MAXIMALITY AND COLLAPSE IN THE HIERARCHY OF α-C.A. DEGREES

KATHERINE ARTHUR, ROD DOWNEY AND NOAM GREENBERG

ABSTRACT. In [DG18, DG20], Downey and Greenberg define a transfinite hierarchy of low_2 c.e. degrees — the totally α-c.a. degrees, for appropriately small ordinals α. This new hierarchy is of particular interest because it has already given rise to several natural definability results, and provides a new definable antichain in the c.e. degrees. Several levels of this hierarchy contain maximal degrees. We discuss how maximality interacts with upper cones, and the related notion of hierarchy collapse in upper cones. For example, We show that there is a totally ω-c.a. degree above which there is no maximal totally ω-c.a. degree. We resolve several problems left open in [DG20].

1. INTRODUCTION

One underlying theme in mathematical logic is calibrating mathematical objects in various hierarchies. In computability theory this method also seems to give alignment between syntactical and algorithmic complexity of the objects in question. For instance, the arithmetical hierarchy aligns itself with iterations of the Turing Jump.

In this spirit, building on earlier work of Downey, Greenberg and Weber [DGW07], Downey and Greenberg [DG18, DG20] introduced a new transfinite hierarchy of (mainly computably enumerable) degrees based around considerations about the "mind-change" functions of computable approximations of Δ^0_2 functions. In [DG20], it is argued that the the hierarchy is both natural and significant because the hierarchy

(i) Unifies the combinatorics of a wide class of constructions, both in the Turing degrees and in areas of applications of computability theory, such as effective model theory and algorithmic randomness.
(ii) Gives a number of new examples of *natural* definability of degree classes in the c.e. degrees.
(iii) Gives such results within the low_2 c.e. degrees; classes to which general metatheorems do not pertain.

Of course, this "mind change" hierarchy was build on the ideas of many early studies. These studies began with Ershov [Ers68a, Ers68b, Ers70], and Epstein, Haass, and Kramer [EHK81], which extended the idea of a d.c.e. set, which is one with a computable approximation where elements begin outside, can enter, and then might leave. Indeed, the Limit Lemma says that each set $A \leqslant_{\mathrm{T}} \emptyset'$ has an approximation where each element might enter and leave a finite number of times. The idea of Ershov, and Epstein et al. was to give a fine-grained analysis to

This work is supported by Marsden Grants to Downey and to Greenberg, and a Rutherford Discovery Fellowship to Greenberg. The results form Arthur's MSc Thesis [Art].

claibrate such approximations. Further work along these lines can be found in work of Selivanov [Sel89], and many authors, such as Shore, Lachlan and others, have studied d.c.e. and ω-c.a. sets and degrees. Some of this work we mention below. The key insight in the Downey-Greenberg Hierarchy work is the use of canonical notations (as explained below) to provide a foundation for the classification work.

In [DG20], a number of apparently difficult and natural questions were left open about the structure of the hierarchy. Several of these questions involved notions of maximality in this hierarchy. In this paper we solve a number of these questions, and hence this paper complements the monograph [DG20] in an essential way. The overarching goal of the present paper is to understand the extent to which the hierarchy collapses, and to identify unusual or interesting features.

1.1. **Background: array computable, and totally ω-c.a. degrees.** The Shoenfield Limit Lemma [Sho59] states that a function $f\colon \omega \to \omega$ is computable from the halting problem $\emptyset'$ if and only if it has a *computable approximation*: a uniformly computable sequence $\langle f_s \rangle$ such that for all x, $\lim_s f_s(x) = f(x)$ (in the sense that for all but finitely many s, $f_s(x) = f(x)$). The *mind-change function* of this approximation is $\lambda x.\#\{s \mid f_s(x) \neq f_{s+1}(x)\}$, the number of times the approximation 'changes its mind' on value x. The intuition is that if a function only has approximations that require many changes, then it is relatively complicated. Indeed, a function is weak-truth-table reducible to $\emptyset'$ if and only if it has a computable approximation whose mind-change function is bounded by a computable function. Such functions are also called *ω-computably approximable*.

A c.e. Turing degree $\mathbf{d}$ is *array computable* [DJS90] if and only if every function $f \in \mathbf{d}$ has a computable approximation whose mind-change function is bounded by the identity function (any fixed *order function*, a computable, non-decreasing and unbounded function, would do). Array computability captures the dynamics of multiple permitting characterising several constructions in computability. Thus, for example, a c.e. Turing degree is array computable if and only if it is the degree of a perfect, thin Π^0_1 class, if and only if it contains a c.e. set of maximal Kolmogorov complexity [Kum96], if and only if it does not have a strong minimal cover in the Turing degrees [Ish99]. Several other equivalences are listed in [DG20].

Weakening array computability, a c.e. degree is *totally ω-computably approximable* if every function $f \in \mathbf{d}$ is ω-computably approximable (c.a. for short), a notion suggested by Joseph Miller. The difference is that we no longer require a uniform bound on the mind-change functions. Like array computability, this notion captures the dynamic combinatorics of a number of constructions, involving, for example, presentations of left-c.e. reals. Further, one such construction gives a natural definition of the class of totally ω-c.a. degrees within the c.e. degrees: these are the degrees that do not bound a critical triple, a lattice-theoretic notion useful in embedding lattices in the c.e. degrees.

1.2. **Totally α-c.a. degrees.** Considerations of embeddings of the 1-3-1 lattice, closely related to critical triples, led Downey and Greenberg to define a further weakening of the notion of totally ω-c.a. degrees, and by generalisation, to a transfinite hierarchy of such lowness-like notions. The generalisation utilises the Ershov hierarchy. Much like complexity related to the mind-change function, Ershov calibrates the complexity of a computable approximation, and by extension, of the function being approximated, by considering well-founded witnesses to the approximation settling down. Namely, for a computable ordinal α, an *α-computable approximation* is a computable approximation $\langle f_s \rangle$ of a Δ^0_2 function f, accompanied by a uniformly computable sequence $\langle o_s \rangle$ of functions $o_s \colon \omega \to \alpha$ such that for all x and s, $o_{s+1}(x) \leqslant o_s(x)$ and if $f_{s+1}(x) \neq f_s(x)$ then $o_{s+1}(x) < o_s(x)$. Thus $\langle o_s(x) \rangle_s$ is a 'counting down' in α, which guarantees that $\langle f_s(x) \rangle$ will stabilise. The longer α is, the more space for mind-changes the approximation has. A function is *α-c.a.* (or Δ^{-1}_α) if it has an α-computable approximation, and a c.e. degree is *totally α-c.a.* if every $f \in \mathbf{d}$ is α-c.a.

An important caveat is that care has to be taken in the choice of the computable presentation of an ordinal α; such a presentation of course is required to make sense of computability of functions with range α. In general, many presentations are very much nonequivalent, in that they yield different classes of functions and degrees. Indeed, every low$_2$ c.e. degree is totally ω^2-c.a. for some particular presentation of the ordinal ω^2. The point is that the complexity of an approximation can be coded into the presentation of ω^2; such a presentation can be bad, for example, in that we cannot effectively tell, for each element, which copy of ω it lies in. When defining the notion of totally α-c.a. degrees, it is important to choose canonical presentations of the ordinals α; these yield robust notions that also behave well with respect to extending or shrinking the ordinals. A choice of canonical representations can be easily made below the ordinal ϵ_0 by requiring Cantor normal form to be effective. The details are given in [DG20]. Such canonical copies also allow us to characterise the lower levels of the hierarchy in terms of mind-change functions: a function is ω^{n+1}-c.a. if and only if it has a computable approximation whose mind-change function is bounded by some ω^n-c.a. function.

1.3. **Hierarchy collapse.** One of the first questions to ask is whether the hierarchy is proper. At what levels is there *collapse*, in that no new degrees inhabit the level? This question was settled in [DG20]. For $\alpha \leqslant \epsilon_0$, call a degree $\mathbf{d}$ *properly totally α-c.a.* if it is totally α-c.a. but not totally β-c.a. for any $\beta < \alpha$. Recall that an ordinal is closed under ordinal addition if and only if it is an ordinal power of ω.

Theorem 1.1 ([DG20])**.** *Let $\alpha \leqslant \epsilon_0$. There is a properly totally α-c.a. degree if and only if α is a power of ω.*

Thus, the first proper levels of the hierarchy are the totally ω-c.a., totally ω^2-c.a., $\omega^3, \omega^4, \dots, \omega^\omega, \omega^{\omega+1}, \dots$.

Next, it is natural to check what happens to the hierarchy in upper and lower cones. In this paper we show that the hierarchy does not collapse in lower cones. Henceforth, all degrees are c.e.

Theorem 1.2. *Let $\alpha < \gamma \leqslant \epsilon_0$ be powers of ω. Every properly totally γ-c.a. degree bounds a properly totally α-c.a. degree.*

We remark that the degree produced for Theorem 1.2 is proper in a strong way: it is not uniformly totally α-c.a. See Section 6.

We do not know whether no collapse occurs in upper cones. We prove the following two results:

Theorem 1.3. *Let $\alpha < \beta \leqslant \epsilon_0$ be powers of ω, and suppose that $\beta \geqslant \alpha^\omega$. Then every totally α-c.a. degree is bounded by a properly totally β-c.a. degree.*

In particular, above every totally ω-c.a. degree there are properly ω^ω-c.a. degrees, properly $\omega^{\omega+1}$-c.a. degrees, and so on.

Theorem 1.4. *Let $\alpha < \epsilon_0$ be a power of ω. Above every totally α-c.a. degree there is a totally $(\alpha^3 \cdot \omega)$-c.a. degree which is not totally α-c.a.*

That is, above every totally ω^δ-c.a. degree there is one which is totally $\omega^{\delta\cdot 3+1}$-c.a. but not ω^δ-c.a. So for example, above every totally ω-c.a. degree there is a totally ω^4-c.a. degree which is not totally ω-c.a.; we do not know if this degree can always be made totally ω^2-c.a., or totally ω^3-c.a., or not. So we ask:

Question 1.5. *Is every totally ω-c.a. degree bounded by a properly totally ω^2-c.a. degree?*

If the degree we start with is superlow, then the answer is positive; see Section 4.

1.4. **Maximal degrees.** In [DG20] it is shown that for every $\alpha \leqslant \epsilon_0$ which is a power of ω, there are *maximal* totally α-c.a. degrees: degrees $\mathbf{a}$ which are totally α-c.a., but such that no $\mathbf{b} > \mathbf{a}$ is totally α-c.a. We remark that since the totally ω-c.a. degrees are definable in the c.e. degrees, this gives us a naturally definable antichain in the c.e. degrees. The only other known example of such an antichain is the collection of maximal contiguous degrees [CDW02].

In [DG20] the following is proved:

Theorem 1.6 ([DG20])**.** *Let $\alpha \leqslant \epsilon_0$. Every totally ω^α-c.a. degree lies strictly below a totally $\omega^{\alpha+1}$-c.a. degree.*

As a corollary we see:

(1) If $\alpha < \beta \leqslant \epsilon_0$ are powers of ω, then no totally α-c.a. degree can be maximal totally β-c.a.; and
(2) Every maximal totally α-c.a. degree is properly totally α-c.a.

Thus maximality is closely related to hierarchy collapse. Indeed, to prove Theorem 1.3, we show:

Theorem 1.7. *Let $\alpha, \beta \leqslant \epsilon_0$ be powers of ω, and suppose that $\beta \geqslant \alpha^\omega$. Then every totally α-c.a. degree is bounded by a maximal totally β-c.a. degree.*

On the other hand we show

Theorem 1.8. *Let $\alpha \leqslant \epsilon_0$ be a power of ω. There is a totally α-c.a. degree which is not bounded by any maximal totally α-c.a. degree.*

Toward proving Theorem 1.8 we examine 'maximal covers', see Theorem 5.1.

Similarly to Question 1.5, we ask:

Question 1.9. *Is every totally ω-c.a. degree bounded by a maximal totally ω^2-c.a. degree?*

A positive solution will necessarily be non-uniform. Again, we get a positive answer for superlow degrees.

Solving these problems required several new techniques. For instance, the proof of Theorem 1.7 is the first example of a construction of degrees in one of these bounded classes where there is infinitary positive activity along the true path of the strategy tree. We believe that our techniques will have wider applications.

Notation; Enumerating functions. Throughout, we mostly follow the notation from [DG20]. In particular, we use lower-case Greek letters to denote the use of the corresponding upper-case functionals. We interpret the use differently, depending on the situation:

- If a functional Φ is given to us, or the oracle X is given to us, then for any input x, the use $\varphi(x)$ is the smallest number not queried. So a computation $\Phi(X, x)[s]$ is preserved if $X_s \upharpoonright \varphi_s(x)$ is preserved. In this case we always assume that the domain of $\Phi(X)$ is an initial segment of ω.
- When a functional Λ that we build applies to a set D that we are enumerating, then the use is the only number queried. So if we define a computation $\Phi_s(D_s, x)$ with use $\lambda_s(x)$, then the computation is valid until we enumerate $\lambda_s(x)$ into D. When we set the use to be -1 we mean that $\Phi(D, x)\downarrow$ no matter what D is.
- Sometimes, functionals that we build will take more than one oracle; some we build, some we don't. We will usually apply the second convention, but this will be specified.

For diagonalising against α-c.a. functions, we use enumerations of approximations. We use the terminology from [DG20, Sec.2.1.2]. An $(\alpha + 1)$-computable approximation $\langle f_s, o_s\rangle$ is *tidy* if f_0 is the constant function 0, and for all n and s, if $o_s(n + 1) < \alpha$ then $o_s(n) < \alpha$. Such an approximation is *eventually α-computable* if for all n there is some s such that $o_s(n) < \alpha$. If the approximation is not eventually α-computable then for all but finitely many n, for all s, $o_s(n) = \alpha$ and so $f = \lim_s f_s$ is eventually constant 0. Thus, the limit of any tidy $(\alpha+1)$-computable approximation is α-c.a. We can think of these as partial α-computable approximations; $o_s(n) = \alpha$ means that we have not yet given an ordinal bound for the α-computable approximation. Tidy approximations can be enumerated effectively:

Proposition 1.10 ([DG20]). *For every $\alpha \leqslant \epsilon_0$, a power of ω, there is a sequence $\left\langle f_s^{\alpha,0}, o_s^{\alpha,0}\right\rangle, \left\langle f_s^{\alpha,1}, o_s^{\alpha,1}\right\rangle, \dots$ of tidy $(\alpha + 1)$-computable approximations, uniformly computable (in both their index and in α), such that for every α-c.a. function f there is some i such that $f = f^{\alpha,i} = \lim_s f_s^{\alpha,i}$, and the approximation $\left\langle f_s^{\alpha,i}, o_s^{\alpha,i}\right\rangle$ is eventually α-computable.*

2. Non-collapsing in upper cones

In this section we prove Theorem 1.4. We prove a slightly more general theorem, which we believe will make the ordinal combinatorics clearer.

Theorem 2.1. *Let $\alpha \leqslant \beta < \epsilon_0$ be powers of ω. Above every totally α-c.a. degree there is a degree which is totally $(\alpha\beta\alpha\omega)$-c.a. but not totally β-c.a.*

For Theorem 1.4 let $\beta = \alpha$.

To prove the theorem, fix a c.e. set A (with an effective enumeration $\langle A_s\rangle$) whose degree is totally α-c.a. We enumerate a c.e. set D such that $\deg_{\mathrm{T}}(A \oplus D)$ is totally $(\alpha\beta\alpha\omega)$-c.a. and not totally β-c.a.

Requirements. Let $\langle \Phi_d\rangle_{d<\omega}$ be an enumeration of all Turing functionals. To make $A \oplus D$ totally $(\alpha\beta\alpha\omega)$-c.a., we need to ensure,

$$\text{for all } d < \omega,\ Q_d\text{: If } \Phi_d(A, D) \text{ is total, then it is } (\alpha\beta\alpha\omega)\text{-c.a.}$$

To make $A \oplus D$ not totally β-c.a., we enumerate a functional Λ and ensure that $\Lambda(D)$ is not β-c.a. We will of course need to ensure that $\Lambda(D)$ is total.[1] We diagonalise against the approximations $\langle f_s^{\beta,i}, o_s^{\beta,i}\rangle$ given by Proposition 1.10. We need to ensure,

for all $i < \omega$, P^i: If $\langle f_s^{\beta,i}, o_s^{\beta,i}\rangle$ is eventually β-computable, then $\Lambda(D) \neq f^{\beta,i}$.

Discussion. The proof uses some ideas from the proofs of Theorems 3.6 and 4.12 from [DG20].

We build a tree of strategies. A node σ working for a requirement P^i appoints a follower p with the aim of ensuring that $\Lambda(D,p) \neq f^{\beta,i}(p)$. Thus, whenever σ sees that $\Lambda(D,p)[s] = f_s^{\beta,i}(p)$, we enumerate $\lambda_s(p)$ into D_{s+1} and redefine $\Lambda(D,p)[s+1]$ to be large, and so not equal to $f_s^{\beta,i}(p)$. Thus, if t is a future stage at which σ acts again, we have $o_s^{\beta,i}(p) > o_t^{\beta,i}(p)$; so σ acts only finitely often for the follower p, and in fact the ordinal $\beta+1$ gives a bound on the "number of times" σ can act for p.

A node τ working for Q_d measures the totality of $\Phi_d(A,D)$. Since A is not under our control, this cannot be done in a Σ_2/Π_2 fashion. If total, then for every input x, a computation $\Phi_d(A,D,x)$ will eventually be 'certified', and henceforth the approximation for $\Phi_d(A,D,x)$ will occur on τ-expansionary stages. When certifying the computation, the node τ needs to provide an ordinal below $\alpha\beta\alpha\omega$, and decrease this ordinal if a computation seen at an expansionary stage is later destroyed.

Such computations will be destroyed either by an A-change or a D-change. We soon discuss how to get an ordinal bound, below $\beta\alpha\omega$, over the "number" of D-changes that can injure a certified computation $\Phi_d(A,D,x)$. Over A-changes however we have no control. To counter this, we use the fact that A is totally α-c.a. The node τ builds a "shadow functional" $\hat{\Phi}_\tau$ which keeps track of the A-part of the computations $\Phi_d(A,D)$. We will need to ensure that if $\Phi_d(A,D)$ is total, then so is $\hat{\Phi}_\tau(A)$. We will then be able to guess an α-computable approximation $\langle f_s^{\alpha,k}, o_s^{\alpha,k}\rangle$ for $\hat{\Phi}_\tau(A)$. Naively, we copy a computation $\Phi_d(A,D,x)$ to $\hat{\Phi}_\tau(A,x)$ with the same use. If we see an A-change below that use, then we are free to redefine $\hat{\Phi}_\tau(A,x)$ to be large; the next time we see an agreement with $f_s^{\alpha,k}(x)$, the ordinal bound $o_s^{\alpha,k}(x)$ will have decreased. In that way we could hope for an ordinal bound, below α, on the number of injuries to $\Phi_d(A,D,x)$ because of A-changes.

However, sometimes D-changes destroy $\Phi_d(A,D,x)$ but no corresponding A-change occurs. We cannot then redefine $\hat{\Phi}_\tau(A,x)$. As we are dealing with Turing reductions, rather than weak truth-table reductions, the use on a new $\Phi_d(A,D,x)$-computation is larger, and so we cannot track subsequent A-changes on $\hat{\Phi}_\tau(A,x)$. We need to track it on a new input. We thus appoint a *tracker* $c = \mathtt{tr}(\tau,x)$, and keep track of the A-side of $\Phi_d(A,D,x)$ at $\hat{\Phi}_\tau(A,c)$. When D-enumerations occur, we cancel the tracker c and appoint a new one c'. The new tracker comes with a new ordinal bound below α, likely larger than c's bound. Thus between any two D-changes we need to insert a copy of α, counting A-changes for a given tracker. That is, the overall bound covering both A- and D-changes is multiplicative: $\alpha \cdot \beta\alpha\omega$.

D-changes are caused by nodes σ for positive requirements P^i operating below τ. For each such node σ, with a follower p, the ordinal $o_s^i(p)$ gives an ordinal bound $\leqslant \beta$ on such action by σ; taking into account finitely many such nodes, it would

[1]Note that we could have added A as an oracle to Λ. However, we will see that this is not useful.

seem that we can give a bound below $\beta\omega$.[2] Where does the extra α-term in the middle come from?

When a node τ as above certifies a computation $\Phi_d(A, D, x)$, it can observe the nodes σ below it that already have followers, and take into account their action in devising the ordinal bound. However, other σ-nodes that appoint followers later cannot be taken into account, and so we need to make sure that such nodes never injure a confirmed $\Phi_d(A, D, x)$ computation. Consider however what happens when such a computation is destroyed by a D-change. A node σ with a follower p that is not allowed to injure $\Phi_d(x)$, must keep the use $\lambda(p)$ larger than the use $\varphi_d(x)$. But even if we allowed A as an extra oracle for Λ, to keep $\Lambda(A, D)$ total, we need to appoint a new use $\lambda(p)$ immediately when we see this D-change — before we see the new $\Phi_d(x)$ computation and its very large use. This is why adding A as an oracle to Λ is not useful. There doesn't seem to be any other option but for us to discard p in this situation, and choose a new, large follower for σ. This is a new ingredient which does not appear in [DG20].

There will be counting difficulties, but first we need to argue that this process will end: that if σ lies on the true path, that eventually it will get a follower which is never cancelled. First, to this end, we see that even though we appoint a new large follower p' for σ, we cannot allow more computations $\Phi_{d'}(x')$ cause further cancellations; the priority vis-a-vis computations belongs to σ rather than its followers individually. Still, we could imagine a loop: $\Phi_d(x)$ sees an A-change and cancels a follower for σ; later we see a D-change that cancells the tracker for $\Phi_d(x)$; a new tracker will see more A-changes;... However, the magic of the ~~marketplace~~ priority tree ensures this does not happen; the D-change cancelling the tracker for $\Phi_d(x)$ must be caused by some other node $\bar{\sigma}$ which is stronger than σ, and it would initialise σ at that point.

Now for the counting difficulties. Suppose that $\bar{\tau}$ is another node working for some $Q_{\bar{d}}$, with $\bar{d}$ possibly different from d; and suppose that σ *is* allowed to injure $\Phi_{\bar{d}}(\bar{x})$. What is the ordinal count now on the "number of times" that σ will create a D-change injuring such a computation? When a new follower p' is assigned to σ, the ordinal count $o^{\beta,i}(p')$ "resets" back to β. So we need a copy of β for each follower for σ; and we need an ordinal bound on the number of times we will need to appoint a new follower for σ. Each such cancellation is caused by an A-change in a computation such as $\Phi_d(A, D, x)$; and these A-changes are tracked, so we have a bound below α on them. Since α is closed under addition, we would expect that adding these bounds for all pairs (d, x) that can cancel followers for σ, we would get a bound still below α; and then overall we would get a bound below $\beta\alpha$ on the number of times σ will ever act. A delicate timing issue means that when τ first confirms $\Phi_{\bar{d}}(x)$, it may see that σ has already appointed a follower, and so is allowed to injure this computation; but the ordinal bound on the number of times σ will act depends on a computation $\Phi_d(x)$, with τ below $\bar{\tau}$, which has not yet been

[2] The ordinal arithmetic here follows ideas that were used in [DG20]. Let $\sigma_1, \sigma_2, \ldots, \sigma_k$ list, in decreasing strength, the P-nodes below τ which have chosen their follower before the stage at which some x was certified by τ. For each $j \leqslant k$ and stage s, let $\epsilon_{j,s} \leqslant \beta$ be the ordinal provided by σ_j at stage s. If some σ_j acts, then weaker σ_i's are initialised and so can be removed from the list; and the ordinal ϵ_j will decrease. Hence overall, the sum $\sum_{j \leqslant k} \epsilon_{j,s}$, which is smaller than $\beta\omega$, will decrease each time any of the nodes σ_j will act. Alternatively, we can forget about the priority ordering between the nodes, and use *commutative* ordinal addition, based on Cantor normal form, which we review below in page 14).

tracked. Therefore the bound on the number of times σ will replace a follower is $\alpha+1$ rather than α; adding finitely many σ's, we get an ordinal below $\alpha\omega$.

We now turn to the formal details of the construction.

Strategy Tree. A node τ working for requirement Q_d has outcomes ∞ and f, ordered $\infty < \mathrm{f}$, which measure $\limsup_s \operatorname{dom}\Phi_d(A,D)[s]$ (restricted to the τ-stages). The node τ is responsible for the enumeration of the shadow functional $\hat{\Phi}_\tau(A)$. The node $\tau\hat{\ }\infty$ has outcomes ∞_n, f_n for all $n<\omega$, ordered in the manner $\infty_0 < \mathrm{f}_0 < \infty_1 < \mathrm{f}_1 < \cdots$ which guess whether or not $\hat{\Phi}_\tau(A)$ is total. Each $\tau\hat{\ }\infty\hat{\ }\infty_n$ node has outcomes $k<\omega$, ordered $0<1<2<\cdots$, where each node $\tau\hat{\ }\infty\hat{\ }\infty_n\hat{\ }k$ guesses that $\hat{\Phi}_\tau(A) = f^{\alpha,k}$ (recall Proposition 1.10).

The nodes $\tau\hat{\ }\mathrm{f}$, $\tau\hat{\ }\infty\hat{\ }\mathrm{f}_y$ for all y, and $\tau\hat{\ }\infty\hat{\ }\infty_n\hat{\ }k$ for all n,k all work for the next requirement in some ω-list of all requirements.

Since A is low$_2$, the set of indices of functionals Ψ such that $\Psi(A)$ is total is Σ^0_3. We can translate the question of membership in a Π^0_2 set into whether or not a given non-decreasing sequence is bounded. By the recursion theorem, the index of the functional $\hat{\Phi}_\tau$ enumerated by τ is known to us. We thus obtain a computable list $l_s(\tau,n)$ of sequences, non-decreasing in s, such that $\hat{\Phi}_\tau(A)$ is total if and only if the sequence $\langle l_s(\tau,n)\rangle_{s<\omega}$ is unbounded for some n. It is this list of sequences that we check against when $\tau\hat{\ }\infty$ is accessible.

A node σ working for requirement P^i has a unique outcome, which works for the next requirement on our list. As mentioned, such a node σ will appoint a follower p. When action by σ on p's behalf injures a computation $\Phi_d(A,D,x)$ monitored by some τ stronger than σ, we need to cancel its tracker. We keep track of those nodes and computations which are affected by σ. Let $\mathtt{prec}_\infty(\sigma)$ be the set of nodes τ which work for some requirement Q_d such that $\tau\hat{\ }\infty\hat{\ }\infty_n \prec \sigma$ for some $n<\omega$. For every $\tau\in\mathtt{prec}_\infty(\sigma)$ we will define a number $m^\tau_s(\sigma)$. We then let $\mathtt{Pr}_s(\sigma)$ consist of the collection of pairs (d,x) such that there is some $\tau\in\mathtt{prec}_\infty(\sigma)$, working for requirement Q_d, such that $x<m^\tau_s(\sigma)$. The computations $\Phi_d(A,D,x)$ for $(d,x)\in\mathtt{Pr}_s(\sigma)$ are the computations which σ cannot injure. We remark that between stages at which σ is initialised, the numbers $m^\tau_s(\sigma)$ remain constant.

Construction. Let s be a stage. During the stage we define γ_s, the collection of nodes accessible at s, by recursion; and describe how to act for each accessible node. The empty node is always accessible.

Suppose that τ, working for Q_d, is accessible at stage s. Let $t<s$ be the last stage before s at which $\tau\hat{\ }\infty$ was accessible, or $t=0$ if there is no such stage. If $\operatorname{dom}\Phi_d(A,D)[s]\leqslant t$, we let $\tau\hat{\ }\mathrm{f}$ be next accessible (added to γ_s); otherwise, we let $\tau\hat{\ }\infty\in\gamma_s$ instead.

Suppose that $\tau\hat{\ }\infty\in\gamma_s$. Before we determine the next accessible node, we maintain our shadow functional $\hat{\Phi}_\tau$. Let $x<t$ be such that $c=\mathtt{tr}_s(\tau,x)$ is defined (recall that this means that at stage s, c is the $\hat{\Phi}_\tau$-tracker for $\Phi_d(A,D,x)$). If $\hat{\Phi}_\tau(A,c)\!\uparrow[s]$ then we define $\hat{\Phi}_{\tau,s+1}(A_s,c)$ with large value but with use $\varphi_{d,s}(x)$. If any $c<s$ is not currently a tracker for τ and $\hat{\Phi}_\tau(A,c)\!\uparrow[s]$, then we define $\hat{\Phi}_{\tau,s+1}(A_s,c)$ with use 0. And for each $x<s$ for which $\mathtt{tr}_s(\tau,x)$ is undefined (either because x is new, or because the previous tracker was cancelled), we appoint a new, large tracker $\mathtt{tr}_{s+1}(\tau,x)$.

Recall that t is the previous stage at which $\tau\hat{}\infty$ was accessible. Let y be least such that either $\Phi_d(A,D,y)\uparrow[t]$ or the computation $\Phi_d(A,D,y)[t]$ was destroyed since stage t. That is, if $A_t \upharpoonright u \neq A_s \upharpoonright u$ or $D_t \upharpoonright u \neq D_s \upharpoonright u$ where $u = \varphi_{d,t}(y)$.

For each $n < s$, let t_n be the last stage before s at which $\tau\hat{}\infty\hat{}\infty_n$ was accessible, or $t_n = 0$ if there is no such stage. If there is some $n \leqslant y$ such that $l_s(\tau, n) \geqslant t_n$, then we guess that the sequence $l(\tau, n)$ is unbounded (and so that $\hat{\Phi}_\tau(A)$ is total), and we let $\tau\hat{}\infty\hat{}\infty_n$ be next accessible for the least such n. If there is no such $n \leqslant y$, then we let $\tau\hat{}\infty\hat{}\mathrm{f}_y$ be next accessible.

Suppose now that $\tau\hat{}\infty\hat{}\infty_n$ is accessible at s for some n. For each $k < s$, let r_k be the last stage before s at which $\tau\hat{}\infty\hat{}\infty_n\hat{}k$ was accessible, or $r_k = 0$ if there was no such stage. We let $\tau\hat{}\infty\hat{}\infty_n\hat{}k$ be next accessible for the least $k \leqslant s$ such that for all $x < r_k$, $c = \mathtt{tr}_s(\tau, x)$ is defined, $o_s^{\alpha,k}(c) < \alpha$, and $\hat{\Phi}_\tau(A,c)[s]\downarrow = f_s^{\alpha,k}(c)$. We note that $r_s = 0$, so such an k must exist, even if it merely satisfies these conditions trivially.

Let σ working for P^i be accessible at s. One of several cases will apply to σ.

(i) s is the first stage, after the last stage at which σ was initialised, at which σ is accessible. For all $\tau \in \mathtt{prec}_\infty(\sigma)$ we let $m^\tau_{s+1}(\sigma) = \operatorname{dom}\Phi_d(A,D)[s]$, where τ works for Q_d. We appoint a new, large follower p for σ, and define $\Lambda(D,p)[s{+}1]$ with large value and use. We end the stage.

(ii) σ has a follower p, but for some $(d,x) \in \mathtt{Pr}_s(\sigma)$, $\varphi_{d,s}(x) > \lambda_s(p)$. We will observe that for all $(d,x) \in \mathtt{Pr}_s(\sigma)$, $\Phi_d(A,D,x)\downarrow[s]$. We cancel the follower p, and appoint a new, large follower q; we define $\Lambda(D,q)[s+1]$ with large value and use. We end the stage.

(iii) σ has a follower p, case (ii) fails, and $\Lambda(D,p)[s] \neq f_s^{\beta,i}(p)$. We let σ's unique successor on the tree of strategies be next accessible.

(iv) σ has a follower p, case (ii) fails, and $\Lambda(D,p)[s] = f_s^{\beta,i}(p)$. We enumerate $\lambda_s(p)$ into D_{s+1} and redefine $\Lambda(D,p)[s+1]$ with large value and use. For every $\tau \in \mathtt{prec}_\infty(\sigma)$, for every $x \geqslant m^\tau_s(\sigma)$, we cancel the tracker $\mathtt{tr}_s(\tau,x)$. We then end the stage.

Note that we eventually run into a new long node σ, so each stage has only finitely many steps. At the conclusion of stage s, we maintain the functional Λ to ensure its totality: for any $q \leqslant s$ which is not currently a follower for any node σ and such that $\Lambda(D,q)\uparrow[s]$, we define $\Lambda(D,q)[s+1]$ with use -1. Finally, we initialise all nodes that are (strictly) weaker than the last accessible node. When a node σ with a follower is initialised, its follower is cancelled. When a node τ is initialised, all its trackers are cancelled.

Verification. We introduce terminology: for a node μ on the tree, we denote s to be a *μ-stage* if $\mu \in \gamma_s$.

Lemma 2.2. *Let σ be a node working for requirement P^i. Let s be a σ-stage such that $\mathtt{Pr}_s(\sigma)$ is defined. Then for all $(d,x) \in \mathtt{Pr}_s(\sigma)$, $\Phi_d(A,D,x)\downarrow[s]$. Further, if $\tau \in \mathtt{prec}_\infty(\sigma)$ works for Q_d, and $\tau\hat{}\infty\hat{}\infty_n\hat{}k \preccurlyeq \sigma$, then $c = \mathtt{tr}_s(\tau,x)$ is defined, $\hat{\Phi}_\tau(A,c)\downarrow[s] = f_s^{\alpha,k}(c)$, and $o_s^{\alpha,k}(c) < \alpha$.*

Proof. Let $t < s$ be the first σ-stage after the last time (prior to s) at which σ was initialised. So at stage t we define the value $m^\tau_{t+1}(\sigma) = m^\tau_s(\sigma)$. Let $x < m^\tau_s(\sigma)$. We define $m^\tau_{t+1}(\sigma) = \operatorname{dom}\Phi_d(A,D)[t] < t$. If $s > t$ then as s is a $\tau\hat{}\infty$-stage,

$\operatorname{dom}\Phi_d(A,D)[s] > t > m_s^\tau(\sigma) > x$. Similarly, the rest follows from the fact that both t and s are $\tau\hat{}\infty\hat{}\infty_n\hat{}k$-stages. ☐

Lemma 2.3. *For any node σ, if p is a follower for σ at the beginning of stage s, then $\Lambda(D,p)\downarrow[s]$.*

Proof. Formally, by induction on s. When σ appoints p, or enumerates $\lambda_s(p)$ into D_{s+1}, a new computation $\Lambda(D,p)[s+1]$ is defined. We just need to note that unless σ acts for p at stage s, then $\lambda_s(p)\notin D_{s+1}$; the point is that if p' is a follower for σ' at the beginning of stage s then $\lambda_s(p)\neq\lambda_s(p')$, as all uses are chosen large. ☐

Lemma 2.4. *Let σ be a node working for some P^i. Let s be a σ-stage, and suppose that at the beginning of stage s, σ has a follower p. Let τ be a node working for Q_d such that $\tau\hat{}\infty\hat{}\mathrm{f}_y \preccurlyeq \sigma$, and let $x<y$. Then $\Phi_d(A,D,x)\downarrow[s]$ and $\varphi_{d,s}(x)<\lambda_s(p)$.*

Proof. Let $t<s$ be the stage at which the stage s use $\lambda_s(p)$ was defined; so t was a σ-stage and $t<\lambda_{t+1}(p)=\lambda_s(p)$ (as new uses are chosen large). Since $\tau\hat{}\infty\hat{}\mathrm{f}_y\in\gamma_t$, $y\leqslant\operatorname{dom}\Phi_d(A,D)[t]$, so $\Phi_d(A,D,x)\downarrow[t]$. Let $u=\varphi_{d,t}(x)<t$. So $u<\lambda_s(p)$. It suffices to show that $A_t\restriction u = A_s\restriction u$ and $D_t\restriction u = D_s\restriction u$. If not, then let $r>t$ be the least $\tau\hat{}\infty$-stage at which we see either $A_t\restriction u\neq A_r\restriction u$ or $D_t\restriction u\neq D_r\restriction u$. Then at stage $r\leqslant s$, either $\tau\hat{}\infty\hat{}\infty_n\in\gamma_s$ or $\tau\hat{}\infty\hat{}\mathrm{f}_n\in\gamma_s$ for some $n\leqslant x<y$. This outcome lies to the left of the outcome f_y, and so $r<s$ and at stage r, σ is initialised, which is impossible. ☐

We need to be sure that if a computation is destroyed by a change in D, its tracker is cancelled immediately to allow us to correctly anticipate further A-changes.

Lemma 2.5. *Let τ be a node working for requirement Q_d. Let s be a stage; let $x<\omega$ be such that $c=\mathtt{tr}_s(\tau,x)$ is defined. Suppose that $\hat\Phi_\tau(A,c)\downarrow[s]$, and let $u=\hat\varphi_{\tau,s}(c)$. Then:*

(i) $\Phi_d(A,D,x)\downarrow[s]$ and $u=\varphi_{d,s}(x)$.
(ii) If $D_s\restriction u\neq D_{s+1}\restriction u$ then the tracker c is cancelled at stage s.

Proof. Suppose that (i) and (ii) hold up to stage s, and that the lemma's hypotheses hold at s. Let $t<s$ be the stage at which the computation $\hat\Phi_\tau(A,c)[s]$ was defined; then $A_t\restriction u = A_s\restriction u$. Since trackers are chosen to be large, $c=\mathtt{tr}_t(\tau,x)$. At the $\tau\hat{}\infty$-stage t we define $u=\hat\varphi_{\tau,t+1}(c)=\varphi_{d,t}(x)$. Since the tracker c was not cancelled at any stage prior to s, by induction, $D_t\restriction u = D_s\restriction u$. The constancy of both A and D imply that $\varphi_{d,s}(x)=\varphi_{d,t}(x)=u$.

For (ii), suppose that $D_s\restriction u\neq D_{s+1}\restriction u$; then at stage s we enumerate a number $\lambda_s(p)<u$ into D, where p is a follower for a node σ, which works for a requirement P^i.

Since $\lambda_s(p)<u=\varphi_{d,t}(x)$, we know that the use $\lambda_s(p)$ was chosen prior to stage t; so p was chosen as a follower for σ prior to stage t.

Now we ask: how does τ relate to σ?

If σ lies to the right of $\tau\hat{}\infty$, then σ is initialised at stage t; this is impossible. If τ is weaker than σ, then τ is initialised at stage s; in particular, c is cancelled at s. If not, then $\tau\hat{}\infty\prec\sigma$.

Suppose that $\tau\hat{}\infty\hat{}\mathrm{f}_y\preccurlyeq\sigma$. Let $\bar s<s$ be the last $\tau\hat{}\infty$-stage prior to stage s. Since t is a $\tau\hat{}\infty$-stage, $t\leqslant\bar s$. Since the computation $\Phi_d(A,D,x)[t]$ still holds at

stage s, it is unchanged between stages $\bar{s}$ and s. Now f_y is the outcome chosen at stage s; so either $\Phi_d(A, D, y)\uparrow [\bar{s}]$ or the computation $\Phi_d(A, D, y)[\bar{s}]$ does not persist at t. We assume that the functional Φ_d is use-monotone, so we conclude that $x < y$. By Lemma 2.4, $\lambda_s(p) > \varphi_{d,s}(x)$, which is not the case.

Hence $\tau \in \mathtt{prec}_\infty(\sigma)$. By our instructions for σ, since p is not cancelled at stage s, $(d, x) \notin \mathtt{Pr}_s(\sigma)$, i.e., $x \geqslant m^\tau_s(\sigma)$. At stage s, σ is instructed to cancel c. $\square$

Corollary 2.6. *Let τ be a node working for requirement Q_d. Let $x < \omega$, and suppose that at some point a tracker $c = \mathtt{tr}(\tau, x)$ is defined and is never cancelled. Suppose that $\hat{\Phi}_\tau(A, c)\downarrow$. Then $\Phi_d(A, D, x)\downarrow$.*

Let the true path, γ_ω, consist of the nodes μ such that:

(i) $\mu \in \gamma_s$ for infinitely many s; and
(ii) μ is initialised only finitely often.

If μ lies to the right of some node in γ_s then μ is initialised at stage s. It follows that γ_ω is a path: linearly ordered by extension of nodes, and closed under taking initial segments. Note that we have not yet shown that γ_ω is infinite.

Lemma 2.7. *Let σ be a node which works for P^i, and let p be a follower for σ. There are only finitely many stages s at which σ acts on p's behalf by enumerating $\lambda_s(p)$ into D_{s+1}.*

Proof. Let $s_0 < s_1 < \cdots$ be the stages s at which σ enumerates $\lambda_s(p)$ into D_{s+1}. If σ acts thus at a stage s, it must be because $\Lambda(D, p)[s] = f^{\beta,i}_s(p)$. We then redefine $\Lambda(D, p)[s + 1]$ to be large, and so greater than $f^{\beta,i}_s(p)$. So $f^{\beta,i}_{s_0}(p) \neq f^{\beta,i}_{s_1}(p) \neq \cdots$, and hence $o^{\beta,i}_{s_0}(p) > o^{\beta,i}_{s_1}(p) > \cdots$ — implying that the sequence $s_0, s_1, \ldots$ must be finite. $\square$

Lemma 2.8. *Let σ be a node working for requirement P^i. Let $s < t$ be σ-stages such that $\mathtt{Pr}_s(\sigma)$ is defined and σ is not initialised at any stage $r \in [s, t)$. Let $(d, x) \in \mathtt{Pr}_s(\sigma)$; let $\tau \in \mathtt{prec}_\infty(\sigma)$ work for Q_d. Then $\mathtt{tr}_s(\tau, x) = \mathtt{tr}_t(\tau, x)$.*

Note that $\mathtt{tr}_s(\tau, x)$ is defined by Lemma 2.2.

Proof. Suppose not; let $r \in [s, t)$ be the stage at which $\mathtt{tr}_s(\tau, x) = \mathtt{tr}_r(\tau, x)$ is cancelled. Let $\bar{\sigma}$ be the node which causes this cancellation. So $\tau \in \mathtt{prec}_\infty(\bar{\sigma})$ and $x \geqslant m^\tau_r(\bar{\sigma})$. By assumption, $x < m^\tau_s(\sigma) = m^\tau_r(\sigma)$. Hence $\sigma \neq \bar{\sigma}$. Since σ is not initialised at stage r, $\bar{\sigma}$ is weaker than σ. Let $v < s$ be the stage at which $m^\tau_s(\sigma)$ is defined. Then $\bar{\sigma}$ is initialised at stage v. Hence $m^\tau_r(\bar{\sigma})$ is defined at a stage $\bar{v} > v$. As both v and $\bar{v}$ are $\tau\hat{\ }\infty$-stages,

$$x \geqslant m^\tau_{\bar{v}}(\bar{\sigma}) = \operatorname{dom} \Phi_d(A, D)[\bar{v}] > v > m^\tau_v(\sigma) = m^\tau_s(\sigma) > x,$$

a contradiction. $\square$

The following lemma fits here but will only be used later.

Lemma 2.9. *Let σ be a node working for requirement P^i. Let $s < t$ be σ-stages such that $\mathtt{Pr}_s(\sigma)$ is defined and σ is not initialised at any stage $r \in [s, t)$. Suppose that a follower for σ is cancelled at stage t. Then there is some $(d, x) \in \mathtt{Pr}_s(\sigma)$ such that $o^{\alpha,k}_s(c) > o^{\alpha,k}_t(c)$, where $\tau \in \mathtt{prec}_\infty(\sigma)$ works for Q_d, $\tau\hat{\ }\infty\hat{\ }\infty_n\hat{\ }k \preccurlyeq \sigma$, and $c = \mathtt{tr}_t(\tau, x)$.*

Proof. σ's follower p is cancelled at stage t because $\varphi_{d,t}(x) > \lambda_t(p)$ for some $(d,x) \in \mathtt{Pr}_t(\sigma) = \mathtt{Pr}_s(\sigma)$. Let $\tau \in \mathtt{prec}_\infty(\sigma)$ work for Q_d, and let $n, k < \omega$ such that $\tau\hat{\ }\infty\hat{\ }\infty_n\hat{\ }k \preccurlyeq \sigma$. Let $c = \mathtt{tr}_s(\tau,x) = \mathtt{tr}_t(\tau,x)$ (Lemma 2.8).

We need to argue that $f_s^{\alpha,k}(c) \neq f_t^{\alpha,k}(c)$. By Lemma 2.2, $\hat{\Phi}_\tau(A,c)\downarrow[s] = f_s^{\alpha,k}(c)$, and the same holds at t. So we need to show that $\hat{\Phi}_\tau(A,c)[s] \neq \hat{\Phi}_\tau(A,c)[t]$. By increasing s we may assume that s is the σ-stage prior to t, so p is the follower for σ at stage $s+1$. Either we define $\Lambda(D,p)[s+1]$ at stage s; then its use $\lambda_{s+1}(p) = \lambda_t(p)$ is large, greater than $\varphi_{d,s}(x)$. Or p is already the follower for σ at the beginning of stage s, in which case $\lambda_s(p)$ is already defined; but since p is not cancelled at stage s, we have $\lambda_t(p) = \lambda_s(p) \geqslant \varphi_{d,s}(x)$. In either case we see that $\varphi_{d,s}(x) \leqslant \lambda_t(p) < \varphi_{d,t}(x)$. By Lemma 2.5, $\hat{\varphi}_{\tau,s}(c) = \varphi_{d,s}(x)$ and the same holds at t. So $\hat{\varphi}_{\tau,s}(c) < \hat{\varphi}_{\tau,t}(c)$. Hence the computation $\hat{\Phi}_\tau(A,c)[s]$ was destroyed and redefined before stage t, with a value large relative to stage s. □

Lemma 2.10. *Let $\sigma \in \gamma_\omega$ be a node working for some P^i. Then σ is eventually assigned a follower which is never cancelled.*

Proof. Let s be the second σ-stage after the last stage at which σ is initialised. By stage s we have already defined $\mathtt{Pr}(\sigma) = \mathtt{Pr}_s(\sigma)$ which equals $\mathtt{Pr}_t(\sigma)$ for all $t \geqslant s$.

Let $(d,x) \in \mathtt{Pr}(\sigma)$; let $\tau \in \mathtt{prec}_\infty(\sigma)$ work for Q_d. By Lemma 2.8, the tracker $c = \mathtt{tr}_s(\tau,x)$ is never cancelled.

Let $n < \omega$ such that $\tau\hat{\ }\infty\hat{\ }\infty_n \prec \sigma$. Since the outcome ∞_n is guessed infinitely often, the sequence $\langle l_t(\tau,n)\rangle_{t<\omega}$ is unbounded. Hence $\hat{\Phi}_\tau(A)$ is total. In particular, $\hat{\Phi}_\tau(A,c)\downarrow$.

By Corollary 2.6, $\Phi_d(A,D,x)\downarrow$. If a follower p is appointed for σ after the correct computation $\Phi_d(A,D,x)$ has appeared, then the pair (d,x) will not be responsible for cancelling p. The lemma follows from the fact that $\mathtt{Pr}(\sigma)$ is finite. □

The following will imply that $A \oplus D$ is low$_2$.

Lemma 2.11. *Suppose that τ works for Q_d and that $\tau\hat{\ }\infty \in \gamma_\omega$.*

1. *Every $x < \omega$ is eventually appointed a tracker which is never cancelled.*
2. *If $\Phi_d(A,D)$ is total then so is $\hat{\Phi}_\tau(A)$, and for some $n < \omega$, $\tau\hat{\ }\infty\hat{\ }\infty_n \in \gamma_\omega$.*
3. *If $\Phi_d(A,D)$ is partial then so is $\hat{\Phi}_\tau(A)$, and $\tau\hat{\ }\infty\hat{\ }\mathrm{f}_y \in \gamma_\omega$, where $y = \operatorname{dom}\Phi_d(A,D)$.*

Proof. Let $x < \omega$. At all but finitely many $\tau\hat{\ }\infty$-stages s, if $\mathtt{tr}_s(\tau,x)$ in undefined, then we define a new tracker $\mathtt{tr}_{s+1}(\tau,x)$. Thus, it suffices to show that a tracker $\mathtt{tr}_s(\tau,x)$ is cancelled only finitely many times.

After the last stage at which τ is initialised, such a tracker can be cancelled only by nodes σ such that $\tau \in \mathtt{prec}_\infty(\sigma)$ and $m_s^\tau(\sigma) \leqslant x$. There are only finitely many such nodes: $\operatorname{dom}\Phi_d(A,D)[s]$ is strictly increasing on the $\tau\hat{\ }\infty$-stages, and so eventually the values $m_t^\tau(\sigma)$ are chosen large. By Lemmas 2.7 and 2.10, each node σ with $m_s^\tau(\sigma) \leqslant x$ cancels x's tracker only finitely many times. This establishes (1).

For (2), suppose that $\Phi_d(A,D)$ is total. Let $c < \omega$. If c is chosen as a tracker for some x and is later cancelled, let t be the stage at which c is cancelled; if c is never chosen as a tracker, let $t = c$. Let s be the least stage $s > t$ at which $\tau\hat{\ }\infty$ is accessible and $\hat{\Phi}_\tau(A,c)\uparrow[s]$. We then define $\hat{\Phi}_{\tau,s+1}(A_s,c) = 0$ with use 0, and so $c \in \operatorname{dom}\hat{\Phi}_\tau(A)$.

Suppose then that c is chosen as a tracker for some x and is never cancelled. At every sufficiently late $\tau\hat{}\infty$-stage s, if $c \notin \operatorname{dom} \hat{\Phi}_\tau(A)[s]$ then we define a new computation $\hat{\Phi}_{\tau,s+1}(A_s, c)$ with use $\varphi_{d,s}(x)$. The computation which is made when we first see the correct $\Phi_d(A, D, x)$-computation is also a correct $\hat{\Phi}_\tau(A, c)$-computation. Overall we see that $\hat{\Phi}_\tau(A)$ is total.

Therefore, there is some $n < \omega$ such that $\langle l_s(\tau, n)\rangle_{s<\omega}$ is an unbounded sequence; let n be least such. Since $\Phi_d(A, D)$ is total, any outcome f_y is guessed only finitely many times. It follows that the outcome ∞_n will be guessed infinitely often.

Now we suppose that $\Phi_d(A, D)$ is partial. Let $y = \operatorname{dom} \Phi_d(A, D)$; let c be the tracker that is eventually assigned to y and is never cancelled. Corollary 2.6 implies that $\hat{\Phi}_\tau(A, c)\uparrow$, so $\hat{\Phi}_\tau(A)$ is partial. Thus, each sequence $\langle l_s(\tau, n)\rangle_{s<\omega}$ is bounded, so each outcome ∞_n is guessed only finitely often. Also, each outcome f_x for $x < y$ is guessed only finitely often. On the other hand, the outcome f_y will be guessed infinitely often. □

Lemma 2.12. *The true path γ_ω is infinite.*

Proof. The empty node lies on the true path. We show that every node on the true path has a child on the true path.

This is clear for a node τ working for some Q_d, as it has only two outcomes. The case $\tau\hat{}\infty \in \gamma_\omega$ is dealt with in Lemma 2.11.

Suppose that $\tau\hat{}\infty\hat{}\infty_n \in \gamma_\omega$. By Lemma 2.11, $\hat{\Phi}_\tau(A)$ is total. Since $\deg_{\mathrm{T}}(A)$ is totally α-c.a., there is some k such that $\hat{\Phi}_\tau(A) = f^{\alpha,k}$ and $\langle f^{\alpha,k}_s, o^{\alpha,k}_s\rangle$ is eventually α-computable (recall that this means that for all c there is some s such that $o^{\alpha,k}_s(c) < \alpha$). Since every input x eventually receives a tracker which is not cancelled, for the least such k, $\tau\hat{}\infty\hat{}\infty_n\hat{}k \in \gamma_\omega$.

Let σ be a node which works for P^i; suppose that $\sigma \in \gamma_\omega$. By Lemmas 2.7 and 2.10, σ acts only finitely many times after the last stage at which it was initialised. Hence for all but finitely many stages s, if $\sigma \in \gamma_s$ then σ's only child is also in γ_s. □

The true path then contains, for every P and Q requirement, a node that works for it. But first we need to show:

Lemma 2.13. *$\Lambda(D)$ is total.*

Proof. Let $p < \omega$. If p is never appointed as a follower, or if p is appointed to a node working for P^i and later cancelled, then $p \in \operatorname{dom}\Lambda(D)$; at a sufficiently late stage t, if $\Lambda(D, p)\uparrow [t]$ then we define $\Lambda(D, p)[t + 1]$ with use -1.

Suppose then that p is appointed as a follower for a node σ working for P^i, and that p is never cancelled. By Lemma 2.3, $p \in \operatorname{dom}\Lambda_s(D)$ at every stage s after the stage at which p was chosen as follower. Since σ acts only finitely often for p (Lemma 2.7), the use $\lambda(p)$ is raised only finitely often. At the last stage t at which it is raised, we define $\Lambda(D, p)[t + 1]$ with a use $\lambda_{t+1}(p)$ which is never enumerated into D, and so this computation is D-correct. □

Lemma 2.14. *Every requirement P^i is met.*

Proof. Fix $i < \omega$, and let σ be the node on the true path working for requirement P^i. By Lemma 2.10, σ has a follower p which is never cancelled. Then $\Lambda(D, p) \neq f^i(p)$: at some sufficiently late σ-stage s we have both $\Lambda(D, p)\downarrow [s]$ via a D-correct computation, and $f^i_s(p) = f^i(p)$; if we had equality, then at stage s, the node σ would act. $\square$

To show that the Q-requirements are met, for simplicity, we use the commutative addition operation on ordinals. For ordinals γ, δ, the commutative ordinal sum $\gamma\oplus\delta$ is determined by their Cantor normal form: if $\gamma = \omega^{\alpha_1}n_1 + \omega^{\alpha_2}n_2 + \cdots + \omega^{\alpha_k}n_k$, with $\alpha_1 > \alpha_2 > \cdots > \alpha_k$ and each $n_i < \omega$, and $\delta = \omega^{\alpha_1}m_1 + \omega^{\alpha_2}m_2 + \cdots + \omega^{\alpha_k}m_k$, then $\gamma \oplus \delta = \omega^{\alpha_1}(n_1 + m_1) + \omega^{\alpha_2}(n_2 + m_2) + \cdots + \omega^{\alpha_k}(n_k + m_k)$.

The following are well known:

- $\oplus$ is commutative and associative.
- Any power of ω is closed under $\oplus$.
- Let $\beta_1, \beta_2, \dots, \beta_n$ and $\gamma_1, \gamma_2, \dots, \gamma_n$ be two n-tuples of ordinals. Suppose that for all $i \leqslant n$, $\beta_i \leqslant \gamma_i$. Then $\bigoplus_{i\leqslant n}\beta_i \leqslant \bigoplus_{i\leqslant n}\gamma_i$, and $\bigoplus_{i\leqslant n}\beta_i < \bigoplus_{i\leqslant n}\gamma_i$ if, and only if, there is some $i \leqslant n$ such that $\beta_i < \gamma_i$.

Lemma 2.15. *Every requirement Q_d is met.*

Proof. Let $d < \omega$, and suppose that $\Phi_d(A, D)$ is total. Let τ be the node on the true path working for Q_d. Then $\tau\hat{\ }\infty \in \gamma_\omega$; by Lemmas 2.11 and 2.12, for some n and $k < \omega$, $\rho = \tau\hat{\ }\infty\hat{\ }\infty_n\hat{\ }k$ lies on the true path. Let $s_0 < s_1 < \cdots$ be the ρ-stages following the last stage at which ρ is initialised. We proceed to define an $(\alpha\beta\alpha\omega)$-computable approximation $\langle f_s, o_s\rangle$ for $\Phi_d(A, D)$.

From now on, fix $x < \omega$. Let $j^* = j^*(x)$ be the least $j > 0$ such that $x < s_{j-1}$. For all $j \geqslant j^*$, $\Phi_d(A, D, x)\downarrow [s_j]$. So for all $j \geqslant j^*$ we let $f_j(x) = \Phi_d(A, D, x)[s_j]$. Certainly $\lim_{j\to\infty} f_j(x) = \Phi_d(A, D, x)$. Let $u_j = u_j(x) = \varphi_{d,s_j}(x)$.

To define the ordinal bound we need to measure both A-changes and D-changes. For the A-changes, we note that for all $j \geqslant j^*$, since $\rho \in \gamma_{s_j}$ and $x < s_{j-1}$, $c_j = c_j(x) = \mathtt{tr}_{s_j}(\tau, x)$ is defined, $o^{\alpha,k}_{s_j}(c_j) < \alpha$, and $\hat{\Phi}_\tau(A, c_j)\downarrow [s_j] = f^{\alpha,k}_{s_j}(c_j)$. By Lemma 2.5, $u_j = \hat{\varphi}_{\tau,s_j}(c_j)$. For brevity, for $j \geqslant j^*$ we let

$$\theta_j = \theta_j(x) = o^{\alpha,k}_{s_j}(c_j).$$

Claim 2.15.1. For all $j \geqslant j^*$, $\theta_j < \alpha$. Suppose that $c_j = c_{j+1}$. Then $\theta_j \geqslant \theta_{j+1}$; and if $A_{s_j} \restriction u_j \neq A_{s_{j+1}} \restriction u_j$ then $\theta_j > \theta_{j+1}$.

Proof. The first part is immediate: $o^{\alpha,k}_{s_j}(c_j) \geqslant o^{\alpha,k}_{s_{j+1}}(c_j)$. The second part is argued like the end of the proof of Lemma 2.9: if $A_{s_j} \restriction u_j \neq A_{s_{j+1}} \restriction u_j$ then the computation $\hat{\Phi}_\tau(A, c_j)[s_j]$ is destroyed before stage s_{j+1} and redefined with large use, so

$$f^{\alpha,k}_{s_j}(c_j) = \hat{\Phi}_\tau(A, c_j)[s_j] \neq \hat{\Phi}_\tau(A, c_j)[s_{j+1}] = f^{\alpha,k}_{s_{j+1}}(c_j). \qquad \square$$

For the D-changes, let $a = a(x)$ be the set of nodes $\sigma \not\succcurlyeq \rho$, working for some requirement $P^i = P^{i(\sigma)}$, such that $\mathtt{Pr}_{s_{j^*}}(\sigma)$ is defined. For all $j \geqslant j^*$ we let $a_j = a_j(x)$ be the set of nodes $\sigma \in a$ which were not initialised at any stage in the interval $[s_{j^*}, s_j)$. So $a_{j+1} \subseteq a_j$.

For $\sigma \in a$, let $\mathtt{Pr}(\sigma) = \mathtt{Pr}_{s_{j^*}}(\sigma)$. If $\sigma \in a_j$ then $\mathtt{Pr}(\sigma) = \mathtt{Pr}_{s_j}(\sigma)$. Let $s_{\mathtt{init}}(\sigma) < s_{j^*}$ be the stage at which $\mathtt{Pr}(\sigma)$ was defined. We let

$$T(\sigma) = \{s_{\mathtt{init}}(\sigma)\} \cup \{s_j \,:\, j \geqslant j^*, \sigma \in a_j,\ s_j \text{ is a } \sigma\text{-stage and } \sigma \text{ ends stage } s_j\}.$$

For each $t \in T(\sigma)$ let $p_t(\sigma)$ be σ's follower at the *end* of stage t. Again for brevity, for $t \in T(\sigma)$ we let

$$\epsilon_t(\sigma) = o^{\beta,i}_t(p_t(\sigma)),$$

where σ works for requirement P^i.

Claim 2.15.2. For all $t \in T(\sigma)$, $\epsilon_t(\sigma) \leqslant \beta$. If $t_0 < t_1$ are stages in $T(\sigma)$ and $p_{t_0}(\sigma) = p_{t_1}(\sigma)$ then $\epsilon_{t_0}(\sigma) > \epsilon_{t_1}(\sigma)$.

Proof. Let $t_0 < t_1$ be stages in $T(\sigma)$ and suppose that $p_{t_0}(\sigma) = p_{t_1}(\sigma)$, call it p. Since p was appointed as follower before stage t_1, and σ acts for p at stage t_1, $\Lambda(D,p)[t_1] = f^{\beta,i}_{t_1}(p)$, where again σ works for P^i. The important fact is that at the end of stage t_0, a new computation $\Lambda(D,p)[t_0+1]$ is defined, with large value (either because p is chosen at stage t_0 or $\lambda_{t_0}(p)$ is enumerated into D_{t_0+1} and a new computation defined). Hence $\Lambda(D,p)[t_1] > t_0$. By convention, $f^{\beta,i}_{t_0}(p) < t_0$ so $f^{\beta,i}_{t_0}(p) \neq f^{\beta,i}_{t_1}(p)$. □

Fix some $\sigma \in a(x)$, and some $(\bar{d},\bar{x}) \in \mathtt{Pr}(\sigma)$. Let $\bar{\tau} \in \mathtt{prec}_\infty(\sigma)$ be the node working for $Q_{\bar{d}}$. For any $t \in T(\sigma)$ with $t > s_{\mathtt{init}}(\sigma)$, $\mathtt{tr}_t(\bar{\tau},\bar{x})$ is defined (Lemma 2.2), and does not depend on t (Lemma 2.8); we denote this tracker, if it exists, by $c(\bar{d},\bar{x},\sigma)$. Let $\bar{k} < \omega$ such that $\bar{\tau}\hat{\ }\infty\hat{\ }\infty_{\bar{n}}\hat{\ }\bar{k} \preccurlyeq \sigma$ for some $\bar{n} < \omega$. For $t \in T(\sigma)$ we let

$$\delta_t(\bar{d},\bar{x},\sigma) = \begin{cases} \alpha, & \text{if } t = s_{\mathtt{init}}(\sigma)\text{; and} \\ o^{\alpha,\bar{k}}_t\big(c(\bar{d},\bar{x},\sigma)\big), & \text{if } t > s_{\mathtt{init}}(\sigma). \end{cases}$$

Note that if $t > s_{\mathtt{init}}(\sigma)$ then $\delta_t(\bar{d},\bar{x},\sigma) < \alpha$ (Lemma 2.2). We then let

$$\delta_t(\sigma) = \bigoplus_{(\bar{d},\bar{x}) \in \mathtt{Pr}(\sigma)} \delta_t(\bar{d},\bar{x},\sigma).$$

Claim 2.15.3. For all $t \in T(\sigma)$, $\delta_t(\sigma) < \alpha\omega$. [3] Let $t_0 < t_1$ be stages in $T(\sigma)$. Then $\delta_{t_0}(\sigma) \geqslant \delta_{t_1}(\sigma)$. If $p_{t_0}(\sigma) \neq p_{t_1}(\sigma)$ then $\delta_{t_0}(\sigma) > \delta_{t_1}(\sigma)$.

Proof. Let $t_0 < t_1$ be stages in $T(\sigma)$, and suppose that $p_{t_0}(\sigma) \neq p_{t_1}(\sigma)$. If $t_0 = s_{\mathtt{init}}(\sigma)$ then for all $(\bar{d},\bar{x}) \in \mathtt{Pr}(\sigma)$, $\delta_{t_0}(\bar{d},\bar{x},\sigma) = \alpha > \delta_{t_1}(\bar{d},\bar{x},\sigma)$ and so certainly $\delta_{t_0}(\sigma) > \delta_{t_1}(\sigma)$. If $t_0 > s_{\mathtt{init}}(\sigma)$ then Lemma 2.9 applies. □

For $t \in T(\sigma)$ let

$$\zeta_t(\sigma) = \beta \cdot \delta_t(\sigma) + \epsilon_t(\sigma).$$

Claims 2.15.2 and 2.15.3 together imply:

Claim 2.15.4. For all $t \in T(\sigma)$, $\zeta_t(\sigma) < \beta\alpha\omega$. If $t_0 < t_1$ are stages in $T(\sigma)$ then $\zeta_{t_0}(\sigma) > \zeta_{t_1}(\sigma)$.

The last fact we need is the following.

Claim 2.15.5. Let $j \geqslant j^*$. Suppose that $\Phi_d(A,D,x)[s_j] \neq \Phi_d(A,D,x)[s_{j+1}]$, and that $s_j \notin \bigcup_{\sigma \in a} T(\sigma)$. Then $\theta_j > \theta_{j+1}$.

Proof. Recall that $u_j = \varphi_{d,s_j}(x)$. By Claim 2.15.1, it suffices to show that $c_j = c_{j+1}$ and that $A_{s_j} \restriction u_j \neq A_{s_{j+1}} \restriction u_j$.

Suppose that $c_j \neq c_{j+1}$. Let σ be the node which cancels c_j, at a stage $t \in [s_j, s_{j+1})$. Since τ is not initialised after stage s_{j^*}, it must be that $\tau \in \mathtt{prec}_\infty(\sigma)$ and $x \geqslant m^\tau_t(\sigma)$. Since $\operatorname{dom} \Phi_d(A,D)[r] > x$ at every $\tau\hat{\ }\infty$-stage $r \geqslant s_{j^*}$, it must

[3] Because α is closed under addition, we could have defined $\delta_{s_{\mathtt{init}}(\sigma)}(\sigma) = \alpha$ rather than $\alpha \cdot |\mathtt{Pr}(\sigma)|$, and obtain the seemingly better bound $\delta_t(\sigma) \leqslant \alpha$. But after adding these ordinals for more than one node σ, this does not give us an overall better bound.

be that $m^\tau_t(\sigma)$ was defined prior to stage s_{j^*}; so $\mathtt{Pr}_{s_{j^*}}(\sigma)$ is defined and σ is not initialised at any stage $r \in [s_{j^*}, t]$.

If σ lies to the right of ρ, then σ is initialised at stage s_{j^*}, which is not the case; and since ρ is not initialised at stage t, σ cannot be stronger than ρ. Hence $\sigma \not\succcurlyeq \rho$. We conclude that $\sigma \in a$ and in fact that $\sigma \in a_j$. Since σ cannot be accessible between stages s_j and s_{j+1}, we must have $t = s_j$; it follows that $s_j \in T(\sigma)$.

Now suppose that $A_{s_j} \restriction u_j = A_{s_{j+1}} \restriction u_j$. Then necessarily $D_{s_j} \restriction u_j \neq D_{s_{j+1}} \restriction u_j$. By Lemma 2.5, c_j would be cancelled prior to stage s_{j+1}, which we just argued is not the case. □

We are now ready to define our ordinal bound. For each $j \geqslant j^*$ and $\sigma \in a$, since $\min T(\sigma) = s_{\mathtt{init}}(\sigma) < s_{j^*}$, we can let

$$t_j(\sigma) = \max\big(T(\sigma) \cap s_j\big),$$

that is, $t_j(\sigma)$ is the last stage in $T(\sigma)$ prior to stage s_j. For $j \geqslant j^*$ we then let

$$o_j(x) = \alpha \cdot \left(\bigoplus_{\sigma \in a} \zeta_{t_j(\sigma)}(\sigma)\right) + \theta_j.$$

Since $\beta\alpha\omega$ is closed under addition, $\bigoplus_{\sigma\in a_j} \zeta_{t_j(\sigma)}(\sigma) < \beta\alpha\omega$, and as $\theta_j < \alpha$, $o_j(x) < \alpha\beta\alpha\omega$.

We need to show that for all $j \geqslant j^*$, $o_j(x) \geqslant o_{j+1}(x)$, and if $\Phi_d(A, D, x)[s_j] \neq \Phi_d(A, D, x)[s_{j+1}]$ then $o_j(x) > o_{j+1}(x)$.

Let $j \geqslant j^*$. If $s_j \in T(\sigma)$ for some $\sigma \in a_j$ then $t_j(\sigma) < t_{j+1}(\sigma)$, and so $\zeta_{t_j(\sigma)}(\sigma) > \zeta_{t_{j+1}(\sigma)}(\sigma)$; as $\theta_j < \alpha$, it follows that $o_j(x) > o_{j+1}(x)$.

Suppose that $s_j \notin \bigcup_{\sigma\in a} T(\sigma)$. For all $\sigma \in a$, $\zeta_{t_j(\sigma)}(\sigma) \geqslant \zeta_{t_{j+1}(\sigma)}(\sigma)$; and $\theta_j \geqslant \theta_{j+1}$, so together we get $o_j(x) \geqslant o_{j+1}(x)$. If $\Phi_d(A, D, x)[s_j] \neq \Phi_d(A, D, x)[s_{j+1}]$ then Claim 2.15.5 says that $\theta_j > \theta_{j+1}$, which implies that $o_j(x) > o_{j+1}(x)$.

This concludes the proof of Lemma 2.15, and so of Theorem 1.4. □

2.1. **Powers of α.** We could modify the previous construction: instead of diagonalising against all β-c.a. functions we can diagonalise against all functions which are γ-c.a. for some $\gamma < \beta$. Of course if β is a successor power of ω then we get nothing new. Suppose then that β is a limit of powers of ω. We then get $\deg_{\mathrm{T}}(A \oplus D)$ to be not totally $(<\beta)$-c.a., which is actually stronger than not totally γ-c.a. for all $\gamma < \alpha$ (see [DG20, Sec.3.4]). Further, every $f \leqslant_{\mathrm{T}} A \oplus D$ has an approximation where on each input we start with an ordinal bounded below $\alpha\gamma\alpha\omega$ for some $\gamma < \alpha$. Suppose further that β is a limit of power of α. Then $\alpha\gamma\alpha\omega < \beta$. In this case we have constructed a properly totally β-c.a. degree. So for example, above every totally ω-c.a. degree there is one which is properly totally ω^ω-c.a., $\omega^{\omega\cdot 2}$-c.a., and so on. This is a special case of Theorem 1.7, and as we discussed, gives a little bit more in the cases it applies (being not totally $(<\beta)$-c.a.). It would be interesting to find a direct construction proving all cases of Theorem 1.7.

3. Existence of maximal degrees in upper cones

In this section we prove Theorem 1.7. Indeed we prove the slightly more general theorem:

Theorem 3.1. *Let $\alpha \leqslant \beta < \epsilon_0$. Above every totally α-c.a. degree there is a degree $\mathbf{d}$ which is totally $(\alpha\beta)$-c.a., but such that no degree strictly above $\mathbf{d}$ is totally β-c.a.*

Theorem 1.7 follows since if $\beta \geqslant \alpha^\omega$ then $\alpha\beta = \beta$.

Let α, β be as in the theorem, and let A be a c.e. set of totally α-c.a. degree. We proceed to enumerate a c.e. set D with the intent that $\mathbf{d} = \deg_{\mathrm{T}}(A \oplus D)$ satisfies the conclusion of the theorem. As well as building on the previous construction, the proof utilises some ideas of the proof of Theorem 4.1 from [DG20].

Requirements. Let $\langle \Phi_d \rangle_{d<\omega}$ be an enumeration of all functionals. To ensure that $\deg_{\mathrm{T}}(A \oplus D)$ is totally $(\alpha\beta)$-c.a., we must meet the set of requirements given by:

$$\text{for all } d < \omega,\ Q_d : \text{If } \Phi_d(A, D) \text{ is total, then it is } (\alpha\beta)\text{-c.a.}$$

For the maximality property of $\deg_{\mathrm{T}}(A \oplus D)$, we need to ensure that for all $e < \omega$, either $W_e \leqslant_{\mathrm{T}} A \oplus D$ or $\deg_{\mathrm{T}}(A \oplus D \oplus W_e)$ is not totally β-c.a. To this end, we enumerate a Turing functional Λ_e with the intent that either $W_e \leqslant_{\mathrm{T}} A \oplus D$, or $\Lambda_e(D, W_e)$ is not β-c.a. Again we do not have use for the oracle A. We meet the following set of requirements:

$$\text{for all } e, i < \omega,\ P_e^i : \text{If } \left\langle f_s^{\beta,i}, o_s^{\beta,i} \right\rangle \text{ is eventually } \beta\text{-computable, then}$$
$$\Lambda_e(D, W_e) \neq f^{\beta,i}, \text{ or } W_e \leqslant_{\mathrm{T}} A \oplus D.$$

Discussion. As usual, we work with a tree of strategies. Nodes τ working for Q_d behave in exactly the same way as in the previous construction. The difference is the analysis of the ordinal number of D-changes that may injure a certified computation; we need to get that below β. This analysis is made easier than in the previous proof, because the extra oracle W_e allows us to lift uses $\lambda_e(p)$ beyond the use $\varphi_d(x)$ of a certified computation. On the other hand, in order to compute W_e, we will need to appoint a possibly infinite sequence of followers for every σ, and σ may act positively infinitely often.

Let us discuss this in greater detail. Suppose that σ appoints a follower p with the aim of making $\Lambda_e(D, W_e, p) \neq f^{\beta,i}(p)$. A new restriction is that we are not allowed any action while $o_s^{\beta,i}(p) = \beta$; we need to provide to τ above us an ordinal strictly smaller than β. This creates a timing difficulty: σ firsts appoints p; then τ certifies $\Phi_d(A, D, x)$; only later do we see $o_s^{\beta,i}(p) < \beta$. We need to protect $\Phi_d(x)$ from p. To do this, we first wait for a W_e-change that allows us to lift $\lambda_e(p)$. Once this has happened, we can attack with p. If we never see such a change, then we made a step toward computing W_e.

It seems though that we nonetheless run into the same problem as in the previous construction: after we lift the use we get an A-change below $\varphi_d(x)$ for a protected computation. Even if we allowed A as an extra oracle for Λ_e, we would still need to immediately redefine the use $\lambda_e(p)$, and the new $\varphi_d(x)$ use which is revealed only later is greater. Recall however that we are not trying to show that W_e is computable, merely computable from $A \oplus D$. Thus, when we get such an A-change, we again cancel the follower p. In this construction, this allows us to make progress on computing W_e from $A \oplus D$.

Now unlike the previous construction, we do not want the α-ordinal count tracked by some $\bar{\tau}$ to force follower cancellations; we are trying to keep the ordinals tracking D-changes below β, not below $\beta\alpha\omega$. Thus, which $\Phi_d(x)$ computations are protected must be determined by follower rather being the same for all followers for σ. That is, the later a follower is appointed, the more computations $\Phi_d(x)$ it needs to protect. There will be no last follower appointed; so we could have infinitely many cycles in which σ starts acting for p, enumerating numbers into D, and then p gets cancelled.

We will need to argue that τ lying *below* σ will not by injured infinitely often by σ. Such a node τ will guess that σ acts infinitely often, and it works in σ's "wake": it waits for a cycle of attack to finish, and then certifies some computations; the next attack cycle will start with lifting the use to a much larger number.

Strategy tree. A node τ working for requirement Q_d is structured exactly as in the previous construction, with descendants being $\tau\hat{}\mathrm{f}$, $\tau\hat{}\infty$, $\tau\hat{}\infty\hat{}\mathrm{f}_y$, $\tau\hat{}\infty\hat{}\infty_n$, and $\tau\hat{}\infty\hat{}\infty_n\hat{}k$. As in the previous construction, since A is low$_2$, we have a computable list $l_s(\tau, n)$ of sequences, non-decreasing in s, such that $\hat{\Phi}_\tau(A)$ is total if and only if the sequence $\langle l_s(\tau, n)\rangle_{s<\omega}$ is unbounded for some n.

A node σ working for requirement P^i_e has outcomes ∞ and f, with $\infty <$ f, which guess whether σ will act infinitely or finitely often (respectively). Both children of σ work for the next (lower priority) requirement.

The node σ is responsible for the enumeration of an enumeration functional Δ_σ, with the aim of having either $\Lambda_e(D, W_e) \neq f^{\beta,i}$, or $\Delta_\sigma(A, D) = W_e^{\complement}$ (the complement of W_e).

During the construction, the node σ will likely appoint several followers. Each follower p is connected with a particular potential element k of W_e; we write $p = p(\sigma, k)$. Unless σ is initialised, at most one follower is appointed for each k, and so the value $p(\sigma, k)$ does not depend on the stage. Nor does the index of a particular follower ever change.

A follower p for σ may become *realised* at some σ-stage. This will be a σ-stage at which we see $o^{\beta,i}_s(p) < \beta$. At a later stage a realised follower may become *permitted.* An unrealised follower, or a permitted follower, can later be cancelled; a realised but unpermitted follower will only be cancelled if σ is initialised.

A follower $p = p(\sigma, k)$ will be permitted at a stage s if $k \in W_{e,s} \setminus W_{e,s-1}$. This may not be a σ-stage. To keep $\Lambda_e(D, W_e)$ total, it is important that we redefine the lifted use $\lambda_{e,s+1}(p)$ immediately. For that reason, the proof of Lemma 2.4 doesn't apply in this construction; it relies on the use being lifted only during σ-stages. We therefore need to actively protect the computations that were covered by that lemma.

To this end, we let $\mathtt{prec}(\sigma)$ be the set of nodes τ which work for some requirement Q_d such that $\tau\hat{}\infty \prec \sigma$. We let $\tau \in \mathtt{prec}_\infty(\sigma)$ if $\tau\hat{}\infty\hat{}\infty_n \prec \sigma$ for some n; for $y < \omega$, we let $\tau \in \mathtt{prec}_y(\sigma)$ if $\tau\hat{}\infty\hat{}\mathrm{f}_y \preccurlyeq \sigma$. When a follower p is realised, we define numbers $m^\tau(p)$ for all $\tau \in \mathtt{prec}_\infty(\sigma)$ (which are not changed later). We then let $\mathtt{Pr}(p)$ be the set of pairs (d, x) such that for some $\tau \in \mathtt{prec}(\sigma)$ working for Q_d, either

- $\tau \in \mathtt{prec}_\infty(\sigma)$, and $x < m^\tau(p)$; or
- $\tau \in \mathtt{prec}_y(\sigma)$, and $x < y$.

Again, the idea is that for $(d, x) \in \mathtt{Pr}(p)$, the computation $\Phi_d(A, D, x)$ should be protected from actions for p.

The functionals Λ_e take as oracles both D and W_e. We will give each oracle a separate use: each oracle is queried on a single number. We denote the D-use of a computation $\Lambda(D, W_e, p)[s]$ by $\lambda_{e,s}(p)$. If $p = p(\sigma, k)$ for some σ working for P^i_e, then the W_e-use of the computation is k; that is, such a computation can be removed if k later enters W_e. Otherwise, no W_e-changes can affect the computation (including when p is not currently a follower for such σ).

Construction. Let s be a stage. First, at s, we observe the effects of enumerations into W_e. Let σ be a node which works for requirement P^i_e; and suppose that $k \in W_{e,s} \smallsetminus W_{e,s-1}$. Suppose that $p = p(\sigma, k)$ is currently defined.

(1) If p is currently unrealised, we cancel it.
(2) If p is realised, then we redefine $\Lambda_{e,s+1}(D_{s+1}, W_{e,s}, p)$ with large value and large D-use $\lambda_{e,s+1}(p)$ (and no W_e-use); we will ensure below that $\Lambda_e(D, W_e, p)\uparrow [s]$. Henceforth p is *permitted*.

If such action occurred, we end the stage.

If the stage was not terminated, we proceed to define the collection γ_s of accessible nodes, and act according to the instructions below. The first accessible node is the empty node.

Let $\tau \in \gamma_s$ work for requirement Q_d. The instructions for τ, and the designation of the next accessible nodes, are precisely as in the previous construction.

Let $\sigma \in \gamma_s$ work for requirement P^i_e. There are several possible circumstances for σ which require individual attention.

(i) σ has a permitted follower. There may be more than one permitted follower. For each permitted follower $p = p(\sigma, k)$:
   (a) If $k \notin \Delta_\sigma(A, D)[s]$, then we cancel the follower p.
   (b) If $k \in \Delta_\sigma(A, D)[s]$ and $\Lambda_e(D, W_e, p)[s] = f^{\beta,i}_s(p)$, then we enumerate $\lambda_{e,s}(p)$ into D_{s+1}, and redefine $\Lambda_{e,s+1}(D_{s+1}, W_{e,s}, p)$ to be large, with large D-use $\lambda_{e,s+1}(p)$ (and no W_e-use). For all $\tau \in \mathtt{prec}_\infty(\sigma)$, for all $x \geqslant m^\tau(p)$, we cancel the tracker $\mathtt{tr}_s(\tau, x)$ if it is defined.

   Now we decide which if any child of σ is next accessible:
   (1) If $\lambda_{e,s}(p)$ was enumerated into D_{s+1} for some follower p, then we initialise all nodes to the right of $\sigma\hat{}\infty$, including $\sigma\hat{}$f. We do *not* initialise nodes extending $\sigma\hat{}\infty$. We end the stage s.
   (2) Otherwise, if all permitted followers were just cancelled, then we let $\sigma\hat{}\infty$ be next accessible.
   (3) Otherwise, we let $\sigma\hat{}$f be next accessible.
(ii) σ has no permitted followers, but does have a yet unrealised follower $p = p(\sigma, k)$. There will be at most one such follower.
   (a) If $o^{\beta,i}_s(p) = \beta$ then we let $\sigma\hat{}$f be next accessible.
   (b) If $o^{\beta,i}_s(p) < \beta$ then we let, for all $\tau \in \mathtt{prec}_\infty(\sigma)$, $m^\tau(p) = \operatorname{dom}\Phi_d(A, D)[s]$, where τ works for Q_d. The follower p is henceforth *realised*. We let $\sigma\hat{}$f be next accessible.
(iii) Suppose that σ has no followers, or that all followers for σ are already realised, but none are permitted.

   We maintain Δ_σ. For each k such that $p(\sigma, k)$ is defined, if $k \notin \Delta_\sigma(A, D)[s]$, then we let $k \in \Delta_{\sigma,s+1}(A_s, D_s)$ with use

$$\delta_{\sigma,s+1}(k) = \max\left\{\varphi_{d,s}(x) \,:\, (d, x) \in \mathtt{Pr}(p)\right\}.$$

   We will observe below that indeed for each $(d, x) \in \mathtt{Pr}(p)$, $\Phi_d(A, D, x)\downarrow [s]$.

   We then let k be the least element of $W^{\mathtt{C}}_{e,s}$ such that $p(\sigma, k)$ is yet undefined. We pick a new, large follower p and let $p = p(\sigma, k)$. We define $\Lambda_{e,s+1}(D_s, W_{e,s}, p)$ to be large with large D-use and W_e-use k. We initialise all nodes to the right of σ (but not nodes extending $\sigma\hat{}$f). We end the stage.

Note that we eventually run into a long node σ with no followers, so each stage has only finitely many steps.

At the end of every stage, we maintain the functionals Λ_e as in the previous construction: for any pair $(e,p) \leqslant s$ such that p is not currently a follower for any node σ working for P^i_e for some $i < \omega$, if $p \notin \operatorname{dom} \Lambda_e(D, W_e)[s]$ then we define a permanent computation $\Lambda_e(D, W_e, p)$ with use -1.

Verification. Before we begin our verification, we need to show that the construction can run smoothly, in that its instructions can be carried out.

Lemma 3.2. *Let σ be a node working for requirement P^i_e.*

(1) If $p = p(\sigma, k)$ is a realised follower for σ at the start of stage s, and $k \in W_{e,s} \smallsetminus W_{e,s-1}$, then $\Lambda_e(D, W_e, p)\uparrow [s]$.

(2) If p is a follower for σ at a σ-stage s, and p is realised by the end of the stage, then for all $(d,x) \in \mathtt{Pr}(p)$, $\Phi_d(A, D, x)\downarrow [s]$.

(3) Between stages at which σ is initialised, for all k, σ appoints at most one follower $p(\sigma, k)$.

Proof. For (1), let $t < s$ be the stage at which p is appointed. At that stage we define a computation $\Lambda_e(D, W_e, p)$ with some D-use $\lambda_{e,t+1}(p)$ and W_e-use k. We never act for p before it is permitted, and so this computation is preserved until stage s, at which k's entry into W_e invalidates it.

For (2), Let $t \leqslant s$ be the σ-stage at which p is first realised (and $m^\tau(p)$ and hence $\mathtt{Pr}(p)$ defined). Let $(d,x) \in \mathtt{Pr}(p)$; let $\tau \in \mathtt{prec}(\sigma)$ work for Q_d. Suppose that $\tau \in \mathtt{prec}_y(\sigma)$; so $y > x$. Since s is a $\sigma\hat{\ }\infty\hat{\ }\mathrm{f}_y$-stage, the instructions imply that $y \geqslant \operatorname{dom} \Phi_d(A, D)[s]$. Suppose that $\tau \in \mathtt{prec}_\infty(\sigma)$; so $x < m^\tau(p)$. If $t = s$ then we define $m^\tau(p) = \operatorname{dom} \Phi_d(A, D)[s]$. If $s > t$ then as s is a $\tau\hat{\ }\infty$-stage, $\operatorname{dom} \Phi_d(A, D)[s] > t > m^\tau(p)$.

For (3), suppose that at stage t, a follower $p = p(\sigma, k)$ is appointed. Before σ is initialised, p will only be cancelled after k enters W_e (either because this happens while p is still unrealised, or after p is permitted.) Later, no new follower $p(\sigma, k)$ will ever be appointed. $\square$

Lemma 3.3. *For any node σ working for P^i_e, if $p = p(\sigma, k)$ is a follower for σ at the beginning of stage s, and it is not the case that $k \in W_{e,s} \smallsetminus W_{e,s-1}$, then $\Lambda_e(D, W_e, p)\downarrow [s]$.*

Proof. A computation is defined with W_e-use k (where $p = p(\sigma, k)$) at the stage $t < s$ at which p is first appointed. Whenever σ itself destroys the computation by enumerating its D-use into D, a new computation is immediately set up. If k enters W_e, we either immediately cancel p (if it is unrealised), or permit p, at which stage a new computation with no W_e-use is defined. $\square$

Our next task is to show that protection works: action for a follower p cannot injure a computation $\Phi_d(A, D, x)$ for any $(d,x) \in \mathtt{Pr}(p)$.

Lemma 3.4. *Let p be a follower for a node σ working for requirement P^i_e, and suppose that $\lambda_{e,s}(p) \in D_{s+1} \smallsetminus D_s$. Then for all $(d,x) \in \mathtt{Pr}(p)$, $\varphi_{d,s}(x) < \lambda_{e,s}(p)$.*

Proof. The stage s is a σ-stage. The follower $p = p(\sigma, k)$ received permission at some stage $t < s$. At stage s, $k \in \Delta_\sigma(A, D)[s]$, as otherwise we would cancel p rather than act for it. Let $r < s$ be the stage at which this enumeration was defined. Then $r < t$, as between t and s, no new $\Delta_\sigma(A, D)$-enumerations are defined.

As σ was not initialised between stages r and s, by stage r, p was already chosen as a follower and realised. At stage r we define $\delta_{\sigma,s}(k) = \delta_{\sigma,r+1}(k) = \max\{\varphi_{d,r}(x) : (d, x) \in \mathtt{Pr}(p)\}$. By our choice of r, we have $A_r \restriction \delta_{\sigma,s}(k) = A_s \restriction \delta_{\sigma,s}(k)$, and the same holds for D. Hence for all $(d, x) \in \mathtt{Pr}(p)$, $\varphi_{d,r}(x) = \varphi_{d,s}(x)$ by the same computation. At stage t we lift $\lambda_{e,t+1}(p)$ to be large, and so $\lambda_{e,s}(p) > t > \varphi_{d,r}(x)$ (recall $r < t$). $\square$

As in the previous proof, we need to be sure that if a computation is destroyed by a change in D, its tracker is cancelled immediately, to allow us to correctly anticipate further A-changes. Note that the following lemma implies that though we may have infinite positive action by a node on the true path, this action will not injure weaker nodes on the true path: they simply wait for the action and confirm only relatively small numbers.

Lemma 3.5. *Let τ be a node working for requirement Q_d. Let s be a stage; let $x < \omega$ be such that $c = \mathtt{tr}_s(\tau, x)$ is defined. Suppose that $\hat{\Phi}_\tau(A, c)\downarrow [s]$, and let $u = \hat{\varphi}_{\tau,s}(c)$. Then:*

(i) $\Phi_d(A, D, x)\downarrow [s]$ and $u = \varphi_{d,s}(x)$.
(ii) If $D_s \restriction u \neq D_{s+1} \restriction u$ then the tracker c is cancelled at stage s.

Proof. Suppose that (i) and (ii) hold up to stage s, and that the lemma's hypotheses hold at s. The proof of part (i) is identical to the corresponding part of Lemma 2.5. Again let $t < s$ be the stage at which the computation $\hat{\Phi}_\tau(A, c)[s]$ was defined.

For (ii), suppose that $D_s \restriction u \neq D_{s+1} \restriction u$; then at stage s we enumerate a number $\lambda_{e,s}(p) < u$ into D, where p is a follower for a node σ, which works for requirement P^i_e.

Since $\lambda_{e,s}(p) < u = \varphi_{d,t}(x)$, we know that the computation $\Lambda_e(D, W_e, p)[s]$ was defined prior to stage t; so p was permitted prior to stage t.

As before we ask: how does τ relate to σ?

If σ lies to the right of $\tau\hat{\ }\infty$, then σ is initialised at stage t; this is impossible. If τ lies to the right of $\sigma\hat{\ }\infty$, then τ is initialised at stage s; in particular, c is cancelled at s.

There are two possibilities left: $\sigma\hat{\ }\infty \preccurlyeq \tau$, and $\tau\hat{\ }\infty \prec \sigma$.

Suppose that $\sigma\hat{\ }\infty \preccurlyeq \tau$. Since t is a τ-stage, it is also a $\sigma\hat{\ }\infty$-stage. But at the end of every $\sigma\hat{\ }\infty$-stage, σ has no permitted followers, contradicting p being such a follower.

Hence $\tau\hat{\ }\infty \prec \sigma$, i.e., $\tau \in \mathtt{prec}(\sigma)$. By Lemma 3.4, $(d, x) \notin \mathtt{Pr}(p)$.

Suppose that $\tau \in \mathtt{prec}_y(\sigma)$. Then $x \geqslant y$. Since s is a $\tau\hat{\ }\infty\hat{\ }\mathrm{f}_y$-stage, the computation $\Phi_d(A, D, x)[t]$ is destroyed by stage s; this is not the case. Hence $\tau \in \mathtt{prec}_\infty(\sigma)$; and as $x \geqslant m^\tau(p)$, at stage s, σ is instructed to cancel c. $\square$

We obtain the analogue of Corollary 2.6:

Corollary 3.6. *Let τ be a node working for requirement Q_d. Let $x < \omega$, and suppose that at some point a tracker $c = \mathtt{tr}(\tau, x)$ is defined and is never cancelled. Suppose that $\hat{\Phi}_\tau(A, c)\downarrow$. Then $\Phi_d(A, D, x)\downarrow$.*

We also obtain the analogue of Lemma 2.7, with the same proof:

Lemma 3.7. *Let σ be a node which works for P^i_e, and let p be a follower for σ. There are only finitely many stages s at which σ acts on p's behalf by enumerating $\lambda_{e,s}(p)$ into D_{s+1}.*

Let the true path, γ_ω, consist of the nodes μ such that:

(i) $\mu \in \gamma_s$ for infinitely many s; and
(ii) μ is initialised only finitely often.

Note that unlike the previous construction, in this construction nodes sometimes get initialised even if they don't lie to the right of γ_s, namely when a node σ acts by enumeration into D. This needs to be taken into consideration when proving some node lies on the true path.

Lemma 3.8. *Suppose that τ works for Q_d and that $\tau\hat{\ }\infty \in \gamma_\omega$.*

(1) Every $x < \omega$ is eventually appointed a tracker which is never cancelled.
(2) If $\Phi_d(A, D)$ is total then so is $\hat{\Phi}_\tau(A)$, and for some $n < \omega$, $\tau\hat{\ }\infty\hat{\ }\infty_n \in \gamma_\omega$.
(3) If $\Phi_d(A, D)$ is partial then so is $\hat{\Phi}_\tau(A)$, and $\tau\hat{\ }\infty\hat{\ }\mathrm{f}_y \in \gamma_\omega$, where $y = \operatorname{dom}\Phi_d(A, D)$.

Proof. The proof is almost identical to the proof of Lemma 2.11. For (1), we need to observe that for all $x < \omega$, there are only finitely many followers p ever appointed such that $m^\tau(p) \leqslant x$; again this happens because $\operatorname{dom}\Phi_d(A, D)[s]$ is strictly increasing on the $\tau\hat{\ }\infty$-stages; now Lemma 3.7 suffices, we have no need for something like Lemma 2.10.

For (2), we only need to add that if $\tau\hat{\ }\infty\hat{\ }\infty_n$ is initialised at stage s, but this node does not lie to the right of γ_s, then τ is initialised at stage s. The same holds for $\tau\hat{\ }\infty\hat{\ }\mathrm{f}_y$ for (3). □

Corollary 3.9. *Let $\sigma \in \gamma_\omega$ work for P^i_e; let $p = p(\sigma, k)$ be a realised follower for σ, appointed after the last stage at which σ is initialised. For every $(d, x) \in \mathtt{Pr}(p)$, $\Phi_d(A, D, x)\downarrow$. Hence, the set $\{\delta_{\sigma,s}(k) : s < \omega$ is such that $k \in \Delta_\sigma(A, D)[s]\}$ is bounded.*

Proof. Let $(d, x) \in \mathtt{Pr}(p)$; let $\tau \in \mathtt{prec}(\sigma)$ which works for Q_d. If $\tau \in \mathtt{prec}_\infty(\sigma)$ then $\Phi_d(A, D)$ is total. If $\tau \in \mathtt{prec}_y(\sigma)$ then $x < y = \operatorname{dom}\Phi_d(A, D)$. The second part follows from the definition of $\delta_{\sigma,s+1}(k)$. □

Lemma 3.10. *Let $\sigma \in \gamma_\omega$ work for requirement P^i_e. Then either $\sigma\hat{\ }\infty \in \gamma_\omega$ or $\sigma\hat{\ }\mathrm{f} \in \gamma_\omega$.*

Proof. If $\sigma\hat{\ }\infty$ is accessible infinitely often, then it lies on the true path. Suppose, then, that this doesn't happen; let s^* be a stage after which σ is never initialised, and $\sigma\hat{\ }\infty$ never accessible.

There are a couple of possibilities. If there is a stage $t > s^*$ such that at t, some follower for p is permitted, then after stage t, no new followers will ever be appointed. By Lemma 3.7, there will be only finitely many stages after stage t at which σ is the last accessible stage and $\sigma\hat{\ }\mathrm{f}$ is initialised. In all other σ stages after stage t, $\sigma\hat{\ }\mathrm{f}$ is accessible. Hence $\sigma\hat{\ }\mathrm{f} \in \gamma_\omega$.

Otherwise, $\sigma\hat{\ }\mathrm{f}$ is not initialised after stage s^*. We may or may not appoint infinitely many followers for σ. However even if we do, if $t > s^*$ is a σ-stage at which we appoint a follower and end the stage, then at the next σ-stage, $\sigma\hat{\ }\mathrm{f}$ is accessible. Hence in this case too, $\sigma\hat{\ }\mathrm{f} \in \gamma_\omega$. □

Lemma 3.11. *The true path γ_ω is infinite.*

Proof. First we need to show that the empty node lies on the true path. This amounts to showing that it is impossible that at all but finitely many stages, some realised follower gets permitted or an unrealised follower cancelled. But in that case, there are only finitely many followers ever appointed, and each such follower is realised or cancelled at most once.

We the observe that every node on the true path has a child on the true path. Nodes τ working for Q_d and their derivative nodes are dealt with precisely as in the proof of Lemma 2.12, using Lemma 3.8. Nodes σ working for some P^i_e are dealt with in Lemma 3.10. ☐

Lemma 3.12. *For all e, $\Lambda_e(D, W_e)$ is total.*

Proof. Like the proof of Lemma 2.13, using Lemmas 3.3 and 3.7. ☐

Lemma 3.13. *Every requirement P^i_e is met.*

Proof. There are three possibilities.

First, suppose that some follower p for σ is permitted and never cancelled. The argument of Lemma 2.14 shows that $\Lambda_e(D, W_e)(p) \neq f^{\beta,i}(p)$, and so the requirement is met.

Second, suppose that some follower p for σ is appointed, never cancelled, and never realised. Then $o^{\beta,i}_s(p) = \beta$ for all s. In this case the requirement P^i_e is met vacuously: the approximation $\langle f^{\beta,i}_s, o^{\beta,i}_s\rangle$ is not eventually β-computable.

The last possibility is that every follower is eventually cancelled or realised; but every permitted follower is later cancelled. In this case we show that $\Delta_\sigma(A, D) = W^{\complement}_e$.

For one direction, we use:

Claim 3.13.1*.* Suppose that s is a σ-stage and suppose that at stage s, σ has no permitted followers. Then $\Delta_\sigma(A, D)[s] \subseteq W^{\complement}_{e,s}$.

Proof. Let $k \in \Delta_\sigma(A, D)[s]$. Let r be the σ-stage at which the enumeration was defined. At stage r, $p = p(\sigma, k)$ is defined, realised but not permitted, and so $k \notin W_{e,r}$. Suppose that k is enumerated into $W_{e,t}$ at some stage $t \in (r, s]$. Then p becomes permitted at stage t, and so $t < s$. By assumption, p is cancelled prior to stage s. It is cancelled (at stage $w \in (t, s)$) because $k \notin \Delta_\sigma(A, D)[w]$, contrary to our choice of r. ☐

Now we note that there are infinitely many σ-stages s at which σ has no permitted follower: suppose that t is a σ-stage at which σ has some permitted followers. While σ has some permitted followers, no new followers are appointed. Each permitted follower is eventually cancelled. Hence, even if more followers are permitted (while some permitted followers from stage t are still around), eventually there will be a $\sigma\hat{\ }\infty$-stage. The next σ-stage after that is as required. Overall, we conclude that $\Delta_\sigma(A, D) \subseteq W^{\complement}_e$.

In the other direction, we observe that there are infinitely many σ-stages s at which option (iii) is taken: Δ_σ is maintained and a new follower is appointed. If r is such a stage (after the last stage at which σ is initialised), then at some the follower p appointed at stage r is either cancelled or realised at some stage $t > r$. Either the next σ stage after stage t is as required; or some follower for σ is permitted before that next σ-stage. We then undergo an "attack cycle"; as above, the next σ-stage after the following $\sigma\hat{\ }\infty$-stage is as required.

By induction on $k \in W_e^{\complement}$, we show that at some stage s we appoint a follower $p = p(\sigma, k)$. Since $k \notin W_e$, this follower is never cancelled. Hence it is never permitted; but by assumption, it is at some point realised. At infinitely many σ-stages s, if $k \notin \Delta_\sigma(A, D)[s]$, then we define a new enumeration at stage s. By Corollary 3.9, all of these enumerations' uses are bounded, and so one must be (A, D)-correct. Hence $k \in \Delta_\sigma(A, D)$. □

The following will conclude the proof of Theorem 3.1.

Lemma 3.14. *For all $d < \omega$, the requirement Q_d is met.*

Proof. The proof is similar to the proof of Lemma 2.15, but simpler, because we do not need to count the (ordinal) number of possible followers for a node σ.

Let $d < \omega$, and suppose that $\Phi_d(A, D)$ is total. Let τ be the node on the true path working for Q_d. Then $\tau\hat{\ }\infty \in \gamma_\omega$; by Lemmas 3.8 and 3.11, for some n and $k < \omega$, $\rho = \tau\hat{\ }\infty\hat{\ }\infty_n\hat{\ }k$ lies on the true path. Let $s_0 < s_1 < \cdots$ be the ρ-stages following the last stage at which ρ is initialised. We proceed to define an $(\alpha\beta)$-computable approximation $\langle f_s, o_s\rangle$ for $\Phi_d(A, D)$.

From now on, fix $x < \omega$. Let $j^* = j^*(x)$ be the least $j > 0$ such that $x < s_{j-1}$. For all $j \geqslant j^*$, $\Phi_d(A, D, x)\downarrow [s_j]$. So for all $j \geqslant j^*$ we let $f_j(x) = \Phi_d(A, D, x)[s_j]$. Certainly $\lim_{j\to\infty} f_j(x) = \Phi_d(A, D, x)$. Let $u_j = u_j(x) = \varphi_{d,s_j}(x)$.

Again we measure A-changes and D-changes. The A-changes are measured exactly as in the previous construction: for $j \geqslant j^*$ we define

$$\theta_j = \theta_j(x) = o^{\alpha,k}_{s_j}(c_j),$$

where $c_j = c_j(x) = \mathtt{tr}_{s_j}(\tau, x)$. We again use the following claim, whose proof is identical to the proof of Claim 2.15.1.

Claim 3.14.1. For all $j \geqslant j^*$, $\theta_j < \alpha$. Suppose that $c_j = c_{j+1}$. Then $\theta_j \geqslant \theta_{j+1}$; and if $A_{s_j} \restriction u_j \neq A_{s_{j+1}} \restriction u_j$ then $\theta_j > \theta_{j+1}$.

To measure the D-changes, we let $a = a(x)$ be the collection of followers p for nodes $\sigma \succcurlyeq \rho$ which are already realised by stage s_{j^*} (and not cancelled before that stage). We do not need to define a_j. For each follower $p \in a(x)$ let $T(p)$ be the set of stages t such that either:

- p is realised at stage t; or
- at stage t, p's node σ enumerates $\lambda_{e,t}(p)$ into D_{t+1} (where σ works for P_e^i).

So for all $p \in a(x)$, $\min T(p) < s_{j^*}$. As above, for $t \in T(p)$ we let

$$\epsilon_t(p) = o^{\beta,i}_t(p).$$

Claim 3.14.2. For all $p \in a(x)$, for all $t \in T(p)$, $\epsilon_t(p) < \beta$. If $t_0 < t_1$ are stages in $T(p)$ then $\epsilon_{t_0}(p) > \epsilon_{t_1}(p)$.

Proof. The first part follows from the fact that p is realised at or before stage t. The second part is similar to the proof of Claim 2.15.2. Since $t_1 \neq \min T(p)$, we have $\Lambda_e(D, W_e, p)[t_1] = f^{\beta,i}_{t_1}(p)$, where p's node σ works for P_e^i. The follower p was permitted before stage t_1. Again the point is that $\Lambda_e(D, W_e, p)[t_1] > t_0$, as the t_1-computation is defined no later than stage t_0: either σ acts for p at stage t_0 and defines a new computation then; or $t_0 = \min T(p)$ and p is permitted at some stage between t_0 and t_1, at which a new computation is defined. □

Finally, we need:

Claim 3.14.3. Let $j \geqslant j^*$. Suppose that $\Phi_d(A, D, x)[s_j] \neq \Phi_d(A, D, x)[s_{j+1}]$, and that $s_j \notin \bigcup_{p \in a} T(p)$. Then $\theta_j > \theta_{j+1}$.

Proof. Essentially identical to the proof of Claim 2.15.5. We need to show that if action for a follower p of a node σ destroys the s_j-computation then $p \in a$. The argument is the same: σ must be an extension of ρ, and p must be realised prior to stage s_{j^*}, otherwise $x < m^\tau(p)$. □

The rest is again the same (but simpler). For each $j \geqslant j^*$ and $p \in a$, we let

$$t_j(p) = \max T(p) \cap s_j.$$

We then let

$$o_j(x) = \alpha \cdot \bigoplus_{p \in a} \epsilon_{t_j(p)}(p) + \theta_j.$$

The claims above show this works. □

4. Above a superlow c.e. degree

Recall that a set A is *superlow* if $A' \equiv_{\mathrm{wtt}} \emptyset'$. If A is c.e. then superlowness is equivalent to a stronger property, namely the approximability of the jump *function*. To make sense of approximations, we identify a partial function φ with a total function from ω to $\omega \cup \{\uparrow\}$, where the value $\uparrow$ denotes that the input is not in the domain of φ. Schaeffer [Sch98] showed that a c.e. set is superlow if and only if every A-partial computable function φ is ω-c.a., considered as a total function. That is, there is an ω-computable approximation $\langle f_s \rangle$ of functions from ω to $\omega \cup \{\uparrow\}$ such that $\varphi = \lim_s f_s$. It follows that every c.e., superlow degree is totally ω-c.a. In fact, because there is a universal A-partial computable function, every c.e. superlow set is array computable.

Theorem 4.1. *Let $\beta \leqslant \epsilon_0$ be a power of ω. Every c.e. superlow degree is bounded by a (c.e.) maximal β-c.a. degree (and hence by a properly β-c.a. degree).*

Let A be a superlow c.e. set. We enumerate a set D such that $\deg_{\mathrm{T}}(A \oplus D)$ is maximal β-c.a.

Requirements. The requirements are similar to the ones for the previous construction (Section 2). Let $\langle \Phi_d \rangle_{d<\omega}$ be an enumeration of all Turing functionals. To make $A \oplus D$ totally β-c.a., we need to ensure,

for all $d < \omega$, Q_d: If $\Phi_d(A, D)$ is total, then it is β-c.a.

For the maximality property, we enumerate a Turing functional Λ_e with the intent that either $W_e \leqslant_{\mathrm{T}} A \oplus D$, or $\Lambda_e(D, W_e)$ is not β-c.a. We meet the following set of requirements:

for all $e, i < \omega$, P_e^i : If $\left\langle f_s^{\beta,i}, o_s^{\beta,i} \right\rangle$ is eventually β-computable, then
$\Lambda_e(D, W_e) \neq f^{\beta,i}$, or $W_e \leqslant_{\mathrm{T}} A \oplus D$.

Discussion. The construction is similar to the previous one (proving Theorem 3.1). The extra strength we have is the ability to approximation not only total A-computable functions but also partial ones. How does this help? Now a node τ, working for Q_d, is not required to make $\hat{\Phi}_\tau(A)$ total, even if $\Phi_d(A,D)$ is. What this means is that we are allowed to reserve future trackers for each input x for $\Phi_d(A,D)$.

Consider first the simpler case, $\beta=\omega$. In this case, a node τ, upon first certifying a computation $\Phi_d(A,D,x)$, knows a bound on the number k of times a D-enumeration can injure a computation it certifies. It can reserve a set of k-many trackers and immediately find a bound on the number of times any one of them will record an A-change. Adding these up we get an overall bound and so can give an ω-computable approximation for $\Phi_d(A,D,x)$. If the number of D-changes does not reach the bound k then $\hat{\Phi}_\tau(c)$ will remain undefined for some unused trackers.

In the slightly more complicated case $\beta>\omega$, we present the ordinal bound on D-changes at a given stage t as δ_t+m_t, where δ_t is a limit ordinal (smaller than β) and $m_t<\omega$. If t is the first stage at which we see the current value of δ_t then we reserve a set of m_t-many trackers, and get a bound on the number of A- and D-changes until the limit ordinal δ_t is decreased. When we get a new version of δ_t, we repeat the process with a new set of trackers.

Strategy Tree. A less important yet convenient advantage in the superlow case is the existence of a universal A-partial computable function; this gives us a way to obtain approximations uniformly, so they don't need to be guessed. A node τ on the tree working for Q_d has two outcomes $\infty<\mathrm{f}$, with $\tau\hat{\ }\mathrm{f}$ working for the next, weaker requirement on our ω-list of requirements. The node $\tau\hat{\ }\infty$ has outcomes $\infty<\mathrm{f}_0<\mathrm{f}_1<\mathrm{f}_2<\dots$.

The node τ enumerates a "shadow" functional $\hat{\Phi}_\tau$ as in the previous constructions. By the fixed-point theorem, we know an index for this functional, that is, a column of the jump function J^A which copies $\hat{\Phi}_\tau(A)$. Using a fixed ω-computable approximation for J^A, we obtain, uniformly in τ, an ω-computable approximation $\langle f^\tau_s, o^\tau_s\rangle$ of $\hat{\Phi}_\tau(A)$. We use this not only to track the number of A-changes of a computation $\Phi_d(A,D,x)$ (between two D-changes), but also to replace the low_2 guessing procedure that indirectly gave us a guess about $\operatorname{dom}\Phi_d(A,D)$; lowness allows us to directly ask if $\hat{\Phi}_\tau(A,c)\downarrow$, and as above if c is the last tracker for $y=\operatorname{dom}\Phi_d(A,D)$ then $\hat{\Phi}_\tau(A,c)\uparrow$.

Nodes σ working for P^i_e again have two outcomes $\infty<\mathrm{f}$. We let $\mathtt{prec}(\sigma)$ be the collection of nodes τ working for some Q_d such that $\tau\hat{\ }\infty\preccurlyeq\sigma$. We let $\mathtt{prec}_\infty(\sigma)$ be the collection of nodes τ such that $\tau\hat{\ }\infty\hat{\ }\infty\preccurlyeq\sigma$; for $y<\omega$, we let $\mathtt{prec}_y(\sigma)$ be the collection of nodes τ such that $\tau\hat{\ }\infty\hat{\ }\mathrm{f}_y\preccurlyeq\sigma$. As above, when a follower p for σ is realised we define $m^\tau(p)$ for all $\tau\in\mathtt{prec}_\infty(\sigma)$; this will define $\mathtt{Pr}(p)$ in the same way as in the previous construction.

Construction. The construction is identical to the construction proving Theorem 3.1, except that the behaviour of the τ-nodes is a bit simpler. Every stage s starts with an attempt to permit realised followers of nodes σ, or cancel unrealised followers for such nodes, based on enumerations of numbers into $W_{e,s}$. The instructions are the same as in the previous construction.

Suppose that a node τ, working for requirement Q_d, is accessible at stage s. Let $t<s$ be the last stage before s at which $\tau\hat{\ }\infty$ was accessible, or $t=0$ if there is no

such stage. If $\operatorname{dom}\Phi_d(A,D)[s] \leqslant t$, we let $\tau\hat{}\mathrm{f}$ be next accessible. Otherwise we let $\tau\hat{}\infty$ be next accessible.

Suppose that $\tau\hat{}\infty$ is accessible. We maintain the shadow functional and assign new trackers. For all x such that $c = \mathtt{tr}_s(\tau,x)$ is already defined, if $\hat{\Phi}_\tau(A,c)\uparrow [s]$ then we define a new computation $\hat{\Phi}_{\tau,s+1}(A_s,c)$ with use $\varphi_{d,s}(x)$ (and large value). For each $x < s$ for which $\mathtt{tr}_s(\tau,x)$ is undefined we appoint a new tracker $\mathtt{tr}_{s+1}(\tau,x)$. Unlike the previous constructions, this new tracker is chosen as the next unused element of the column $\omega^{[x]}$ which is reserved for trackers for x.

Next we determine which outcome is accessible. Let r be the previous $\tau\hat{}\infty\hat{}\infty$-stage, $r = 0$ if there is no such stage. If for all $x < r$, $c = \mathtt{tr}_s(\tau,x)$ is already defined and $\hat{\Phi}_\tau(A,c)\downarrow [s] = f^\tau_s(c)$ then we let $\tau\hat{}\infty\hat{}\infty$ be next accessible.

Otherwise, let y be the least number such that $\Phi_d(A,D)\uparrow [t]$, or the computation $\Phi_d(A,D,y)[t]$ was destroyed since stage t. We let $\tau\hat{}\infty\hat{}\mathrm{f}_y$ be next accessible.

The instructions for a node σ working for P^i_e are identical to those given in the previous construction.

At the end of every stage we maintain the functionals Λ_e as above.

Verification. Most of the verification follows that of the previous construction, except that in places it is simpler. We list the lemmas we need. Lemma 3.2 and its proof is copied verbatim, as are the statements and proofs of Lemmas 3.3 to 3.5 and 3.7 and Corollary 3.6.

Lemma 3.8 is different. Indeed, $\hat{\Phi}_\tau(A)$ will be partial. The following is what we need:

Lemma 4.2. *Suppose that τ works for D and that $\tau\hat{}\infty \in \gamma_\omega$.*

(1) Every $x < \omega$ is eventually appointed a tracker which is never cancelled.
(2) If $\Phi_d(A,D)$ is total then $\tau\hat{}\infty\hat{}\infty \in \gamma_\omega$.
(3) If $\Phi_d(A,D)$ is partial then $\tau\hat{}\infty\hat{}\mathrm{f}_y \in \gamma_\omega$, where $y = \operatorname{dom}\Phi_d(A,D)$.

Proof. (1) is proved in the same way as the corresponding part of Lemma 3.8.

(2) is not complicated; for every $r < \omega$, eventually each $x < r$ will be appointed a permanent tracker c, and as $\Phi_d(A,D,x)\downarrow$, eventually a permanent computation $\hat{\Phi}_\tau(A,c)$ will be defined, and eventually we will see it is equal to $f^\tau(c)$.

The interesting part is (3). Suppose that $y = \operatorname{dom}\Phi_d(A,D)$ is finite. For every $x < y$, the outcome f_x is guessed only finitely many times. It remains to show that the outcome ∞ is chosen only finitely many times. Let c be the tracker which is appointed for y and is never cancelled. Again by the analogue of Corollary 3.6, $\hat{\Phi}_\tau(A,c)\uparrow$. This by itself is not sufficient to prove that the outcome ∞ is chosen only finitely many times, as we may see infinitely many false convergences. However, since $\hat{\Phi}_\tau(A,c)\uparrow$, for all but finitely many s, $f^\tau_s(p) = \uparrow$, and this guarantees what we need. □

Corollary 3.9 and Lemmas 3.10 to 3.13 now follow, with the same proofs. All that remains is:

Lemma 4.3. *For all $e < \omega$, the requirement Q_d is met.*

Proof. We start as in the proof of Lemma 3.14. Let τ on the true path work for Q_d, and suppose that $\Phi_d(A,D)$ is total. Let $\rho = \tau\hat{}\infty\hat{}\infty$; ρ is on the true path. Let $s_0 < s_1 < s_2 < \dots$ be the ρ-stages after the last stage at which ρ is initialised. Fix $x < \omega$; let j^* be the least such that $x < s_{j-1}$. For all $j \geqslant j^*$ we let $f_j(x) = \Phi_d(A, D, x)[s_j]$.

Our mechanisms for counting A- and D-changes are the same. We define c_j as in the previous argument; we then let

$$\theta_j = o^\tau_{s_j}(c_j);$$

then $\theta_j < \omega$. We then prove Claim 3.14.1 in the same way, noting that $f^\tau_{s_j}(c_j) = \hat{\Phi}_\tau(A, c_j)[s_j]$ for all j. For the D-changes we define $a = a(x)$ and for $p \in a$, the set $T(p)$ of stages, as above. Similarly we define $\epsilon_t(p)$ for $t \in T(p)$, and prove Claims 3.14.2 and 3.14.3. In fact, we require the following, which is part of the proof of Claim 3.14.3:

Claim 4.3.1. Let $j \geqslant j^*$. If $c_j \neq c_{j+1}$ then:

- $s_j \in \bigcup_{p\in a} T(p)$; and
- c_{j+1} is the successor of c_j in $\omega^{[x]}$.

This means: if $c_j \neq c_{j+1}$ then we know that c_j is cancelled at stage s_j by some $p \in a$; further, c_{j+1} is the next tracker for x appointed at the next $\tau\hat{\ }\infty$-stage after stage s_j — that tracker is not cancelled until possibly stage s_{j+1}. The point again is that nodes accessible between stages s_j and s_{j+1} cannot cancel trackers for x: only nodes extending ρ may do so.

Now define $t_j(p)$ as above; we then write, for each $j \geqslant j^*$,

$$\bigoplus_{p\in a} \epsilon_{t_j(p)}(p) = \delta_j + m_j,$$

where $\delta_j < \beta$ is a limit ordinal (or 0), and $m_j < \omega$.

We let C_j be the set of consecutive elements of $\omega^{[x]}$ starting with c_j and having size $m_j + 1$. That is, C_j consists of the current tracker c_j, and the next m_j-many numbers that may be assigned as future trackers. We let

$$n_j = \sum_{c\in C_j} o^\tau_{s_j}(c).$$

We then let

$$o_j(x) = \delta_j + m_j + n_j.$$

To prove that this works, we first need to show that if $\delta_j = \delta_{j+1}$ then $n_j \geqslant n_{j+1}$. This will follow once we show that $C_j \supseteq C_{j+1}$. Note that under the assumption $\delta_j = \delta_{j+1}$, we have $m_j \geqslant m_{j+1}$. If $c_j = c_{j+1}$ then $C_j \supseteq C_{j+1}$ is clear. If $c_j \neq c_{j+1}$ then $s_j \in \bigcup_{p\in a} T(p)$, and so $\delta_j + m_j > \delta_{j+1} + m_{j+1}$; it follows that $m_j > m_{j+1}$, and then c_{j+1} being c_j's successor in $\omega^{[x]}$ yields $C_j \supseteq C_{j+1}$. Hence for all j, $o_j(x) \geqslant o_{j+1}(x)$.

Suppose that $\Phi_d(A, D, x)[s_j] \neq \Phi_d(A, D, x)[s_{j+1}]$. If $s_j \in \bigcup_{p\in a} T(p)$ and $\delta_j = \delta_{j+1}$ then $m_j > m_{j+1}$ (and $n_j \geqslant n_{j+1}$); so $o_j(x) > o_{j+1}(x)$. Otherwise, $\theta_j > \theta_{j+1}$, and as $c_j \in C_j$, this implies $n_j > n_{j+1}$; in this case too $o_j(x) > o_{j+1}(x)$. $\square$

4.1. **Higher levels of superlowness.** We can generalise superlowness to higher ordinals:

Definition 4.4. Let $\alpha \leqslant \epsilon_0$. A set A is *α-superlow* if A' is α-c.a.

If A is c.e. then A is α-superlow if every A-partial computable function is α-c.a., equivalently, if a universal A-partial computable function (often denoted by J^A) is α-c.a. Every α-superlow (c.e.) degree is totally α-c.a. This is tight:

Proposition 4.5. *Let $\alpha \leqslant \epsilon_0$ be a power of ω. Then there is an α-superlow degree which is not totally γ-c.a. for any $\gamma < \alpha$.*

Sketch of proof. This is a finite injury construction with shifting priorities. We enumerate a c.e. set C and a functional Λ. The requirements to meet are:

For $\gamma < \alpha$ and $i < \omega$, $P^{\gamma,i}$: $\Lambda(C) \neq f^{\gamma,i}$.

and

For all $x < \omega$, N_x: $J^C(x)$ has an α-computable approximation.

A requirement $P^{i,\gamma}$ appoints a follower p, defines $\Lambda(C,p)[s]$ and when it observes that $\Lambda(C,p)[t] = f_t^{\gamma,i}(p)$ then it enumerates $\lambda_t(p)$ into C_{t+1} and defines a new computation (with a new large use).

Let $\langle P_e \rangle$ be an ω-listing of all positive requirements. We start with the priority ordering $P_0 < N_0 < P_1 < N_1 < \dots$.

A requirement N_x observes the action of positive requirements stronger than it. When it sees a new $J^C(x)$ computation, it will cancel the followers of all weaker positive requirements. However, when a positive requirement P_e acts, at some stage t, it not only cancels the followers for all weaker positive requirements, it also demotes them down the ordering: every requirement $P_{e'}$ for $e' > e$ is now declared weaker than N_t.

By induction, we see that every positive requirement P_e is eventually never kicked down the list, and acts only finitely often. Indeed, suppose that stage s is the last stage at which positive requirements $P_{\bar{e}}$ for $\bar{e} < e$ ever acts. Then at stage s the final priority for P_e is determined to be between N_s and N_{s+1}. After stage s, each requirement N_x for $x \leqslant s$ will cancel P_e's follower at most once. This is because no positive requirement stronger than N_x will act after stage s. Thus, every $J^C(x)$ computation observed by N_x after stage s is permanent.

What's left is calculating an ordinal bound on the "number of times" that a computation $J^C(x)$ observed by N_x is injured by the action of a positive requirement stronger than N_x. Let P_e be stronger than N_x at some stage s. Then we know that P_e will not appoint more than x many followers while it is still stronger than N_x. This is the same argument: while P_e is stronger than N_x, no requirement $P_{\bar{e}}$ for $\bar{e} < e$ acts. During that period, for each N_y stronger than P_e, we have $y < x$, and N_y will cancel P_e's follower at most once, as its own $J^C(y)$ computation cannot be injured. Hence we can put an ordinal bound of $\gamma \cdot x$ on the total number of injuries that P_e can cause N_x, where $P_e = P^{\gamma,i}$ for some i. Since α is closed under addition, $\gamma \cdot x < \alpha$. Adding for x many positive requirements (P_e for $e \leqslant x$) still lands us below α. $\square$

However, trying to generalise the proof of Theorem 4.1 to α-superlow degrees instead of superlow degrees does not work. Here the fundamental difference is the one between the number of times and (an ordinal) "number of times". If A is α-superlow then the number n_j in the proof of the last lemma is replaced by some ordinal below α; but we need, at each stage, consider c_j and only finitely many other trackers in the set C_j. There is no ordinal way to add up the ordinals for infinitely many potential trackers. This means that in examining the ordinal number of potential D-changes, we cannot replace the presentation $\delta_j + m_j$ by some $\delta_j + \gamma_j$, where $\gamma_j < \alpha$ and δ_j is an ordinal multiple of α: such γ_j does not tell us in advance a finite bound on the number of tracker needed before we see a change in δ_j. If we stick to the presentation $\delta_j + m_j$, with $m_j < \omega$, then defining $o_j(x) = \delta_j + m_j + \eta_j$ where $\eta_j < \alpha$ (or some other combination of m_j and the ordinals for potential trackers in C_j); but δ_j is not a multiple of α, and then $o_j(x)$ is not necessarily decreasing. Hence we ask:

Question 4.6. *Is every ω^2-superlow degree bounded by a maximal totally ω^2-c.a. degree? Maximal ω^n-c.a. for some or all $n < \omega$?*

5. A maximal interval and a proper ideal

5.1. **A maximal interval.** In this section we prove:

Theorem 5.1. *Let $\alpha \leqslant \epsilon_0$ be a power of ω. There are totally α-c.a. degrees $\mathbf{a} < \mathbf{b}$ such that every totally α-c.a. degree above $\mathbf{a}$ is below $\mathbf{b}$.*

So not only is $\mathbf{b}$ maximal totally α-c.a., it is even maximal over $\mathbf{a}$.

We enumerate c.e. sets A and B with the intention that $\mathbf{a} = \deg_{\mathrm{T}}(A)$ and $\mathbf{b} = \deg_{\mathrm{T}}(A \oplus B)$ are as promised in the theorem.

Requirements. Some of the requirements are familiar. Let $\langle \Phi_d \rangle_{d<\omega}$ be an enumeration of all Turing functionals. To make $A \oplus B$ totally α-c.a., we need to ensure,

$$\text{for all } d < \omega,\ Q_d\text{: If } \Phi_d(A, B) \text{ is total, then it is } \alpha\text{-c.a.}$$

For the maximality property, for each e, we enumerate a Turing functional Λ_e with the intent that either $W_e \leqslant_{\mathrm{T}} A \oplus B$, or $\Lambda_e(A, W_e)$ is not α-c.a. We meet the following set of requirements:

$$\text{for all } e, i < \omega,\ P_e^i : \text{If } \left\langle f_s^{\alpha,i}, o_s^{\alpha,i} \right\rangle \text{ is eventually } \alpha\text{-computable, then}$$
$$\Lambda_e(A, W_e) \neq f^{\alpha,i}, \text{ or } W_e \leqslant_{\mathrm{T}} A \oplus B.$$

Finally, we add Friedberg-Muchnik requirements:

$$\text{for all } c < \omega,\ R_c : \Psi_c(A) \neq B.$$

Here $\langle \Psi_e \rangle$ is an effective list of all Turing functionals; we use different notation to avoid confusion between requirements.

Discussion. This is a mild elaboration on the construction of a maximal totally α-c.a. degree from [DG20]. Essentially, we show that that construction is compatible with the introduction of extra Friedberg-Muchnik requirements. Each such requirement acts at most once, and so it is not difficult to take into account how many injuries of this kind a certified computation $\Phi_d(A, B, x)$ will sustain.

Compared to previous constructions, this construction is simplified by not having to work over a given A.

Strategy Tree. The tree is simplified as a result. A node τ working for Q_d has two outcomes $\infty <$ f. Both outcomes work for the next requirement.

A node σ working for P_e^i will appoint followers $p(\sigma, k)$ as in the previous two constructions. We let $\mathtt{prec}_\infty(\sigma)$ be the set of nodes τ working for some Q_d such that $\tau\hat{\ }\infty \preccurlyeq \sigma$. Followers will be realised, permitted and cancelled as in previous constructions; again we define $m^\tau(p)$ for each $\tau \in \mathtt{prec}_\infty(\sigma)$, and let $\mathtt{Pr}(p)$ be the set of pairs (d, x) such that $x < m^\tau(p)$ for $\tau \in \mathtt{prec}_\infty(\sigma)$ working for Q_d. And as above, the node σ will build an enumeration functional Δ_σ. The behaviour of σ is much simplified; in fact, we barely need the functional Δ_σ, and keep it for consistency of presentation with the previous constructions. Indeed, if we see no reason to cancel a follower when it is permitted, then we will not need to cancel it later, and the attack with this follower will succeed.

Nodes σ working for requirements P^i_e, and nodes ρ working for requirements R_c, have only one outcome. Their action will be finitary.

Construction. Let s be a stage. Let σ be a node which works for requirement P^i_e; and suppose that $k \in W_{e,s} \setminus W_{e,s-1}$. Suppose that $p = p(\sigma, k)$ is currently defined.

(1) If $k \notin \Delta_\sigma(A, B)[s]$ then we cancel p.
(2) Otherwise, we redefine $\Lambda_{e,s+1}(A_{s+1}, W_{e,s}, p)$ with large value and large A-use $\lambda_{e,s+1}(p)$ (and no W_e-use). Henceforth p is *permitted*. We cancel all other followers for σ, and initialise all nodes weaker than σ (including proper extensions of σ). We end the stage.

Note that if more than one node σ wishes as in (2), then we act for the strongest one only, as the others will be immediately initialised.

If the stage was not terminated, we proceed to define the collection γ_s of accessible nodes, and act according to the instructions below. The first accessible node is the empty node.

Let $\tau \in \gamma_s$ work for requirement Q_d. Let t be the last $\tau\hat{\ }\infty$-stage prior to stage s, if there was one; $t = 0$ otherwise. We let $\tau\hat{\ }\infty \in \gamma_s$ if $\operatorname{dom} \Phi_d(A, B)[s] > t$; otherwise $\tau\hat{\ }\mathrm{f} \in \gamma_s$.

Let $\sigma \in \gamma_s$ work for requirement P^i_e. One of the following will hold:

(i) σ has permitted follower p. If $\Lambda_e(A, W_e, p)[s] = f^{\beta,i}_s(p)$, then we enumerate $\lambda_{e,s}(p)$ into A_{s+1}, and redefine $\Lambda_{e,s+1}(A_{s+1}, W_{e,s}, p)$ to be large, with large A-use $\lambda_{e,s+1}(p)$ (and no W_e-use); we then end the stage. Otherwise, we let σ's child be next accessible.
(ii) σ has a yet unrealised follower p. If $o^{\alpha,i}_s(p) < \alpha$ then we let, for all $\tau \in \mathtt{prec}_\infty(\sigma)$, $m^\tau(p) = \operatorname{dom} \Phi_d(A, B)[s]$, where τ works for Q_d. The follower p is henceforth *realised*. In either case, we let σ's child be next accessible.
(iii) σ has no followers, or all followers for σ are already realised, but none are permitted.

    We maintain Δ_σ. For each k such that $p(\sigma, k)$ is defined, if $k \notin \Delta_\sigma(A, B)[s]$, then we let $k \in \Delta_{\sigma,s+1}(A_s, B_s)$ with use
$$\delta_{\sigma,s+1}(k) = \max\{\varphi_{d,s}(x) \,:\, (d, x) \in \mathtt{Pr}(p)\}.$$
    We then let k be the least element of $W^{\mathtt{C}}_{e,s}$ such that $p(\sigma, k)$ is yet undefined. We pick a new, large follower p and let $p = p(\sigma, k)$. We define $\Lambda_{e,s+1}(A_s, W_{e,s}, p)$ to be large with large A-use and W_e-use k. We then let σ's child be next accessible.

Suppose that $\rho \in \gamma_s$ works for requirement R_c.

(i) If ρ has no follower, then we appoint a new, large follower q. We end the stage.
(ii) If ρ has a follower q, $\Psi_c(A, q)\downarrow [s] = 0$, and $q \notin B_s$, then we enumerate q into B_{s+1}, and end the stage.
(iii) Otherwise, we let ρ's child be next accessible.

At the end of the stage we initialise all nodes weaker than the last accessible node (including all proper extensions of that node). We also maintain the functionals Λ_e as usual to ensure their totality.

Verification. The usual lemma that ensures that the construction goes smoothly holds. Lemma 3.2 holds (in (1), of course, replace D by A; in (2), by B); the proofs are simpler. Similarly with Lemmas 3.3, 3.7 and 3.12.

The finitary nature of the construction and the simplicity of the Σ_2/Π_2 behaviour of nodes that work for Q_d show that the true path is infinite, and that every Friedberg-Muchnik requirement R_c is met. Toward showing that the P^i_e requirements are met, we need to prove that $A \oplus B$ is low$_2$.

Lemma 5.2. *Let τ be a node working for Q_d. Let $\bar{t} < t$ be $\tau\hat{\ }\infty$-stages; let $x < \bar{t}$.*

(1) *Let $\rho \succcurlyeq \tau\hat{\ }\infty$ be a node working for R_c. Suppose that at t, ρ has a follower q; and suppose that q was appointed at some stage $r \in (\bar{t}, t)$. Then $\Phi_d(A, B, x)\downarrow [t]$ and $q > \varphi_{d,t}(x)$.*
(2) *Let $\sigma \succcurlyeq \tau\hat{\ }\infty$ be a node working for P^i_e. Suppose that at t, σ has a permitted follower p; and suppose that p became realised at some stage $r \in (\bar{t}, t)$. Then $\Phi_d(A, B, x)\downarrow [t]$ and $\lambda_{e,t}(p) > \varphi_{d,t}(x)$.*

Proof. (1): The stage r is a $\tau\hat{\ }\infty$-stage greater than $\bar{t}$, so $x < \operatorname{dom} \Phi_d(A, B)[r]$; q is chosen large, so $q > \varphi_{d,r}(x)$. We claim that $A_r \restriction \varphi_{d,r}(x) = A_t \restriction \varphi_{d,r}(x)$ and that the same holds for B; it follows that $x < \operatorname{dom} \Phi_d(A, B)[t]$ and that $\varphi_{d,t}(x) = \varphi_{d,r}(x)$. For suppose that at some stage $s \in [r, t)$, a number $u < \varphi_{d,r}(x)$ enters either A or B. This is done by some node μ. Since ρ is not initialised at stage s, μ must be weaker than ρ. Certainly $\mu \neq \rho$ since $q > \varphi_{d,r}(x)$. Hence μ is initialised at stage r. But this means that any number associated with μ at stage s must be greater than r, which is impossible.

(2) is similar, with the complication being that the stage $w \in (r, t)$ at which p became permitted is not a $\tau\hat{\ }\infty$-stage. We do note however that r is a $\tau\hat{\ }\infty$-stage and so that $m^\tau(p) > x$. Since p is not cancelled at stage w, we know that $k \in \Delta_\sigma(A, B)[w]$, where $p = p(\sigma, k)$. Let u be the stage at which this enumeration was defined; so u is a σ-, hence a $\tau\hat{\ }\infty$-stage, and $r \leqslant u < w < t$. At stage u we define $\delta_{\sigma,u}(k) \geqslant \varphi_{d,u}(x)$, and there is no change in either A or B below $\delta_{\sigma,u}(x)$ between stages u and w. At stage w we define $\lambda_{e,w+1}(p)$ to be large, and initialise nodes weaker than σ. Now the argument for (1) works: by considering initialisations, no node μ can enumerate a number smaller than w into either A or B during the interval of stages (w, t), whence $\varphi_{d,t}(x) = \varphi_{d,w}(x) < \lambda_{e,t}(p)$. □

Lemma 5.3. *Let τ be a node on the true path working for Q_d. Then $\tau\hat{\ }\infty$ is on the true path if and only if $\Phi_d(A, B)$ is total.*

Proof. In the non-trivial direction, suppose that $\tau\hat{\ }\infty \in \gamma_\omega$. Let $x < \omega$; let $\bar{t}$ be a $\tau\hat{\ }\infty$-stage such that $\bar{t} > x$. Lemma 5.2 says that only followers appointed or realised at or before stage $\bar{t}$ can ever injure a computation $\Phi_d(A, B, x)[t]$; there are finitely many such followers, and by the analogue of Lemma 3.7, each follower acts finitely many times. □

Lemma 5.4. *For every $i, e < \omega$, the requirement P^i_e is met.*

Proof. The argument is like that for Lemma 3.13, but simpler, as permitted followers cannot be cancelled. Let σ be the node on the true path which works for P^i_e; let s^* be a stage after which σ is never initialised.

First, suppose that some follower p for σ is permitted after stage s^*. Then the usual argument shows that $\Lambda_e(A, W_e, p) \neq f^{\alpha,i}(p)$, so the requirement is met.

Otherwise, suppose that some follower p for σ is never cancelled and never realised; then the requirement is met vacuously.

Finally, if no follower is permitted, but all followers are realised, then $\Delta_\sigma(A, B) = W_e^{\complement}$, with a similar argument to that of Lemma 3.13. The key ingredient is the analogue of Corollary 3.9, which in our case follows from Lemma 5.3. Claim 3.13.1 is easier, since no attacks are cancelled: at every stage s we have $\Delta_\sigma(A, B)[s] \subseteq W_{e,s}^{\complement}$, as otherwise, some follower is permitted. □

Lemma 5.5. *For all $d < \omega$, the requirement Q_d is met.*

Proof. We use the same terminology. Let $s_0 < s_1 < \dots$ be the $\tau\hat{\ }\infty$-stages after the last stage at which τ is initialised, where $\tau \in \gamma_\omega$ works for Q_d. Fix some $x < \omega$. Let j^* be the the least such that $x < s_{j^*-1}$. To track changes in $f_j(x) = \Phi_d(A, B, x)[s_j]$, we define:

- $a(x)$ be the collection of followers p for nodes $\sigma \succcurlyeq \tau\hat{\ }\infty$ working for P^i_e, which have been realised prior to stage s_{j^*};
- $b(x)$ be the collection of followers q for nodes $\rho \succcurlyeq \tau\hat{\ }\infty$ working for R_c, which were appointed prior to stage s_{j^*}.

Lemma 5.2, together with initialisations at $\tau\hat{\ }\infty$-stages, shows that for all $j \geqslant j^*$, if $\Phi_d(A, B, x)[s_j] \neq \Phi_d(A, B, x)[s_{j+1}]$, then action is taken during stage s_j for some follower in either $a(x)$ or $b(x)$. Each follower in $b(x)$ acts at most once; between any two actions for some $p \in a(x)$, the associated ordinal $o^{\alpha,i}(p)$ drops. Hence again, for $p \in a(x)$ we let $T(p)$ be the collection of stages t at which either p is first realised, or $\lambda_{e,t}(p)$ is enumerated into A_{t+1}; and for $j \geqslant j^*$ we let $t_j(p) = \max T(p) \cap s_j$. We then let

$$\delta_j(p) = o^i_{t_j(p)}(p),$$

where p is a follower for a node that works for P^i_e; note that $\delta_j(p) < \alpha$ as p is realised at or after stage $t_j(p)$. Finally we let

$$o_j(x) = \bigoplus_{p \in a(x)} \delta_j(p) + \left| \left\{ q \in b(x) \,:\, q \notin B_{s_j} \right\} \right|.$$

If p acts at stage s_j, then $\delta_{j+1}(p) < \delta_j(p)$; if some follower in $b(x)$ acts at stage s, then the second summand shrinks. □

5.2. **A maximal ideal.** In this subsection we prove Theorem 1.8: for every α there is some totally α-c.a. which is bounded by no maximal totally α-c.a. degree. Paradoxically, the proof is yet another elaboration on the construction of a maximal degree. We expand the construction of the previous subsection to construct a "maximal (proper) ideal":

Theorem 5.6. *Let $\alpha \leqslant \epsilon_0$ be a power of ω. There are totally α-c.a. degrees $\mathbf{d}_0 < \mathbf{d}_1 < \mathbf{d}_2 < \dots$ such that every totally α-c.a. degree above $\mathbf{d}_0$ lies below $\mathbf{d}_n$ for some n.*

To prove the theorem, we enumerate sets A and $B_0, B_1, B_2, \ldots$ with the intention of letting $\mathbf{d}_n = A \oplus B_0 \oplus B_1 \oplus \cdots \oplus B_{n-1}$.

Requirements. Let $\langle \Phi_d \rangle_{d<\omega}$ be an enumeration of all Turing functionals. To make each $\mathbf{d}_n$ totally α-c.a., we need to meet,

for all $d, n < \omega$, Q^n_d: If $\Phi_d(A, B_0, \ldots, B_{n-1})$ is total, then it is α-c.a.

For the maximality property, for each e, we enumerate a Turing functional Λ_e and meet the following set of requirements:

for all $e, i < \omega$, P^i_e : If $\left\langle f^{\alpha,i}_s, o^{\alpha,i}_s \right\rangle$ is eventually α-computable, then $\Lambda_e(A, W_e) \neq f^{\alpha,i}$, or $W_e \leqslant_{\mathrm{T}} A \oplus B_0 \oplus \cdots \oplus B_{n-1}$ for some n.

And the Friedberg-Muchnik requirements:

for all $n, c < \omega$, $R^n_d : \Psi_c(A, B_0, \ldots, B_{n-1}) \neq B_n$.

Again $\langle \Psi_e \rangle$ is an effective list of all Turing functionals.

Discussion. The construction is a very mild modification of the previous construction. The entire thing proceeds rather pleasantly, and without complication. We just need to explain where n comes from, to meet requirement P^i_e: if σ on the true path works for P^i_e, then there are only finitely many nodes $\tau \in \mathtt{prec}_\infty(\sigma)$, and each such τ works for a requirement that mentions only finitely many B_n's; we take all these which are mentioned. Other than that, the construction is identical to the previous one, so we omit the details.

6. Uniformly Totally α-c.a. Degrees

One of the ideas in [DG20] was to further refine the totally ω^α-c.a. hierarchy by exploring an analogue of the fact that array computability was a *uniform* version of being totally ω-c.a. This generalization was done as follows.

Let $\alpha \leqslant \epsilon_0$. We call $h \colon \omega \to \alpha$ an *α-order function* if h is nondecreasing, computable, and its range is unbounded in α. Again, for computability, we take any canonical computable copy of α. An *h-computable approximation* is an α-computable approximation $\langle f_s, o_s \rangle$ such that for all x, $o_0(x) < h(x)$. Just like for α-computable approximations, we can produce an effective list $\left\langle f^{h,i}_s, o^{h,i}_s \right\rangle$ of tidy $(h+1)$-computable approximations whose limits $f^{h,i}$ consists of all h-c.a. functions. Of course every α-c.a. function is h-c.a. for some α-order function h, but when uniform bounds are required for all functions in a degree, we get a stronger notion. And just like the ω-case, the bound does not matter:

Proposition 6.1 ([DG20])**.** *The following are equivalent for a c.e. degree* $\mathbf{d}$*:*

(1) For some α-order function h, all functions $f \in \mathbf{d}$ are h-c.a.
(2) For Every α-order function h, all functions $f \in \mathbf{d}$ are h-c.a.

A degree $\mathbf{d}$ satisfying these conditions is called *uniformly totally α-c.a.* A c.e. degree is uniformly totally ω-c.a. if and only if it is array computable. These new levels refine the hierarchy of totally α-c.a. degrees:

Proposition 6.2 ([DG20])**.** *Let $\alpha \leqslant \epsilon_0$ be a power of ω.*

(1) For every $\beta < \alpha$, every totally β-c.a. degree is uniformly totally α-c.a;
(2) There is a degree which is uniformly totally α-c.a. and not totally β-c.a. for any $\beta < \alpha$;
(3) There is a degree which is totally α-c.a. but not uniformly so.

And again for ordinals which are not powers of ω, we get nothing new: if α is a power of ω and $\gamma \in (\alpha, \alpha \cdot \omega)$, then total uniform γ-c.a.-ness coincides with total γ-c.a.-ness coincides with total α-c.a.-ness.

The last theorem in this paper examines the cone below a c.e. degree which is not totally α-c.a.

Theorem 6.3. *Let $\alpha \leqslant \epsilon_0$ be a power of ω. Every degree which is not totally α-c.a. bounds a degree which is totally α-c.a. but not uniformly so.*

This implies Theorem 1.2: indeed, even the refined hierarchy does not collapse at any level in any lower cone.

To prove the theorem, recall that a *modulus function* g has a "self-modulating" computable approximation $\langle g_s \rangle$:

- for all n and s, $g_s(n) \leqslant s$;
- if $g_s(n) \neq g_{s-1}(n)$ then $g_s(n) = s$, indeed $g_s(m) = s$ for all $m \geqslant n$.

So $g_{s-1} \leqslant g_s$ (pointwise), and if $g_s(n) \neq g_{s-1}(n)$ then $g_s(m) \neq g_{s-1}(m)$ for all $m \geqslant n$. Such a degree clearly has a c.e. degree, and every c.e. degree contains such a function. Indeed, given a c.e. degree $\mathbf{d}$ which is not totally α-c.a. we can find a modulus function $g \in \mathbf{d}$ which is not α-c.a. Fixing this g, we will give a self-modulating computable approximation $\langle f_s \rangle$ of a modulus function f such that $f \leqslant_{\mathrm{T}} g$, and $\deg_{\mathrm{T}}(f)$ is as required.

Requirements. Let $\langle \Phi_d \rangle_{d<\omega}$ be an enumeration of all functionals. To ensure that $\deg_{\mathrm{T}}(f)$ is totally α-c.a., we must meet the set of requirements given by:

$$\text{for all } d < \omega,\ Q_d : \text{If } \Phi_d(f) \text{ is total, then it is } \alpha\text{-c.a.}$$

To ensure that $\deg_{\mathrm{T}}(f)$ is not uniformly totally α-c.a., we fix an α-order function h and arrange that f itself is not h-c.a. We need to meet the requirements,

$$\text{for all } i < \omega,\ P^i : f \neq f^{h,i},$$

where $\left\langle f^{h,i} \right\rangle$ is a list of h-c.a. functions equipped with tidy $(h+1)$-c.a. approximations as described above.

Discussion. We apply the technique of non-total α-c.a. *permitting* to the construction from [DG20] that separates between totally α-c.a. and uniformly totally α-c.a. degrees.

To meet P^i, we would like to choose a follower p and change $f(p)$ each time we see that $f_s(p) = f_s^{h,i}(p)$. This action will be taken at most $h(p) + 1$ many "times". Any change in f necessitates a change in g below a previously specified location. Permission will not always be given, so a node σ working for P_i will appoint several followers, and we will argue that eventually it will receive permission; if not, then it appoints infinitely many followers and we use that to build an α-computable approximation for g.

Nodes τ working for Q_d can observe all ordinals of all followers appoints prior to the first stage at which a $\Phi_d(f)$-computation is certified. Since all approximations are "total", we do not have to wait for realisations and lift the use: all ordinals supplied to τ are below α, as they are bounded by h.

When we change $f(p)$ for some follower p for a node σ, we need to cancel all larger followers for σ. The reason is that this change may have injured a computation $\Phi_d(f, x)[s]$ that was not protected from p, but protected from p'; when the computation comes back, its use will be bigger than p'.

This has implications to the nature of the reduction of f to g. Originally we would like the use of this reduction to be the identity: say that p is permitted at a stage s if $g_s \upharpoonright p \neq g_{s+1} \upharpoonright p$. The idea is that if p will not receive enough permissions, then we can bound the "number" of changes to $g \upharpoonright p$. But when p' is cancelled, in order for this process to give an approximation for all of g, in the future, we need to allow g changes below p' to permit changes in $f(p)$. Hence the use of the reduction on p has risen beyond p'; the reduction is therefore a Turing reduction, and we will need to argue that the uses do stabilise.

Strategy tree. A node τ working for Q_d has two outcomes, $\infty <$ f; a node σ working for P^i has a unique outcome.

To define the reduction of f to g we will define moveable markers $\zeta_s(n)$. A change in g below $\zeta_s(n)$ allows us to change $f(n)$, and to redefine the marker $\zeta_s(n)$. We start with $\zeta_0(n) = n$ for all n.

Construction. Let s be a stage. Let σ be a node which works for requirement P^i. A follower p for σ is *permitted* at stage s if $g_s \upharpoonright \zeta_s(p) \neq g_{s+1} \upharpoonright \zeta_s(p)$. The follower *requires attention* if $f_s(p) = f^{h,i}_s(p)$. If some node σ has a follower which requires attention and is permitted at this stage, then we choose the strongest such σ, and the smallest such p for this σ, and define $f_{s+1}(m) = s+1$ for all $m \geqslant p$ (and $f_{s+1} \upharpoonright p = f_s \upharpoonright p$). We redefine $\zeta_{s+1}(m) = \max\{s+1, m\}$ for all $m \geqslant p$, and $\zeta_{s+1}(m) = \zeta_s(m)$ for all $m < p$. We cancel all followers $p' > p$ for σ, and initialise all nodes weaker than σ. We then end the stage.

Suppose that the stage did not end. We let $f_{s+1} = f_s$ and $\zeta_{s+1} = \zeta_s$. We then define the path γ_s of nodes accessible at stage s. The instructions for a node τ working for Q_d are as in the previous section; choose the outcome ∞ if $\operatorname{dom} \Phi_d(f)[s]$ is greater than the previous $\tau\hat{}\infty$-stage; otherwise choose f.

Suppose that a node σ working for requirement P^i is accessible at stage s. If for some follower p for σ we have $f_s(p) \neq f^{h,i}_s(p)$ then we let σ's child be next accessible. Otherwise (this includes the case that σ has no followers) we appoint a new, large follower p for σ, end the stage, and initialise all nodes weaker than σ.

Verification.

Lemma 6.4. *The sequence $\langle f_s \rangle$ converges to a limit f, which is computable from g.*

Proof. For every p we show that there are finitely many stages s at which $f_s(p) \neq f_{s+1}(p)$ but $f_s \upharpoonright p = f_{s+1} \upharpoonright p$. Fix p. If there is such a stage, then p is a follower for some node σ working for a requirement P^i. The usual argument shows that σ will act for p only finitely often: if it acts for p at stages $\bar{s} < s$ then as $f_{\bar{s}}(p) = f^{h,i}_{\bar{s}}(p)$ and $f_s(p) = f^{h,i}_s(p)$ and $f_s(p) \geqslant f_{\bar{s}+1}(p) = s+1 > f_s(p)$ implies that $o^{h,i}_{\bar{s}}(p) > o^{h,i}_s(p)$.

The fact that $\langle f_s \rangle$ stabilises implies that the sequence of use functions $\langle \zeta_s \rangle$ reaches a limit as well. The permitting instructions imply that if $g_s \upharpoonright \zeta_s(m) = g \upharpoonright \zeta_s(m)$ then $f_s \upharpoonright m = f \upharpoonright m$, whence $f \leqslant_{\mathrm{T}} g$. □

Meeting the requirement Q_d is done in the usual way, using the following:

Lemma 6.5. *Let $\tau \in \gamma_\omega$ be a node working for Q_d; let $\bar{t} < t$ be $\tau\hat{}\infty$-stages, let $\sigma \succcurlyeq \tau\hat{}\infty$ be a node working for P^i, and let p be a follower for σ at stage t which is appointed after stage $\bar{t}$. Then for all $x < \bar{t}$, $\Phi_d(f,x)\downarrow [t]$ and $p > \varphi_{d,t}(x)$.*

Proof. Similar to the previous section, except that we have to deal with more than one follower for each node. Let $u = \varphi_{d,r}(x)$, where $r > \bar{t}$ is the stage at which p was appointed; so $p > u$. Let $s \in (r,t)$ be a stage at which $f_s \restriction u \neq f_{s+1} \restriction u$; suppose that this is done on behalf of a follower $\bar{p}$ for a node $\bar{\sigma}$. If σ is stronger than $\bar{\sigma}$ then the latter is initialised at stage r, whence $\bar{p} > r > u$. If $\bar{\sigma}$ is stronger than σ then σ is initialised at stage s, and p cancelled. If $\bar{\sigma} = \sigma$ then $\bar{p} < u < p$, whence p would be cancelled at stage s. $\square$

The familiar process now gives an α-computable approximation to $\Phi_d(f)$ in case it is total and a node $\tau \in \gamma_\omega$ works for Q_d. We omit the details. The proof of the theorem will be complete once we show:

Lemma 6.6. *Let $\sigma \in \gamma_\omega$ work for requirement P^i. Then σ ends the stage only finitely many times, and the requirement is met.*

Proof. We claim that there is a follower p for σ which is never cancelled and such that $f(p) \neq f^{h,i}(p)$. If this is so, then the requirement is certainly met, and once both $f_s(p) = f(p)$ and $f^{h,i}_s(p) = f^{h,i}(p)$ permanently, no further followers will be appointed by σ, and eventually σ will cease all action.

Suppose then that for every follower p for σ, either p is eventually cancelled, or $f(p) = f^{h,i}(p)$. Since σ is accessible infinitely often, by considering "non-deficiency followers" (the smallest follower for σ ever to receive attention after a given stage) we see that there are infinitely many followers for σ that are never cancelled.

Fix $k < \omega$. Let $s^*(k)$ be the least stage s after which σ is never initialised, and at which there is a follower $p > k$ for σ. We let $p^*(k)$ be this follower. For all $s \geqslant s^*(k)$ we let $P_s(k)$ be the set of followers $p \leqslant p^*(k)$ for σ at stage s. Observe that if $\bar{s} < s$ then $P_{\bar{s}}(k) \supseteq P_s(k)$. Some followers in $P_{\bar{s}}(k)$ may get cancelled, but new followers are chosen large. On the other hand, $P_s(k)$ is nonempty for all $s \geqslant s^*(k)$; $\min P_s(k)$ is the first follower appointed for σ since the last stage it was initialised. For each $s \geqslant k$ we let $p_s(k) = \max P_s(k)$.

We let $s_0(k) < s_1(k) < s_2(k) < \dots$ be an enumeration of the stages $s \geqslant s^*(k)$ at which $f_s(p_s(k)) = f^{h,i}_s(p_s(k))$. By assumption, there are infinitely many such stages (note that $\langle p_s(k) \rangle$ stabilises). For $j \geqslant 0$ we let

$$\hat{g}_j(k) = g_{s_j(k)}(k)$$

and

$$o_j(k) = \bigoplus_{p \in P_{s_j(k)}(k)} o^{h,i}_{s_j(k)}(p) \, + \, |P_{s_j(k)}|.$$

Certainly $\lim_j \hat{g}_j(k) = g(k)$. Note that $o_j(k) < \alpha$. Since $P_s(k)$ is decreasing in s, we see that for all $j \geqslant 0$, $o_j(k) \geqslant o_{j+1}(k)$.

Claim 6.6.1. For all $s \geqslant s^*(k)$, $\zeta_s(p_s(k)) > k$.

Proof. By induction on s. At stage s^* we have $\zeta_s(p_{s^*}(k)) \geqslant p_{s^*}(k) > k$. Since $\zeta_s(p) \leqslant \zeta_{s+1}(p)$, we are done if $p_s(k) = p_{s+1}(k)$. Otherwise, at stage s we act for $p = p_{s+1}(k)$ and redefine $\zeta_{s+1}(p) = s + 1 > k$. $\square$

Now suppose that $\hat{g}_j(k) \neq \hat{g}_{j+1}(k)$. We need to show that $o_j(k) > o_{j+1}(k)$. For brevity let $\bar{s} = s_j(k)$ and $s = s_{j+1}(k)$. If $P_{\bar{s}}(k) \neq P_s(k)$ then $|P_{\bar{s}}(k)| > |P_s(k)|$ and we are done. Otherwise, let $p = p_{\bar{s}}(k) = p_s(k)$. We show that $o^{h,i}_{\bar{s}}(p) > o^{h,i}_s(p)$.

This is clear if $f^{h,i}_{\bar{s}}(p) \neq f^{h,i}_{s}(p)$. Suppose that $f^{h,i}_{\bar{s}}(p) = f^{h,i}_{s}(p)$. Since $f_{\bar{s}}(p) = f^{h,i}_{\bar{s}}(p)$ and $f_s(p) = f^{h,i}_{s}(p)$ we see that $f_{\bar{s}}(p) = f_s(p)$. This implies that for all $r \in [\bar{s}, s)$, $f_r(p) = f_{r+1}(p)$. On the other hand there is some stage $r \in [\bar{s}, s)$ such that $g_r(k) \neq g_{r+1}(k)$. By Claim 6.6.1, $\zeta_r(p) > k$. Since p is not cancelled at stage r, and we do not redefine $f_{r+1}(p) = r + 1 > f_r(p)$, it must be that $f_r(p) \neq f^{h,i}_{r}(p)$. Hence $f^{h,i}_{r}(p) \neq f^{h,i}_{s}(p)$, so $o^{h,i}_{\bar{s}}(p) > o^{h,i}_{r}(p) > o^{h,i}_{s}(p)$ as required.

Overall, we see that $\langle \hat{g}_j, o_j \rangle$ is an α-computable approximation for g, which we assumed does not exist. □

References

[Art] Katherine Arthur. *Maximality in the α-c.a. Degrees.* MSc Thesis, Victoria University of Wellington, 2016.

[CDW02] Peter Cholak, Rod Downey, and Stephen Walk. Maximal contiguous degrees. *J. Symbolic Logic*, 67(1):409–437, 2002.

[DG18] Rod Downey and Noam Greenberg. A hierarchy of computably enumerable degrees. *Bull. Symb. Log.*, 24(1):53–89, 2018.

[DG20] Rod G. Downey and Noam Greenberg. *A Hierarchy of Turing Degrees: A Transfinite Hierarchy of Lowness Notions in the Computably Enumerable Degrees, Unifying Classes, and Natural Definability*, volume 206 of *Annals of Mathematics Studies*. Princeton University Press, 2020.

[DGW07] Rod Downey, Noam Greenberg, and Rebecca Weber. Totally ω-computably enumerable degrees and bounding critical triples. *J. Math. Log.*, 7(2):145–171, 2007.

[DJS90] Rodney G. Downey, Carl G. Jockusch, Jr., and Michael Stob. Array nonrecursive sets and multiple permitting arguments. In *Recursion theory week (Oberwolfach, 1989)*, volume 1432 of *Lecture Notes in Math.*, pages 141–173. Springer, Berlin, 1990.

[EHK81] Richard L. Epstein, Richard Haas, and Richard L. Kramer. Hierarchies of sets and degrees below $\mathbf{0}'$. In *Logic Year 1979–80 (Proc. Seminars and Conf. Math. Logic, Univ. Connecticut, Storrs, Conn., 1979/80)*, volume 859 of *Lecture Notes in Math.*, pages 32–48. Springer, Berlin, 1981.

[Ers68a] Yuri L. Ershov. A certain hierarchy of sets. I. *Algebra i Logika*, 7(1):47–74, 1968.

[Ers68b] Yuri L. Ershov. A certain hierarchy of sets. II. *Algebra i Logika*, 7(4):15–47, 1968.

[Ers70] Yuri L. Ershov. A certain hierarchy of sets. III. *Algebra i Logika*, 9:34–51, 1970.

[Ish99] Shamil Ishmukhametov. Weak recursive degrees and a problem of spector. In M Arslanov and S Lempp, editors, *Recursion Theory and Complexity*, volume 2, pages 81–87. de Gruyter, Berlin, 1999.

[Kum96] Martin Kummer. Kolmogorov complexity and instance complexity of recursively enumerable sets. *SIAM J. Comput.*, 25(6):1123–1143, 1996.

[Sch98] Benjamin Schaeffer. Dynamic notions of genericity and array noncomputability. *Annals of Pure and Applied Logic*, 95(1-3):37–69, 1998.

[Sel89] V. L. Selivanov. Fine hierarchies of arithmetic sets, and definable index sets. *Trudy Inst. Mat. (Novosibirsk)*, 12(Mat. Logika i Algoritm. Probl.):165–185, 190, 1989.

[Sho59] J. R. Shoenfield. On degrees of unsolvability. *Ann. of Math. (2)*, 69:644–653, 1959.

School of Mathematics and Statistics, Victoria University, Wellington