$ at stage s, then we do the following. Let I_s be the set of all j such that $P_j \leqslant Q_{e,R}$ and P_j has been realised by the beginning of stage s. Let N_s be the set of all j such that $N_j \leqslant Q_{e,R}$ and such that we have not acted for N_j in subcase 2b since it was last initialised. We define $c_{e,R,s}$ to be a fresh large number such that for all strings ν with $|\nu| \geqslant c_{e,R,s}$,

$$h(\nu) > \left(\bigoplus_{j \in I_s} \bigoplus_{\sigma \in S_{j,s}} o_{j,s}(\sigma) \right) \oplus |I_s| \oplus |N_s|$$

We say that we are waiting for initial convergence for $Q_{e,R}$ at stage $s+1$. We proceed to the next stage.

<u>*$s \equiv 2$ mod 3*</u>

If there is some Q-subrequirement which requires attention via (1) at stage s, then we follow the instructions below, and otherwise, we proceed to the next stage.

Let R be the strongest P- or N-requirement for which there is some $e \in \omega$ such that $Q_{e,R}$ requires attention via (1) at stage s, and choose the least e for this R. Let ν be as in the definition of requires attention. We do not say that we are waiting for initial convergence for $Q_{e,R}$ at stage $s+1$. We define $\nu_{e,R,s}$ to be ν.

If R is a P-requirement, then we do the following. Let i be such that $R = P_i$. Suppose that we last acted for P_i in subcase 1a or subcase 2a at stage q. We choose some $\sigma \in S_{i,s-1}$ such that $A_t \not\succ \sigma$ for all $t \in [q,s)$, and set $A_s = \sigma\,\hat{}\,0^\omega$. Let $\beta = \Phi_{s-1}(A_{s-1})$. We enumerate $\langle \sigma, \beta \rangle$ into Φ.

If R is an N-requirement, then we do the following. Let i be such that $R = N_i$. We let $A_s = A_{s-1}$. Define $\tau_{i,s}$ to be some inital segment of A_s of a fresh large length. Let $\beta = \Phi_{s-1}(A_s)$. We enumerate $\langle \tau_{i,s}, \beta \rangle$ and $\langle \tau_{i,s}\,\hat{}\,0, \beta\,\hat{}\,0 \rangle$ into Φ. We let $l_{i,s} = |\beta| + 1$. We say that we are waiting for further convergence for $Q_{e,R}$ above the string $\beta\,\hat{}\,0$ at stage $s+1$.

We say that we act for $Q_{e,R}$ at stage s. We initialise all requirements and subrequirements of weaker priority than $Q_{e,R}$, and proceed to the next stage.

<u>*$s \equiv 0$ mod 3*</u>

If there is some Q-subrequirement which requires attention via (3) at stage s, then we follow the instructions below, and otherwise, we proceed to the next stage.

Let R be the strongest P- or N-requirement for which there is some $e \in \omega$ such that $Q_{e,R}$ requires attention via (2) at stage s, and choose the least e for this R. If there is some N-requirement N_i with $N_i \leqslant Q_{e,R}$ such that $\tau_{i,s-1}$ is undefined then we proceed to the next stage. Otherwise, we follow the instructions below.

We initialise all requirements and subrequirements of weaker priority than $Q_{e,R}$. Let γ be as in the definition of requires attention. We define $\gamma_{e,R,s}$ to be γ. For every string μ such that $\mu \prec \Gamma_{e,s}(\gamma)$ and $g_{e,s-1}(\mu)$ is comparable with $\Gamma_{e,s}(\gamma)$, we do the following.

We define $g_{e,s}(\mu)$ to be some proper extension of μ which is not an initial segment of $\Gamma_{e,s}(\gamma)$. Let $I_{e,\mu,s}$ be the set of all j such that $\nu_{e,P_j,s}$ is defined and $\Gamma_{e,s}(\nu_{e,P_j,s}) \preccurlyeq \mu$. Let $N_{e,\mu,s}$ be the set of all j such that $\nu_{e,N_j,s}$ is defined, $\Gamma_{e,s}(\nu_{e,N_j,s}) \preccurlyeq \mu$, and we have not acted for N_j in subcase 2b since it was last initialised. We set

$$p_{e,s}(\mu) = \left(\bigoplus_{j \in I_{e,\mu,s}} \bigoplus_{\sigma \in S_{j,s}} o_{j,s}(\sigma) \right) \oplus |I_{e,\mu,s}| \oplus |N_{e,\mu,s}|.$$

We do not say that we are waiting for further convergence for $Q_{e,R}$ at stage $s+1$. We say that we act for $Q_{e,R}$ at stage s. We proceed to the next stage.

End of Construction

Lemma 6.2. *The construction can be carried out as described.*

Proof. The only difficulty is showing that if we wish to act for Q_{e,P_i} at some stage s with $s \equiv 2$ mod 3, then we can choose a string σ as described. Suppose that we act for P_i in subcase 1a at some stage q. We may assume that we do not initialise P_i at any stage after stage q. Then for all $s > q$, it is straightforward to see that the number of strings σ in $S_{i,s-1}$ for which there is some stage $t \in (q,s)$ such that $A_t \succ \sigma$ is equal to the number of $d \leqslant i$ such that we have acted for Q_{d,P_i} at some stage $u \in (q,s)$ with $u \equiv 2$ mod 3. Since $S_{i,q}$ consists of $i+2$ many pairwise incomparable strings, we can choose a string σ as described in stage s of the construction. □

Lemma 6.3. *Suppose that $S_{i,s} \neq \varnothing$. Then there is some $\sigma \in S_{i,s}$ such that $\sigma \prec A_s$.*

Proof. This follows from Lemma 6.2. □

Lemma 6.4. *The functional Φ is consistent.*

Proof. By induction on the stage number. □

Lemma 6.5. *Suppose that P_i is initialised for the final time at stage s^*. Then we may act for P_i at at most finitely many stages after stage s^*.*

Proof. We act for P_i in subcase 1a at stage $s^* + 1$, and by assumption, we may not act for P_i in subcase 1a at any later stage. We may act at most once for P_i in subcase 1b at some stage after stage s^*. By assumption, we have that $S_{i,t} = S_{i,s^*+1}$ for all $t > s^*$. Now since S_{i,s^*+1} is finite, and $\langle f_{i,s}, o_{i,s}\rangle_{s<\omega}$ is an α-change test, we can act at most finitely many times for P_i in subcase 1c after stage s^*. □

Lemma 6.6. *Suppose that N_i is initialised for the final time at stage s^*. Then we may act for N_i at at most finitely many stages after stage s^*.*

Proof. This is immediate. □

Lemma 6.7. *Suppose that $Q_{e,R}$ is initialised for the final time at stage s^*. Then we may act for $Q_{e,R}$ at at most finitely many stages after stage s^*.*

Proof. R is either a P-requirement or an N-requirement. First suppose that R is a P-requirement, and let i be such that $R = P_i$. We may act for Q_{e,P_i} at some stage s with $s \equiv 2 \bmod 3$ at most once after stage s^*. Suppose that we act for Q_{e,P_i} at some stage s with $s \equiv 0 \bmod 3$ after stage s^*. We do not say that we are waiting for further convergence for Q_{e,P_i} at the beginning of stage $s + 1$. Now suppose that we act for Q_{e,P_i} at some later stage t with $t \equiv 0 \bmod 3$. By the definition of requires attention, we must be waiting for further convergence for Q_{e,P_i} at stage t. Therefore, we must have acted for P_i in subcase 1c at some stage u with $u \in (s, t)$. As in the proof of Lemma 6.5, we may act for P_i in subcase 1c at at most finitely many stages after stage s. This establishes the lemma for this case.

Now suppose that R is an N-requirement, and let i be such that $R = N_i$. We may act for Q_{e,N_i} at some stage s with $s \equiv 2 \bmod 3$ at most once after stage s^*. Suppose that we act for Q_{e,N_i} at some stage s with $s \equiv 0 \bmod 3$ after stage s^*. If we act for Q_{e,N_i} at some later stage t with $t \equiv 0 \bmod 3$, then we must act for N_i in subcase 2b at some stage before stage t, and we will not be able to act again for Q_{e,N_i}. This establishes the lemma.

□

Let $A = \liminf_{s\to\infty} A_s$.

Lemma 6.8. *$\Phi(A)$ is total and A is Δ^0_2.*

Proof. That $\Phi(A)$ follows from the fact that we act for infinitely many N_i requirements. By Lemma 6.5, Lemma 6.6, Lemma 6.7, and the initialisation performed during the construction, A is Δ^0_2. □

Lemma 6.9. *Each P- and N-requirement is met.*

Proof. Fix some P-requirement P_i. By Lemma 6.5, Lemma 6.6, and Lemma 6.7, suppose that we last initialise P_i at stage s^*. If f_i is not a total α-change test, then P_i is met. So assume that f_i is a total α-change test. Again by Lemma 6.5, there is a last stage t at which we act for P_i, and by assumption and Lemma 6.3, we must act for P_i in subcase 1c at stage t. Let σ be as in the definition of requires attention at stage t. Then we have that $f_{i,u}(\sigma) = f_{i,t}(\sigma)$ for all $u \geqslant t$, and $f_{i,t}(\sigma) \prec A_u$ for all $u \geqslant t$, which shows that P_i is met.

Now fix some N-requirement N_i, and suppose that we last initialise N_i at stage s^*. There are at most finitely many stages $t > s^*$ such that $\tau_{i,t} \neq \tau_{i,t-1}$. Given this, it is clear that the instructions in case 2 of the construction ensure that N_i is met. $\square$

For all $e \in \omega$, $\mu \in 2^{<\omega}$, and $s \in \omega$, define $I_{e,\mu,s}$ and $N_{e,\mu,s}$ as in the construction.

Lemma 6.10. *Let $e \in \omega$ and $\mu \in 2^{<\omega}$. Then for all $s \in \omega$,*

$$h(\mu) > \left(\bigoplus_{j \in I_{e,\mu,s}} \bigoplus_{\sigma \in S_{j,s}} o_{j,s}(\sigma) \right) \oplus |I_{e,\mu,s}| \oplus |N_{e,\mu,s}|.$$

Proof. We show this by induction on s. We have that $I_{e,\mu,0} = \varnothing$ and $N_{e,\mu,0} = \varnothing$, and so since $h(\mu) > 0$, the statement holds at stage 0. Suppose by induction that $s > 0$ and that the statement holds for all t with $t < s$. If we do not define some string $\nu_{e,R,s}$ at stage s, then $I_{e,\mu,s} = I_{e,\mu,s-1}$ and $N_{e,\mu,s} = N_{e,\mu,s-1}$, and the statement holds at stage s. So suppose that we define $\nu_{e,R,s}$ at stage s for some Q_e-subrequirement $Q_{e,R}$.

There are two cases to consider. First suppose that $\Gamma_{e,s}(\nu_{e,R,s}) \not\preccurlyeq \mu$. Then $I_{e,\mu,s} = I_{e,\mu,s-1}$ and $N_{e,\mu,s} = N_{e,\mu,s-1}$, and the statement holds at stage s.

So suppose that $\Gamma_{e,s}(\nu_{e,R,s}) \preccurlyeq \mu$. Since we define $\nu_{e,R,s}$ at stage s, we are waiting for further convergence for $Q_{e,R}$ at stage s. Suppose that stage t is the last stage before stage s at which we began waiting for further convergence for $Q_{e,R}$. Then we may not initialise $Q_{e,R}$ at any stage after stage t and before stage s. Let I_t, N_t, and $c_{e,R,t}$ be as at stage t of the construction. Then since h is an α-order function, we have that

$$h(\mu) > \left(\bigoplus_{j \in I_t} \bigoplus_{\sigma \in S_{j,t}} o_{j,t}(\sigma) \right) \oplus |I_t| \oplus |N_t|.$$

We first claim that $I_{e,\mu,s} \subseteq I_t$. Let $j \in I_{e,\mu,s}$. Then $\nu_{e,P_j,s}$ is defined and $\Gamma_{e,s}(\nu_{e,P_j,s}) \preccurlyeq \mu$. To show that $j \in I_t$, we must show that $P_j \leqslant Q_{e,R}$ and that P_j has been realised by stage t. To show that $P_j \leqslant Q_{e,R}$, suppose for contradiction that $Q_{e,R} < P_j$. Then $Q_{e,R} < Q_{e,P_j}$, and since we initialise all subrequirements of weaker priority than $Q_{e,R}$ at stage s, we let $\nu_{e,P_j,s}$ be undefined, which is a contradiction. So $P_j \leqslant Q_{e,R}$. We now show that P_j has been realised by stage t. Since $\nu_{e,P_j,s}$ is defined, P_j must have been realised by the beginning of stage s. Suppose for contradiction that P_j was realised at some stage after stage t. If $P_j < Q_{e,R}$, then realising P_j will initialise $Q_{e,R}$ at some stage after stage t and before stage s, which is a contradiction. If P_j and $Q_{e,R}$ are of equally strong priority, then $P_j = R$. However, since we began waiting for futher convergence for $Q_{e,R}$ at stage t, then P_j had been realised by stage t, which is a contradiction. This establishes the claim.

We now claim that $N_{e,\mu,s} \subseteq N_t$. Let $j \in N_{e,\mu,s}$. Then $\nu_{e,N_j,s}$ is defined, $\Gamma_{e,s}(\nu_{e,N_j,s}) \preccurlyeq \mu$, and we have not acted for N_j in subcase 2b since it was last initialised. By the definition of N_t, it suffices to show that $N_j \leqslant Q_{e,R}$. As above, suppose for contradiction that $Q_{e,R} < N_j$. Then $Q_{e,R} < Q_{e,N_j}$, and since we initialise all subrequirements of weaker priority than $Q_{e,R}$ at stage s, we let $\nu_{e,N_j,s}$ be undefined, which is a contradiction.

Since for all $j \in \omega$, $\langle f_{j,s}, o_{j,s} \rangle_{s<\omega}$ is an α-change test, we have that for all $j \in I_{e,\mu,s}$ and all $\sigma \in S_{j,s}$, $o_{j,s}(\sigma) \leqslant o_{j,t}(\sigma)$. This, together with the above claims, suffices to show that the statement holds at stage s. $\square$

The next two lemmas are immediate.

Lemma 6.11. *Suppose that we act for some R-requirement in either subcase 1a or subcase 1b at stage s, or that we act for some Q-subrequirement at stage s. Then $\Phi_s(A_s) = \Phi_{s-1}(A_{s-1})$.*

Lemma 6.12. *If $\nu_{e,M,s}$ and $\nu_{e,R,s}$ are defined and $M < R$, then $\Gamma_{e,s}(\nu_{e,M,s}) \prec \Gamma_{e,s}(\nu_{e,R,s})$.*

The next two lemmas are shown by a straightforward induction on the stage number.

Lemma 6.13. *Suppose that we act for $Q_{e,R}$ at some stage s with $s \equiv 0$ mod 3. Then for all M with $Q_{e,M} < Q_{e,R}$,*

(1) if M is a P-requirement and P has been realised by the beginning of stage s, then $\nu_{e,M,s}$ and $\gamma_{e,M,s}$ are defined, and we are not waiting for initial or further convergence for $Q_{e,M}$ at stage s, and
(2) if M is an N-requirement and we have not acted for M in subcase 2b since it was last initialised, then $\nu_{e,M,s}$ and $\gamma_{e,M,s}$ are defined, and we are not waiting for initial or further convergence for $Q_{e,M}$ at stage s.

Lemma 6.14. *Suppose that we are not waiting for initial or further convergence $Q_{e,R}$ at stage s. Then if R is a P-requirement and R has been realised by stage s, or R is an N-requirement and we have not acted for R in subcase 2b since it was last initialised, then $\gamma_{e,R,s}$ is defined, and for all strings $\mu \prec \Gamma_{e,s}(\gamma_{e,R,s})$, $g_{e,s}(\mu)$ is not comparable with $\Gamma_{e,s}(\gamma_{e,R,s})$.*

Lemma 6.15. *Suppose that we act for $Q_{e,R}$ at stage s and update the arrow for μ. Suppose that we update the arrow for μ at some later stage u. Then there is some stage t with $t \in (s, u)$ at which we either*

- *act for some P_j with $j \in I_{e,\mu,s}$ in subcase 1c, or*
- *act for some N_j with $j \in N_{e,\mu,s}$ in subcase 2b.*

Proof. Let γ be as at stage s of the construction. Since we update the arrow for μ at stage s, we have that $\mu \prec \Gamma_{e,s}(\gamma)$. We also have that $\Gamma_{e,s}(\nu_{e,R,s}) \preccurlyeq \Gamma_{e,s}(\gamma)$. So μ and $\Gamma_{e,s}(\nu_{e,R,s})$ are comparable. There are two cases to consider.

The first case is that $\Gamma_{e,s}(\nu_{e,R,s}) \preccurlyeq \mu$. We first show that we must act for some P- or N-requirement M with $M \leqslant Q_{e,R}$ at some stage t with $t \in (s, u)$. Suppose for contradiction that we do not act for any P- or N-requirement M with $M \leqslant Q_{e,R}$ at any stage t with $t \in (s, u)$. We have that $\Phi_s(A_s) \succcurlyeq \gamma$. At stage s, we initialise all P- and N-requirements of weaker priority than $Q_{e,R}$. Then by the choice of followers for P- and N-requirements, and the axioms for Φ that we define at stages at which we act for P- and N-requirements, we have that $\Phi_t(A_t) \succcurlyeq \gamma$ for all t with $t \in (s, u)$. This contradicts the fact that we update the arrow for μ at stage u.

Suppose that we do not act for any N-requirement N_j with $N_j \leqslant Q_{e,R}$ at any stage t with $t \in (s, u)$. By the result of the previous paragraph, and Lemma 6.11, we must act for some P-requirement P_j in subcase 1c with $P_j \leqslant Q_{e,R}$ at some stage t with $t \in (s, u)$. We show that $j \in I_{e,\mu,s}$. First suppose that P_j and $Q_{e,R}$ are of equally strong priority. Then $R = P_j$, and so $\nu_{e,P_j,s}$ is defined and $\Gamma_{e,s}(\nu_{e,Pj,s}) \preccurlyeq \mu$. Therefore, $j \in I_{e,\mu,s}$. Now suppose that $P_j < Q_{e,R}$. We have that $Q_{e,P_j} < Q_{e,R}$. Then by Lemma 6.13 and Lemma 6.12, $\nu_{e,P_j,s}$ is defined and $\Gamma_{e,s}(\nu_{e,P_j,s}) \prec \Gamma_{e,s}(\nu_{e,R,s})$. Now since $\Gamma_{e,s}(\nu_{e,R,s}) \preccurlyeq \mu$, $\Gamma_{e,s}(\nu_{e,P_j,s}) \prec \mu$, and $j \in I_{e,\mu,s}$.

Now suppose that we do act for some N-requirement N_j with $N_j \leqslant Q_{e,R}$ at some stage t with $t \in (s, u)$. We may assume that t is the least such. We show that

$j \in N_{e,\mu,s}$. If we do act for some P-requirement P_j with $P_j \leqslant Q_{e,R}$ at some stage after stage s and before stage t, then, as above, we must act for P_j in subcase 1c and $j \in I_{e,\mu,s}$. So suppose that we do not. By the instructions we follow at stage s, $\tau_{j,s-1}$ is defined. Now by assumption, we cannot act for N_j in subcase 2a at stage t, and so we must act for N_j in subcase 2b at stage t. Therefore, we have not acted for N_j in subcase 2b since it was last initialised. First suppose that N_j and $Q_{e,R}$ are of equally strong priority. Then $R = N_j$, and so $\nu_{e,N_j,s}$ is defined, and $\Gamma_{e,s}(\nu_{e,P_j,s}) \preccurlyeq \mu$. Therefore, $j \in N_{e,\mu,s}$. Now suppose that $N < Q_{e,R}$. We have that $Q_{e,N_j} < Q_{e,R}$. Then by Lemma 6.13 and Lemma 6.12, $j \in N_{e,\mu,s}$.

The second case is that $\mu \prec \Gamma_{e,s}(\nu_{e,R,s})$. We show that we must act for some P- or N-requirement M with $M < R$ at some stage t with $t \in (s, u)$. Suppose for contradiction that we do not. Then we will have that $\Gamma_{e,s}(\nu_{e,R,s}) \preccurlyeq \Phi_t(A_t)$ for all $t \geqslant s$, which contradicts the fact that we update the arrow for μ at stage u.

Let S be the set of P- and N-requirements of stronger priority than $Q_{e,R}$, excluding those N-requirements for which we have acted in subcase 2b by stage s. Using the result of the previous paragraph, it can be shown that S is nonempty. Let M be the requirement in S of weakest priority.

We claim that $\gamma_{e,M,s}$ is defined and $\Gamma_{e,s}(\gamma_{e,M,s}) \preccurlyeq \mu$. The fact that $\gamma_{e,M,s}$ is defined follows from the choice of M and Lemma 6.13. To show that $\Gamma_{e,s}(\gamma_{e,M,s}) \preccurlyeq \mu$, suppose for contradiction that $\Gamma_{e,s}(\gamma_{e,M,s}) \not\preccurlyeq \mu$. Since $M < R$, we have that $\Gamma_{e,s}(\gamma_{e,M,s}) \prec \Gamma_{e,s}(\nu_{e,R,s})$. So both $\Gamma_{e,s}(\gamma_{e,M,s})$ and μ are initial segments of $\Gamma_{e,s}(\nu_{e,R,s})$, and are therefore comparable. So $\mu \prec \Gamma_{e,s}(\gamma_{e,M,s})$. Since we act for $Q_{e,R}$ and update the arrow for μ at stage s, $g_{e,s}(\mu)$ must be comparable with $\Gamma_{e,s}(\gamma_{e,R,s})$. We also have that $\Gamma_{e,s}(\gamma_{e,M,s})$ and $\Gamma_{e,s}(\gamma_{e,R,s})$ are comparable. So $\mu \prec \Gamma_{e,s}(\gamma_{e,M,s})$ and $g_{e,s}(\mu)$ is comparable with $\Gamma_{e,s}(\gamma_{e,M,s})$, which contradicts Lemma 6.14. This establishes the claim.

Suppose that we do not act for any N-requirement N_j with $N_j < Q_{e,R}$ at any stage t with $t \in (s, u)$. Then we must act for some P-requirement P_j in subcase 1c with $P_j \leqslant Q_{e,M}$ at some stage t with $t \in (s, u)$. The remainder of the argument is as in the first case above, with M now replacing R.

□

Lemma 6.16. *For all $e \in \omega$, t_e is an h-change test.*

Proof. Fix some $\mu \in 2^{<\omega}$. Recall that at stage 0, we set $p_{e,0}(\mu) = h(\mu)$. Therefore, it suffices to show that for all $s > 0$, if $g_{e,s}(\mu) \neq g_{e,s-1}(\mu)$ then $p_{e,s}(\mu) < p_{e,s-1}(\mu)$. If we never update the arrow for μ, then we are done. So suppose that we do update the arrow for μ at some stage, and let $S = \{s_0 < s_1 < \ldots\}$ be the nonempty, and possibly infinite set of all stages at which we update the arrow for μ. We must show that for all k such that s_k is defined, $p_{e,s_k}(\mu) < p_{e,s_k-1}(\mu)$.

We first claim that $p_{e,s_0}(\mu) < p_{e,s_0-1}(\mu)$. By the definition of s_0, we have that $p_{e,s_0-1}(\mu) = h(\mu)$. By Lemma 6.10, the claim follows.

Now suppose by induction that s_k is defined, and that for all $j \leqslant k$, $p_{e,s_j}(\mu) < p_{e,s_j-1}(\mu)$. If s_{k+1} is not defined, then we are done. So we assume that s_{k+1} is defined. We must show that $p_{e,s_{k+1}}(\mu) < p_{e,s_k-1}(\mu)$. In fact, it suffices to show that $p_{e,s_{k+1}}(\mu) < p_{e,s_k}(\mu)$.

Note that if we define some string $\nu_{e,R,s}$ at stage s, then if we define some string $\nu_{d,M,t}$ at any later stage t, the length of $\nu_{d,M,t}$ is fresh and large at stage t. Therefore, by the definition of $I_{e,\mu,s}$ and $N_{e,\mu,s}$, if $s < t$, then $I_{e,\mu,t} \subseteq I_{e,\mu,s}$ and $N_{e,\mu,t} \subseteq N_{e,\mu,s}$.

Since we update the arrow for μ at stage s_{k+1}, by Lemma 6.15, we must either act for some P_j with $j \in I_{e,\mu,s_k}$ in subcase 1c, or act for some N-requirement N_j with $j \in N_{e,\mu,s}$ in subcase 2b at some stage t with $t \in (s_k, s_{k+1})$.

First suppose that we do not act for any N-requirement N_j with $j \in N_{e,\mu,s_k}$ in subcase 2b at any stage t with $t \in (s_k, s_{k+1})$. Then we must act for some P-requirement P_j with $j \in I_{e,\mu,s}$ at some stage t with $t \in (s_k, s_{k+1})$. There are two cases to consider. The first case is that $I_{e,\mu,s_{k+1}}$ is a proper subset of I_{e,μ,s_k}. Then it is clear from the definition of $p_{e,s_{k+1}}(\mu)$ that $p_{e,s_{k+1}}(\mu) < p_{e,s_k}(\mu)$. The second case is that $I_{e,\mu,s_{k+1}} = I_{e,\mu,s_k}$. Since we act for P_j in subcase 1c at stage t, there is some $\sigma \in S_{P_j,s_k}$ for which $o_{j,s_{k+1}}(\sigma) < o_{j,s_k}(\sigma)$. This suffices to show that $p_{e,s_{k+1}}(\mu) < p_{e,s_k}(\mu)$.

Now suppose that we do act for some N-requirement N_j with $j \in N_{e,\mu,s_k}$ in subcase 2b at some stage t with $t \in (s_k, s_{k+1})$. Then $N_{e,\mu,s_{k+1}}$ is a proper subset of N_{e,μ,s_k} and $p_{e,s_{k+1}}(\mu) < p_{e,s_k}(\mu)$. □

Lemma 6.17. *Every Q-requirement is met.*

Proof. Let $e \in \omega$ be such that $\Gamma_e(\Phi(A))$ is total. By Lemma 6.16, t_e is an h-change test. We therefore must show that $\Gamma_e(\Phi(A))$ does not meet the range of t_e.

Note that there are infinitely many $i \in \omega$ such that f_i is a total α-change test. Therefore, since $\Gamma_e(\Phi(A))$ is total, there is an infinite set S of P- and N-requirements such that for all $R \in S$, at all but finitely many stages, we are not waiting for initial or further convergence for $Q_{e,R}$. Then for all $R \in S$, $\gamma_{e,R} = \lim_s \gamma_{e,R,s}$ exists, and $\{\gamma_{e,R} : R \in S\}$ is infinite and cofinal along $\Gamma_e(\Phi(A))$. So let $\mu \prec \Gamma_e(\Phi(A))$. Then by Lemma 6.14, $g_e(\mu)$ is not an initial segment of $\Gamma_e(\Phi(A))$. □

□

Although downward density of weakly α-change generics below Δ^0_2 weakly α-change generic degrees fails, we do have the following.

Theorem 6.18. *Let $\alpha \leqslant \varepsilon_0$ be a power of ω and let $\mathbf{a} \leqslant_{\mathrm{T}} \mathbf{0}'$ be a weakly α-change generic degree. Then there is a weakly α-change generic degree $\mathbf{b} <_{\mathrm{T}} \mathbf{a}$.*

Proof. Let $A \in \mathbf{a}$ be a set which meets the range of every total α-change test, and let $\langle A_s \rangle_{s<\omega}$ be a computable approximation to A. We construct a Turing functional Φ and meet for every $i \in \omega$ the requirement

$$R_i : \text{if } f_i \text{ is total, then } \Phi(A) \text{ meets the range of } f_i$$

where $\langle f_i \rangle_{i<\omega} = \langle \langle f_{i,s}, o_{i,s} \rangle_{s<\omega} \rangle_{i<\omega}$ is an effective list of all α-change tests, and for every $e \in \omega$ the requirement

$$Q_e : \Psi_e(\Phi(A)) \neq A$$

where $\langle \Psi_e \rangle_{e<\omega}$ is an effective list of all Turing functionals.

For every requirement R_i, we will build an α-change test $t_i = \langle g_{i,s}, p_{i,s} \rangle_{s<\omega}$. In the case that f_i is total, then we will ensure that t_i is total as well. We define t_i in such a way that if $A \succ \lim_s g_{i,s}(\tau)$ for some τ, then $\Phi(A) \succ \lim_s f_{i,s}(\Phi(\tau))$. In order to implement this strategy while keeping the functional Φ consistent, we will ensure that the arrows in the test t_i, and the strings in the domain of Φ, are chosen *sparsely*. Essentially, we need to make sure that for any string τ, we can choose the arrow for τ to be some string which does not extend any string ρ with $\rho \in \operatorname{dom} \Phi_s$

and $\rho > \tau$. This can be easily managed by simply requiring the strings enumerated into the domain of Φ be increasing in length.

If we say at stage s that we *reset* the test t_i, then we abandon the test t_i, and will at later stages build a new test, which, abusing notation, we will also call t_i. If a test t_i is reset at stage s, then we set $g_{i,s'}(\sigma) = \sigma$ and $p_{i,s'}(\sigma) = \infty$ for all $s' \leqslant s$ and all $\sigma \in 2^{<\omega}$. Unless otherwise specified, we will let $g_{i,s+1}(\sigma) = g_{i,s}(\sigma)$ and $p_{i,s+1}(\sigma) = p_{i,s}(\sigma)$.

The strategy for Q_e will look for strings $\alpha \prec A_s$ in the domain of Φ_s such that $\Psi_{e,s}(\Phi_{s-1}(\alpha)) \prec A_s$. If we see such a string α, then we will want A to move to some string which is incomparable with $\Psi_e(\Phi(\alpha))$, while keeping $\Phi(\alpha) \prec \Phi(A)$. To bring about such a change in A, we will challenge its genericity. For every requirement Q_e, we will define a c.e. set of strings C_e. The set C_e will consist of strings that we would like A to extend for the sake of this strategy. We use the fact that A is Δ^0_2 and 1-generic to show that A must indeed meet C_e.

Similar to the tests above, if we say at stage s that we *empty* the set C_e, then we abandon the set C_e, and will at later stages define a new set of strings, which we will also call C_e. If a set C_e is emptied at stage s, then we set $C_{e,s} = \varnothing$. Unless otherwise specified, we will let $C_{e,s+1} = C_{e,s}$.

We fix some computable bijection $\langle \cdot, \cdot \rangle : \omega \times \omega \to \omega$. We order the R- and Q-requirements as follows:

$$R_0 < Q_0 < R_1 < Q_1 < \cdots .$$

Construction

Stage 0: we set $\Phi(\lambda) = \lambda$. For all $i \in \omega$ and $\tau \in 2^{<\omega}$, we set $g_{i,0}(\tau) = \tau$ and $p_{i,0}(\tau) = \infty$. For all $e \in \omega$, we set $C_{e,0} = \varnothing$.

Stage s, $s \geqslant 1$:

$\underline{s = 4n + 1 \textit{ for some } n \in \omega}$:

Let i and m be such that $n = \langle i, m \rangle$. If A_s does not meet range $f_{i,s}$, then we do the following. Suppose that d is greatest such that $p_{i,s-1}(\tau) < \infty$ for all strings τ of length strictly less than d. Let

$$l = \max \{ |\Phi_{s-1}(\tau)| : |\tau| = d \}.$$

If $o_{i,s}(\sigma) < \infty$ for all strings σ of length at most l, then for all strings τ of length d, we let $p_{i,s}(\tau) = o_{i,s}(\Phi_{s-1}(\tau))$, and choose some string ν of a fresh large length which extends τ but does not extend any string in dom Φ_{s-1} which properly extends τ, and set $g_{i,s}(\tau) = \nu$ and $\Phi_s(\nu) = f_{i,s}(\Phi_{s-1}(\tau))$. We say that we act for R_i at stage s. We reset all tests t_j for $j > i$, and empty all sets C_f for $f > i$. We proceed to the next stage.

$\underline{s = 4n + 2 \textit{ for some } n \in \omega}$:

Let e and m be such that $n = \langle e, m \rangle$. If A_s does not extend some string in $C_{e,s-1}$, then we do the following. Suppose d is the length of the longest string in $C_{e,s-1}$. Let

$$l = \max \{ y : (\forall x \leqslant y)\, A_s(x) = \Psi_e(\Phi_{s-1}(A_s))(x) \}.$$

If $l > d$, then let $\alpha \prec A_s$ be some string in the domain of Φ_{s-1} which is of length greater than l, and such that $|\Psi_{e,s}(\Phi(\alpha))| = l$. Let σ be the sibling of $\Psi_e(\Phi(\alpha))$, and let γ be some string of a fresh large length which extends σ but does not extend any string in $\operatorname{dom}\Phi_{s-1}$ which properly extends σ. We enumerate γ into C_e, and we define $\Phi_s(\gamma) = \Phi(\alpha)$. We say that we act for Q_e at stage s. We reset all tests t_j for $j > e$, and empty all sets C_f for $f > e$. We proceed to the next stage.

$s = 4n+3$ *for some* $n \in \omega$:

Let i and m be such that $n = \langle i, m\rangle$. If A_s does not meet range $f_{i,s}$, then we do the following. For all strings τ such that $p_{i,s-1}(\tau) < \infty$, if $\Phi(g_{i,s-1}(\tau)) \neq f_{i,s}(\Phi(\tau))$, then we choose some string ν of a fresh large length which extends τ but does not extend any string currently in the domain of Φ which properly extends τ, and set $g_{i,s}(\tau) = \nu$ and $\Phi_s(\nu) = f_{i,s}(\Phi_{s-1}(\tau))$. If we do update the arrow for some such τ, we reset all tests t_j for $j > i$, and empty all sets C_f for $f > i$. We say that we act for R_i at stage s. We proceed to the next stage.

$s = 4n$ *for some* $n > 0$:

Let $\alpha \prec A_s$ be greatest such that $\alpha \in \operatorname{dom}\Phi_{s-1}$. We choose some string $\beta \prec A_s$ of a fresh large length which does not extend any string in $\operatorname{dom}\Phi_{s-1}$ which properly extends α, and set $\Phi_s(\beta) = \Phi(\alpha)\hat{\ }0$. We proceed to the next stage.

End of Construction

Lemma 6.19. *The construction can be successfully carried out.*

Proof. We show by induction on the stage number that at all stages s and for all strings $\tau \in 2^{<\omega}$, there is a string ν which extends τ but does not extend any string in $\operatorname{dom}\Phi_{s-1}$ that extends τ. Note that at any stage t, if we define $\Phi_t(\delta)$ for some string δ, then δ is chosen to be of a fresh large length. Therefore, the domain of Φ consists of strings which are of different lengths. Then we may take ν to be some string which extends τ, and such that no initial segment of ν of length greater than $|\tau|$ is already in $\operatorname{dom}\Phi_{s-1}$. □

Lemma 6.20. Φ *is consistent.*

Proof. We show this by induction on the stage number. We define $\Phi(\lambda) = \lambda$ at stage 0. Therefore, Φ_0 is consistent. Suppose by induction that Φ_{s-1} is consistent. We consider the different ways in which we can act at stage s, which depend on the value of s mod 4.

First suppose that $s \equiv 1 \bmod 4$ and that we define $\Phi_s(\nu)$ for some ν extending a string τ as in the construction. By the choice of the string ν, the only strings in $\operatorname{dom}\Phi_{s-1}$ which are comparable with ν are the initial segments of τ. Given our inductive assumption, we only need to show that $\Phi_s(\nu) \succcurlyeq \Phi(\tau)$. This follows from the fact that for all strings σ, $f_{i,s}(\sigma) \succcurlyeq \sigma$.

Now suppose that $s \equiv 2 \bmod 4$ and that we define $\Phi_s(\gamma)$ for some γ as in the construction. Let α and σ be as in the construction. By the choice of γ, the only strings in $\operatorname{dom}\Phi_{s-1}$ which are comparable with γ are the initial segments of σ. Let β be the greatest initial segment of σ which is in $\operatorname{dom}\Phi_{s-1}$. Given our inductive assumption, we only need to show that $\Phi_s(\gamma) \succcurlyeq \Phi(\beta)$. We set $\Phi_s(\gamma) = \Phi(\alpha)$, and by the choice of α and our inductive assumption, we have $\alpha \succ \beta$ and so $\Phi(\alpha) \succcurlyeq \Phi(\beta)$.

If $s \equiv 3 \bmod 4$, then Φ_s remains consistent by the same proof as when $s \equiv 1 \bmod 4$, and it is easy to see that Φ_s remains consistent if $s \equiv 0 \bmod 4$.

□

Lemma 6.21. *$\Phi(A)$ is total.*

Proof. There are infinitely many stages of the form $4n$ for some $n > 0$, and at each such stage, we make a definition for Φ. Therefore, $\operatorname{dom}\Phi$ is an infinite c.e. set of strings. We show that A cannot avoid $\operatorname{dom}\Phi$. Suppose for contradiction that there is some $\tau \prec A$ such that no extension of τ is in $\operatorname{dom}\Phi$. Let t be such that $A_s \succ \tau$ for all $t \geqslant s$. Then at some stage $u > t$ of the form $4n$ for some $n > 0$, we will enumerate some extension of τ into $\operatorname{dom}\Phi$. This is a contradiction. As A is weakly α-change generic, it is 1-generic, and so must meet $\operatorname{dom}\Phi$. In fact, A must meet $\operatorname{dom}\Phi$ infinitely many times, which is sufficient to show that $\Phi(A)$ is total. □

Lemma 6.22. *Every R- and Q-requirement is met.*

Proof. Let i be such that f_i is a total α-change test. Assume by induction that we do not act for any requirement of stronger priority than R_i after stage s. Then the test t_i is never reset after stage s. We assume for contradiction that $\Phi(A)$ does not meet range f_i. Then because f_i is a total test and by the previous lemma, we will act at infinitely many stages of the form $4\langle i, m\rangle + 1$ for some $m \in \omega$, and t_i will too be a total α-change test. As A is weakly α-change generic, A must meet the range of t_i. By the action taken at stages of the form $4\langle i, m\rangle + 3$ for some $m \in \omega$ and the definitions we make for the functional Φ, if $\tau \prec A$ is such that $A \succ \lim_s g_{i,s}(\tau)$, then $\lim_s f_{i,s}(\Phi(\tau)) \prec \Phi(A)$. Therefore, $\Phi(A)$ does meet range f_i, which is a contradiction.

Now assume by induction that we do not act for any requirement of stronger priority than Q_e after stage s. Then the set C_e is never reset after stage s. Assume for contradiction that $\Psi_e(\Phi(A)) = A$. Then we will act at infinitely many stages of the form $4\langle i, m\rangle + 2$ for some $m \in \omega$, and the set C_e will be infinite. We show that A cannot avoid the set of strings C_e. Suppose for contradiction that $\tau \prec A$ is such that there is no extension of τ in C_e. Let t be such that $A_u \succ \tau$ for all $u \geqslant t$. Note that C_e must contain strings which are arbitrarily long. Let $u > t$ be a stage where we enumerate some string γ of length strictly greater than $|\tau|$ into C_e. Then $\gamma \succ \tau$, which is a contradiction. It is clear by construction that if A meets C_e, then $\Psi_e(\Phi(A)) \neq A$. As A is weakly α-change generic, it is 1-generic, and so must meet C_e. This is a contradiction.

□

□

Using the approach from Proposition 2.10, we can modify the proof of the previous theorem to show the following.

Theorem 6.23. *Let $\alpha \leqslant \varepsilon_0$ be a power of ω and let $\mathbf{a} \leqslant_{\mathrm{T}} \mathbf{0}'$ be a uniformly α-change generic degree. Then there is a uniformly α-change generic degree $\mathbf{b} <_{\mathrm{T}} \mathbf{a}$.*

7. Further directions

In [21], we show the following.

Theorem 7.1. *Let $\alpha \leqslant \varepsilon_0$ be a power of ω. Then there is a Δ^0_2 Turing degree which is weakly α-change generic but not α-change generic.*

This theorem, together with Theorem 4.16 and Theorem 5.4 shows that the separation of each level in the hierarchy of multiple genericity notions can be witnessed by Δ^0_2 Turing degree.

The main open question left from the topics we have investigated in this paper is whether the α-generic degrees are downwards dense below Δ^0_2 α-change generic degrees. We conjecture that this fails in the strongest possible sense.

Conjecture 7.2. *For every $\alpha \leqslant \varepsilon_0$ which is a power of ω, there is an α-change generic degree $\mathbf{a} \leqslant_{\mathrm{T}} \varnothing'$ which bounds a noncomputable degree $\mathbf{b}$ which does not bound a uniformly ω-change generic degree.*

This would show that a conjecture of Schaeffer [24] holds. The first author's thesis [20] contains some discussion of the basic strategy for the previous conjecture.

For any notion of randomness or genericity, it is important to determine its associated lowness notion. We can define a real being low for any of the notions of multiple genericity by relativising the definition of an α-change test. In fact, there are two ways we could relativise an α-change test to an oracle A. We could fully relativise by allowing both sequences of functions $\langle f_s\rangle_{s<\omega}$ and $\langle o_s\rangle_{s<\omega}$ access to A, or we could instead partially relativise by allowing only $\langle f_s\rangle_{s<\omega}$ access. Liang Yu, in personal communication with the authors, has pointed out that Shore and Slaman's extension [25] of the Posner-Robinson theorem to all computable ordinals should show that for any $\alpha \leqslant \varepsilon_0$, the reals which are low for α-change genericity are exactly the computable reals.

Another basic question is whether van Lambalgen's theorem holds for these notions. We suspect that it would fail for the uniformly and weakly α-change generic degrees, but hold for the α-change generic degrees.

More generally, it would be interesting to see how these new notions refine the results of [1] and [2] regarding typical behaviour for generic degrees. Our investigation into downward density is an example of this.

Other possible directions include interaction with the computably bounded randoms of [3], and lattice embeddings below multiply generic degrees.

References

[1] George Barmpalias, Adam Day, and Andy Lewis-Pye. *The typical Turing degree*, Proceedings of the London Mathematical Society, 109(1), 1–39, 2014.

[2] George Barmpalias and Andy Lewis-Pye. *The information content of typical reals*, in *Turing's Ideas - Their Significance and Impact*, G. Sommaruga, T. Strahm (eds.), Birkhauser / Springer, Basel, 2014.

[3] Paul Brodhead, Rod Downey, and Keng Meng Ng. *Bounded randomness.* M.J. Dinneen et al. (Eds.), *Workshop on Theoretical Computer Science 2012 (Calude Festschrift)*, LNCS 7160, 59–70. Springer, Heidelberg, 2012

[4] Chi Tat Chong and Rod Downey. *Minimal degrees recursive in 1-generic degrees*, Annals of Pure and Applied Logic, 48, 215–225, 1990.

[5] Chi Tat Chong and Carl Jockusch. *Minimal degrees and 1-generic sets below* $0'$ in Börger E., Oberschelp W., Richter M.M., Schinzel B., Thomas W. (eds), *Computation and Proof Theory.* Lecture Notes in Mathematics, vol 1104. Springer, Berlin, Heidelberg, 1984.

[6] Rod Downey and Amy Gale. *On genericity and Ershov's hierarchy.* Mathematical Logic Quarterly, vol. 47, 161–182, 2001.

[7] Rod Downey and Noam Greenberg. Pseudo-jump inversion, upper cone avoidance, and strong jump-traceability. *Advances in Mathematics*, 237 (2013), 252–285.

[8] Rod Downey and Noam Greenberg. *A transfinite hierarchy of lowness notions in the computably enumerable degrees, unifying classes, and natural definability.* Annals of Mathematics Studies, 2020.

[9] Rod Downey and Denis Hirschfeldt. *Algorithmic Randomness and Complexity.* Theory and Applications of Computability. Springer, New York, 2010.

[10] Rod Downey, Denis Hirschfeldt, Andre Nies and Frank Stephan. Trivial reals. *Proceedings of the 7th and 8th Asian Logic Conferences*, (2003), 103–131.

[11] Rod Downey, Carl Jockusch, and Michael Stob. *Array nonrecursive sets and multiple permitting arguments*, in *Recursion Theory Week. Proceedings of the Conference Held at the Mathematisches Forschungsinstitut, Oberwolfach, March 19 25, 1989*, K. Ambos-Spies, G. H. Müller, and G. E. Sacks, editors, volume 1432 of Lecture Notes in Mathematics, pages 141–174. Springer, Berlin, 1990.

[12] Rod Downey, Carl Jockusch, and Michael Stob. *Array nonrecursive sets and genericity*, in *Computability, Enumerability, Unsolvability: Directions in Recursion Theory, Edited by S. B. Cooper, T. A. Slaman, and S. S. Wainer*, pages 93-105, 1996.

[13] Rod Downey and Satyadev Nandakumar. *A weakly-2-generic which bounds a minimal degree.* The Journal of Symbolic Logic, 84(4), 1326–1347, 2019.

[14] Christine Haught. *The degrees below a 1-generic degree* $< \mathbf{0}'$, The Journal of Symbolic Logic, 51(3), 770–777, 1986.

[15] Carl G Jockusch, Jr. *Degrees of generic sets.* In *Recursion Theory: its Generalisations and Applications*, Cambridge University Press, London Mathematical Society Lecture Note Series, 110–139, 1980.

[16] Carl G Jockusch, Jr. and David Posner. *Double jumps of minimal degrees.* The Journal of Symbolic Logic, 43(4), 715–724, 1978.

[17] Masahiro Kumabe *A 1-generic degree which bounds a minimal degree.* The Journal of Symbolic Logic, 55(2), 733–743, 1990.

[18] Stuart Kurtz. *Randomness and genericity in the degrees of unsolvability.* PhD thesis, University of Illinois, Urbana-Champaign, 1981.

[19] Stuart Kurtz. *Notions of weak genericity.* The Journal of Symbolic Logic, 48, 764–770, 1983.

[20] Michael McInerney. *Topics in algorithmic randomness and computability theory.* PhD thesis, Victoria University of Wellington, 2016.

[21] Michael McInerney and Keng Meng Ng. *Separating weak α-change and α-change genericity, submitted*

[22] Andre Nies. Lowness properties and randomness. *Advances in Mathematics*, 197 (2005), 274–305.

[23] Andre Nies. *Computability and Randomness.* Oxford University Press, 2009.

[24] Benjamin Schaeffer. *Dynamic notions of genericity and array noncomputability.* Annals of Pure and Applied Logic, vol. 96, 37–69, 1998.

[25] Richard A. Shore and Theodore A. Slaman. *Defining the Turing jump.* Math. Res. Lett., 6(5-6) : 711–722, 1999.

School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore