CODING IN THE AUTOMORPHISM GROUP OF A COMPUTABLY CATEGORICAL STRUCTURE

DAN TURETSKY

ABSTRACT. Using new techniques for controlling the categoricity spectrum of a structure, we construct a structure with degree of categoricity but infinite spectral dimension, answering a question of Bazhenov, Kalimullin and Yamaleev. Using the same techniques, we construct a computably categorical structure of non-computable Scott rank.

1. INTRODUCTION

Isomorphism problems for computable structures are one of the main topics in computable model theory. This is a broadly defined topic concerned with the complexity of isomorphisms between isomorphic computable structures. Recent work has focused on degrees of categoricity and of isomorphism [FKM10, CFS13, AC16, CHT17, Fra17, BKY18], and lowness for categoricity and for isomorphism [AC16, FS14, Fra17, FT18, MS18]. We refer the reader to the text by Ash and Knight [AK00] for background on computable model theory and computable ordinals.

Given isomorphic computable structures $\mathcal{A}$ and $\mathcal{B}$, the *degree of isomorphism* between $\mathcal{A}$ and $\mathcal{B}$ is the least Turing degree $\mathbf{d}$ which can compute an isomorphism between $\mathcal{A}$ and $\mathcal{B}$, if such a degree exists. In this case, every isomorphism between $\mathcal{A}$ and $\mathcal{B}$ computes $\mathbf{d}$, so we can think of $\mathbf{d}$ as being coded into the isomorphism problem of $\mathcal{A}$ and $\mathcal{B}$.

A related notion is the degree of categoricity. Given a computable structure $\mathcal{A}$, the *degree of categoricity* of $\mathcal{A}$ is the least degree which computes an isomorphism between any two computable presentations of $\mathcal{A}$, if such a degree exists. Fokina, Kalimullin and R. Miller introduced this notion and showed that every degree which is d.c.e. in and above some $\mathbf{0}^{(n)}$ is the degree of categoricity of some computable structure [FKM10]. Csima, Franklin and Shore extended this to every degree which is d.c.e. in and above some $\mathbf{0}^{(\alpha+1)}$ [CFS13]. Recently, Csima and Ng have announced that every Δ_2^0 degree is the degree of categoricity for some computable structure.

When building a structure to have degree of categoricity $\mathbf{d}$, one must achieve two goals: $\mathbf{d}$ must be able to compute an isomorphism between any two computable presentations; and $\mathbf{d}$ must be the least degree which can do this. The natural way to achieve the second goal is to build two presentations, $\mathcal{A}$ and $\mathcal{B}$, such that every isomorphism between $\mathcal{A}$ and $\mathcal{B}$ computes $\mathbf{d}$. In other words, $\mathbf{d}$ is the degree of isomorphism for the pair $\mathcal{A}, \mathcal{B}$. When there exists such a pair of presentations, we say that $\mathbf{d}$ is the *strong degree of categoricity* of $\mathcal{A}$.

The original constructions of degrees of categoricity actually constructed strong degrees of categoricity. It is natural to wonder whether there is a degree which is the degree of categoricity of some computable structure, but is not the strong degree

of categoricity of any computable structure. This question remains open—so far, every degree which is known to be the degree of categoricity of some computable structure is also known to be the strong degree of categoricity of some computable structure.

A simpler question is whether there exists a structure $\mathcal{A}$ with some degree of categoricity $\mathbf{d}$, but $\mathbf{d}$ is not a strong degree of categoricity for $\mathcal{A}$. Bazhenov, Kalimullin and Yamaleev [BKY18], and separately Csima and Stephenson [CS19], showed that this can occur. They constructed a structure $\mathcal{A}$ with degree of categoricity $\mathbf{0}'$, but such that $\mathbf{0}'$ is not the degree of isomorphism for any two computable presentations of $\mathcal{A}$. In both the Bazhenov-Kalimullin-Yamaleev example and the Csima-Stephenson example the structure is rigid, and there are three computable presentations, $\mathcal{A}_0, \mathcal{A}_1, \mathcal{A}_2$, such that if f_1 is the unique isomorphism between $\mathcal{A}_0$ and $\mathcal{A}_1$, and f_2 is the unique isomorphism between $\mathcal{A}_0$ and $\mathcal{A}_2$, then $f_1 \oplus f_2 \equiv_T \emptyset'$. So $\mathbf{0}'$ is the degree of isomorphism for the triple $\mathcal{A}_0, \mathcal{A}_1, \mathcal{A}_2$, in the sense that it is the least degree able to compute an isomorphism between any two elements of the triple.

Bazhenov et al. observed that their methods can be generalized to produce, for any $n \in \omega$, a computable structure $\mathcal{A}$ such that $\mathbf{0}'$ is the degree of categoricity of $\mathcal{A}$, but $\mathbf{0}'$ is not the degree of isomorphism for any set of n presentations $\mathcal{A}_0, \dots, \mathcal{A}_{n-1}$. For the examples constructed using these methods, there will always be a finite set of presentations $\mathcal{A}_0, \dots, \mathcal{A}_{k-1}$ such that $\mathbf{0}'$ is the degree of isomorphism of this set. They defined the term spectral dimension to describe this phenomenon: for a computable structure $\mathcal{A}$ with degree of categoricity $\mathbf{d}$, the *spectral dimension* of $\mathcal{A}$ is the least cardinality of a set of presentations of $\mathcal{A}$ for which the degree of isomorphism is $\mathbf{d}$.[1] They then asked whether a finite spectral dimension is necessary. Is there a computable structure with a degree of categoricity $\mathbf{d}$, but with infinite spectral dimension? We show in Theorem 4 that there is such a structure, where $\mathbf{d} = \mathbf{0}''$.

Changing gears slightly, a computable structure is *computably categorical* if it has degree of categoricity $\mathbf{0}$—between any two computable copies, there is a computable isomorphism. A computable structure $\mathcal{A}$ is *relatively Δ_α^0-categorical* if for any (not necessarily computable) copy $\mathcal{B}$ of $\mathcal{A}$, there is a $\Delta_\alpha^0(\mathcal{B})$ isomorphism between $\mathcal{B}$ and $\mathcal{A}$. Observe that a relatively Δ_1^0-categorical structure is computably categorical. Goncharov [Gon77] showed that the converse fails. Downey, Kach, Lempp, Lewis, Montalbán and Turetsky [DKL$^+$15] showed that for every $\alpha < \omega_1^{\mathrm{ck}}$, there is a computably categorical structure which is not relatively Δ_α^0-categorical. They used their techniques to show that the set of (indices for) computably categorical structures is Π_1^1-complete.

The structures built by Downey et al. are always relatively Δ_β^0-categorical for some $\beta < \omega_1^{\mathrm{ck}}$. It is thus natural to ask whether there is a computably categorical structure which is not relatively Δ_α^0-categorical for any $\alpha < \omega_1^{\mathrm{ck}}$. This turns out to relate to several other notions.

By the Spector-Gandy theorem, every Π_1^1 formula $\forall f\, \theta(f, n)$ can be reinterpreted as $\exists f \in \Delta_1^1\, \theta'(f, n)$, where θ' is an arithmetic relation. As $\Delta_1^1 = \bigcup_{\alpha < \omega_1^{\mathrm{ck}}} \Delta_\alpha^0$, this implies that every Π_1^1 set can be stratified into ω_1^{ck} levels: if X is a Π_1^1 set with

---

[1] The definition given here is a special case of Bazhenov et al.'s definition. They defined spectral dimension for all computable structures, rather than just those with a degree of categoricity, using the notion of categoricity spectrum.

$n \in X \iff \exists f \in \Delta_1^1 \, \theta(n, f)$, then

$$X = \bigcup_{\alpha < \omega_1^{\mathrm{ck}}} \{n : \exists f \in \Delta_\alpha^0 \, \theta(f, n)\}.$$

For a given Π_1^1 set X, it can be enlightening to examine stratifications $X = \bigcup_{\alpha < \omega_1^{\mathrm{ck}}} X_\alpha$, where each X_α is uniformly Δ_α^0. While the argument above always gives such a stratification, there is no known natural stratification for the set of computably categorical structures.[2] If every computably categorical structure were relatively Δ_α^0-categorical for some $\alpha < \omega_1^{\mathrm{ck}}$, this would give us such a stratification: categorize computably categorical structures based on the α such that they are relatively Δ_α^0-categorical.

Let us briefly recall the notion of the Scott rank of a structure. For tuples $\bar{a}, \bar{b} \in \mathcal{A}$ of the same length, define $\bar{a} \equiv_0 \bar{b}$ if $\bar{a}$ and $\bar{b}$ satisfy the same atomic formulae in $\mathcal{A}$. For an ordinal $\alpha > 0$, define $\bar{a} \equiv_\alpha \bar{b}$ if for every $\beta < \alpha$ and every $\bar{c} \in \mathcal{A}$ there exist $\bar{d}_0, \bar{d}_1 \in A$ such that $\bar{a}\bar{c} \equiv_\beta \bar{b}\bar{d}_0$ and $\bar{a}\bar{d}_1 \equiv_\beta \bar{b}\bar{c}$. Define $r(\bar{a})$ to be the least α such that for all $\bar{b}$, $\bar{a} \equiv_\alpha \bar{b}$ implies $\bar{a}$ and $\bar{b}$ are in the same orbit. By a Σ_1^1-bounding argument, for a computable structure $\mathcal{A}$, $r(\bar{a})$ will always exist and be an ordinal at most ω_1^{ck}. The *Scott rank* of $\mathcal{A}$ is defined to be $\sup_{\bar{a} \in \mathcal{A}}(r(\bar{a}) + 1)$.

The following is a synthesis of published results and folklore.

Proposition 1. *For a computable structure $\mathcal{A}$, the following are equivalent:*

    (1) $\mathcal{A}$ is relatively Δ_1^1-categorical.
    (2) For some $\alpha < \omega_1^{\mathrm{ck}}$, $\mathcal{A}$ is relatively Δ_α^0-categorical.
    (3) The Scott rank of $\mathcal{A}$ is a computable ordinal.
    (4) The orbits of $\mathcal{A}$ are uniformly hyperarithmetic.

Proof. $(2) \Rightarrow (1)$ is immediate. For the converse, fix a f a Σ_1^1-generic permutation of ω, and define $\mathcal{B}$ as the pullback of $\mathcal{A}$ along f, so that $f : \mathcal{B} \to \mathcal{A}$ is an isomorphism. As f is Σ_1^1-generic, $\omega_1^{\mathrm{ck}}(f) = \omega_1^{\mathrm{ck}}$ [GM17]. As $\mathcal{B} \leqslant_T f$, $\omega_1^{\mathrm{ck}}(\mathcal{B}) = \omega_1^{\mathrm{ck}}$.

Since $\mathcal{A}$ is relatively Δ_1^1-categorical, there is a $g \in \Delta_1^1(\mathcal{B})$ with $g : \mathcal{B} \to \mathcal{A}$. So $g \in \Delta_\alpha^0(\mathcal{B})$ for some $\alpha < \omega_1^{\mathrm{ck}}(\mathcal{B}) = \omega_1^{\mathrm{ck}}$. By Ash's proof of the characterization of relatively Δ_α^0-categoricity using Scott families [Ash87], $\mathcal{A}$ is relatively Δ_α^0-categorical. The key observation is that Ash's proof does not use the full power of Δ_α^0-categoricity, but just the existence of a $\Delta_\alpha^0(\mathcal{B})$ isomorphism, where $\mathcal{B}$ is defined to be a sufficiently generic permutation of $\mathcal{A}$.

$(2) \Leftrightarrow (3)$ is by the standard Scott rank analysis and the characterization of relatively Δ_α^0-categoricity using Scott families (see Ash & Knight section 6.7 and theorem 10.14 [AK00]).

$(3) \Rightarrow (4)$ is by the standard Scott rank analysis, while $(4) \Rightarrow (3)$ is by Σ_1^1-bounding. $\square$

In Theorem 2, we construct a computably categorical structure which has non-computable Scott rank, and thus a structure which is not relatively Δ_1^1-categorical, defeating the earlier potential stratification of the computably categorical structures.

With a simple modification, the computably categorical structure we create in Theorem 2 can be turned into a structure of computable dimension 2. A computable structure has *computable dimension 2* if it has exactly two computable copies up

---

[2]We acknowledge that our criteria for naturalness is vague.

to computable isomorphism. Prior constructions of such structures used a method called the special component technique [Gon77, CGK99, HKS03]. When a structure of computable dimension 2 is built using this technique, the two computable copies which are not computably isomorphic will be Δ_3^0 isomorphic. In contrast, for our structure of computable dimension 2, there is no hyperarithmetic isomorphism between the two copies.

The proofs of Theorems 2 and 4 use similar techniques, although Theorem 4 is the more complicated construction. We recommend the reader begin with Theorem 2 as a warmup. Both constructions are based on an extension of the techniques of Franklin and Turetsky for coding Π_1^0-classes into the automorphism group of a structure [FT], while also incorporating the methods of Downey et al. for constructing computably categorical structures [DKL$^+$15].

2. A computably categorical structure with high Scott rank

This section is devoted to the proof of the following theorem.

Theorem 2. *There is a computably categorical structure with non-computable Scott rank.*

Fix $T \subset \omega^{<\omega}$ the tree of descending sequences through the Harrison ordering. The only important properties of T are that it is computable and that the set of extendible nodes in T is not hyperarithmetic.

Our structure will have U a unary relation, $(W_\sigma)_{\sigma \in \omega^{<\omega}}$ all unary relations, $(E_i)_{i \in \omega}$ all binary relations, P a binary relation, f a unary function, and $(V_n)_{n \in \omega}$ all unary relations. Our structure's universe will be $([\omega]^{<\omega} \times \omega^{<\omega}) \sqcup C$, where C is an infinite computable set.

The purpose of U is to distinguish C from the rest of the structure, so $U(x)$ will hold if and only if $x \in C$. P, the E_i and the W_σ will never hold with elements of C, while the V_n will never hold with elements outside of C. The purpose of C, f and V_n is to allow us to apply labels to elements of $[\omega]^{<\omega} \times \omega^{<\omega}$ in a c.e. fashion. For each $x \in C$, $f(x) \notin C$, and for each $x \notin C$, $f(x) = x$. Also, for each $x \in C$, there will be a unique n such that $V_n(x)$ holds. For $y \notin C$, we will write $S_n(y)$ for "$\exists x \in C\ f(x) = y \wedge V_n(x)$". During our construction, there will be times when we wish to declare $S_n(y)$ for some n and y. When we do, we choose the first element $x \in C$ which has not yet been used and define $V_n(x)$ and $f(x) = y$. We simultaneously define $\neg V_m(x)$ for all $m \neq n$. As there will be infinitely many times we wish to make such a declaration, this will completely define the structure on C. For simplicity of presentation, we may sometimes declare $S_n(y)$ and then make the same declaration again at a later stage. In this case, we do not mean to choose a new x for this declaration; instead, the duplicate declaration should simply be ignored.

For the remainder of the construction, we will not mention U, C, f or the V_n. Instead, we will proceed as if our structure has universe $[\omega]^{<\omega} \times \omega^{<\omega}$ and our language is $(W_\sigma)_{\sigma \in \omega^{<\omega}}$, $(E_i)_{i \in \omega}$, P, $(S_n)_{n \in \omega}$, and each S_n only needs to be declared in a c.e. fashion rather than a computable fashion. Note that this does not affect computable categoricity: if we can find a computable isomorphism g between two presentations $\mathcal{A}_0$ and $\mathcal{A}_1$ of this new structure, we can extend to a computable isomorphism between the corresponding versions of the old structure – for each $y \in \mathcal{A}_0$, when we see an x with $f(x) = y$ and $V_n(x)$, we search for an x' with $f(x') = g(y)$ and $V_n(x')$, and then define $g(x) = x'$.

With the exception of the S_n, we can give a complete description of our structure immediately. For $(F, \tau), (G, \rho) \in [\omega]^{<\omega} \times \omega^{<\omega}$:

- $W_\sigma((F, \tau))$ holds iff $\sigma = \tau$.
- $E_i((F, \tau), (G, \rho))$ holds iff $\tau = \rho$ and $F \triangle G = \{i\}$.
- $P((F, \tau), (G, \rho))$ holds iff $\rho = \tau {}^\frown i$ for some i, and one of the following holds:
  - $i \notin F$ and $|G|$ is even; or
  - $i \in F$ and $|G|$ is odd.

We think of $[\omega]^{<\omega}$ as an affine space acted on by $\bigoplus_{i<\omega} \mathbb{Z}/2$, where $F + e_i = F \triangle i$. Then $E_i(F, G) \iff F + e_i = G$. Alternatively, we can think of $[\omega]^{<\omega}$ as the vertices of an infinite dimensional cube, where there is an edge between F and G iff $|F \triangle G| = 1$. If $F \triangle G = \{i\}$, we color the edge between F and G with color i, and this is represented by the relation E_i. In either case, we assign a copy of this structure to every string in $\omega^{<\omega}$, and the various W_σ let us easily identify which copy a given element belongs to.

Claim 2.1. *The automorphisms of $([\omega]^{<\omega}, (E_i)_{i \in \omega})$ are precisely the maps of the form $g(F) = F \triangle H$ for some fixed $H \in [\omega]^{<\omega}$.*

Proof. To see that such a map is an automorphism, observe that

$$
\begin{aligned}
E_i(F, G) &\iff F \triangle G = \{i\} \\
&\iff F \triangle G \triangle \emptyset = \{i\} \\
&\iff F \triangle G \triangle (H \triangle H) = \{i\} \\
&\iff (F \triangle H) \triangle (G \triangle H) = \{i\} \\
&\iff E_i(g(F), g(G)).
\end{aligned}
$$

Conversely, suppose g is an automorphism. Let $H = g(\emptyset)$. We prove by induction on $|F|$ that $g(F) = F \triangle H$. The case $|F| = 0$ is immediate. For $|F| > 0$, fix $i \in F$, and let $G = F - \{i\}$. Then $E_i(F, G)$, so $E_i(g(F), g(G))$, and thus

$$
\begin{aligned}
g(F) \triangle g(G) &= \{i\} \\
g(F) &= g(G) \triangle \{i\} \\
g(F) &= (G \triangle H) \triangle \{i\} \\
g(F) &= (G \triangle \{i\}) \triangle H \\
g(F) &= F \triangle H. \qquad \square
\end{aligned}
$$

Interpreting $[\omega]^{<\omega}$ as an affine space, it partitions into two hyperplanes perpendicular to e_i: $\{F : i \notin F\}$ is one, and $\{F : i \in F\}$ is the other. Under the cube interpretation, these are the two infinite components that result if all the edges with color i are deleted. If we are examining the copy of the structure assigned to $\tau \in \omega^{<\omega}$, we have associated these two sets with a partition of the copy assigned to $\tau {}^\frown i$: the former set is associated, via P, with the elements of even cardinality, while the latter set is associated with the elements of odd cardinality.

During the course of our construction, we will build an arithmetic tree $Q \subseteq \omega^{<\omega}$ which will be arithmetically isomorphic to T. For $\sigma \in Q$, $S_n((F, \sigma))$ will hold for every n and every $F \in [\omega]^{<\omega}$. For $\sigma \notin Q$, there will be an n such that $S_n((F, \sigma))$ holds iff $F = \emptyset$. The purpose of this setup is the following.

Claim 2.2. *The automorphisms of $\mathcal{A}$ which send $(\emptyset, \sigma)$ to $(\{i\}, \sigma)$ are Muchnik equivalent to $[\sigma^\frown i] \cap [Q]$. More generally, the automorphisms of $\mathcal{A}$ which send $(\emptyset, \sigma)$ to (F, σ) are Muchnik equivalent to the join $\bigoplus_{i \in F}([\sigma^\frown i] \cap [Q])$.*

In particular, such an automorphism exists iff the appropriate set is nonempty.

Proof. Suppose g is an automorphism of $\mathcal{A}$ sending $(\emptyset, \sigma)$ to (F, σ), and $i \in F$. We construct, effectively in g, a sequence $\sigma_0 \subset \sigma_1 \subset \sigma_2 \subset \ldots$ with $\sigma_0 = \sigma^\frown i$ and maintaining the inductive assumption that $g((\emptyset, \sigma_j)) \neq (\emptyset, \sigma_j)$. By our earlier discussion about the S_n and their relationship to Q, it will follow that each $\sigma_j \in Q$.

Note that $P((\emptyset, \sigma), (\emptyset, \sigma^\frown i))$, but $\neg P((F, \sigma), (\emptyset, \sigma^\frown i))$. This shows that $g((\emptyset, \sigma^\frown i)) \neq (\emptyset, \sigma^\frown i)$. So $\sigma_0 = \sigma^\frown i$ satisfies the inductive assumption.

Now, suppose we have defined σ_j. Let G be such that $g((\emptyset, \sigma_j)) = (G, \sigma_j)$. By assumption, $G \neq \emptyset$, so fix $k \in G$. As $P((\emptyset, \sigma_j), (\emptyset, \sigma_j^\frown k))$ but $\neg P((G, \sigma_j), (\emptyset, \sigma_j^\frown k))$, it must be that $g((\emptyset, \sigma_j^\frown k)) \neq (\emptyset, \sigma_j^\frown k)$. So we let $\sigma_{j+1} = \sigma_j^\frown k$.

Conversely, fix $(h_i)_{i \in F}$ with $h_i \in ([\sigma^\frown i] \cap [Q])$. We will construct, effectively in $(h_i)_{i \in F}$, an automorphism g of $\mathcal{A}$ sending $(\emptyset, \sigma)$ to (F, σ). Define $g((G, \sigma)) = (G \triangle F, \sigma)$ for all $G \in [\omega]^{<\omega}$. For each τ which is not an initial segment of any h_i, we define $g((G, \tau)) = (G, \tau)$. For τ extending σ and an initial segment of h_i, fix k such that $\tau^\frown k$ is an initial segment of h_i. We define $g((G, \tau)) = (G \triangle \{k\}, \tau)$ for all $G \in [\omega]^{<\omega}$. For τ a proper initial segment of σ, our construction depends on the parity of $|F|$. If $|F|$ is even, we define $g((G, \tau)) = (G, \tau)$ for all $G \in [\omega]^{<\omega}$. If $|F|$ is odd, we fix the k such that $\tau^\frown k$ is an initial segment of σ and define $g((G, \tau)) = (G \triangle \{k\}, \tau)$ for all $G \in [\omega]^{<\omega}$.

As $g((G, \tau)) = (G, \tau)$ for any τ which is not an initial segment of any h_i, and thus for any $\tau \notin Q$, we see that g respects all the S_n. As $g((G, \tau)) = (G \triangle H_\tau, \tau)$ for some H_τ, for every τ, we see that g respects all the E_i and W_τ. Finally, $|H_{\tau^\frown k}|$ is odd precisely if $k \in H_\tau$, and so g respects P. Thus g is an automorphism. $\square$

Claim 2.3. $\mathcal{A}$ *has non-computable Scott rank.*

Proof. It suffices to show that the orbits of $\mathcal{A}$ are not uniformly hyperarithmetic. Note that $\sigma \in Q$ is extendible to a path in $[Q]$ precisely if there is an i such that $(\emptyset, \sigma)$ and $(\{i\}, \sigma)$ are in the same orbit. Thus if the orbits of $\mathcal{A}$ were uniformly hyperarithmetic, the set of extendible nodes in Q would be hyperarithmetic. But Q is arithmetically isomorphic to T, the descending sequences through the Harrison order, and the set of extendible nodes of T is not hyperarithmetic. $\square$

In fact, one can make a similar argument to show that the orbit of $(\emptyset, \langle\rangle)$ is not hyperarithmetic, and thus the Scott rank of $\mathcal{A}$ is $\omega_1^{\mathrm{ck}} + 1$.[3]

Construction. Fix $(\mathcal{M}_i)_{i \in \omega}$ a listing of all partial computable structures in our language.

We will perform a Π_2^0 priority construction on a tree of strategies. We have strategies of type N_π for $\pi \in T$, which are responsible for choosing the image of π in Q and ensuring that the S_n are as described at this string. We also have strategies of type M_i for $i \in \omega$, which are responsible for constructing a computable isomorphism between $\mathcal{A}$ and $\mathcal{M}_i$ when $\mathcal{A} \cong \mathcal{M}_i$. Finally, we have a global strategy G which is responsible for ensuring that the S_n are as described for those $\sigma \in \omega^{<\omega}$ not chosen by an N_π-strategy.

---

[3]And even further, $\mathcal{A}$ has Scott complexity $\Pi_{\omega_1^{\mathrm{ck}}+2}$.

Each N_π-strategy will choose at most one string σ to be the target for π. If τ is an N_π-strategy which chooses σ as the image of π, we say that τ has *chosen* σ. No string σ will be chosen by more than one strategy, and it will be the case that if $\sigma \in s^{<s}$ has not been chosen by the beginning of stage s, then it will never be chosen, and thus will not be in Q.

We arrange the N_π and M_i in some priority ordering of order type ω, such that for $\pi_0 \subset \pi_1 \in T$, N_{π_0} occurs before N_{π_1} in the ordering, and such that $N_{\langle\rangle}$ occurs first in the ordering. The set of possible outcomes for an M_i-strategy is $\{k, \infty_k : k \in \omega\} \cup \{\infty_\infty\}$. We call the outcomes of the form ∞_k for $k \in \omega \cup \{\infty\}$ the *infinite outcomes*. Each N_π-strategy will have only a single outcome: `outcome`. We construct a priority tree of these strategies.

Strategy for G. At the end of stage s, for every $\sigma \in s^{<s}$ not yet chosen by any strategy, we declare $S_0((\emptyset, \sigma))$.

Growing a string σ. At a stage s, a strategy may declare that σ is growing. When this happens, we let n be largest such that we have already declared $S_n((\emptyset, \sigma))$ (or $n = -1$ if there is no such n). We declare $S_k((\emptyset, \sigma))$ for $0 \leqslant k < n + 2$. We also declare $S_k((F, \sigma))$ for every $F \subseteq s$ and $0 \leqslant k < n$.

Strategy for $N_{\langle\rangle}$. This strategy is at the root of the priority tree and is visited at every stage. It declares $\langle\rangle$ to be the image of $\langle\rangle$. At every stage s, when this strategy is visited, the string $\langle\rangle$ grows. This strategy always takes its unique outcome of `outcome`.

Strategy for $N_{\pi^\frown i}$. Suppose τ is a strategy on the priority tree for requirement $N_{\pi^\frown i}$. Then there is a unique $\rho \subset \tau$ which is a strategy for N_π. It has declared some unique string σ to be the image of π. Let s_0 be the first stage at which τ is visited. At this stage, we choose an $m > s_0$ which has not yet been mentioned in the construction and declare $\sigma^\frown m$ to be the image of $\pi^\frown i$.

At any stage when τ is visited, $\sigma^\frown m$ grows.

At every stage τ is visited, it takes its unique outcome of `outcome`.

An auxiliary definition. For every σ and s with $\sigma \in (s-1)^{<s-1}$, we define $n_\sigma(s)$ to be the largest n such that we have declared $S_n((\emptyset, \sigma))$ by the start of stage s. By our action for G and the N_π-strategies, $n_\sigma(s)$ is always defined.

Strategy for M_i. Suppose τ is a strategy on the priority tree for requirement M_i. Let C_τ be those strings σ which are chosen by some N_π-strategy ρ with $\rho \subset \tau$. It will be the case that C_τ is fully determined by the first stage at which τ is visited. We construct a partial computable function $f_\tau : \omega^{<\omega} \to \mathcal{M}_i$. We will also define sequences $x_\tau(\sigma, s)$ for $\sigma \in C_\tau$. If $\mathcal{M}_i \cong \mathcal{A}$ and τ is on the true path, we will use f_τ and these sequences to post hoc construct an isomorphism between $\mathcal{A}$ and $\mathcal{M}_i$.

When τ is visited at stage s, let k_0 be the number of times τ has previously taken an infinite outcome, and let $t < s$ be the last stage at which τ took an infinite outcome ($t = 0$ if there is no such stage). Let k_1 be the number of times τ has previously taken outcome ∞_∞. Let $B_\tau(s)$ consist of every string $\sigma \in t^{<t}$ except for those which have been chosen by some N_π-strategy ρ with $\rho \supseteq \tau^\frown k_0$ or $\rho \subset \tau$. For each $\sigma \in B_\tau(s)$, if $f_\tau(\sigma)$ has not yet been defined, we search for s steps for an element $x \in \mathcal{M}_i$ with $W_\sigma(x)$ and $S_{n_\sigma(t+1)}(x)$. If we find such an element, we define $f_\tau(\sigma) = x$.

If there is at least one $\sigma \in B_\tau(s)$ for which at least one of the following fails, then we finish our action for τ at stage s and take outcome k_0:

- $f_\tau(\sigma)$ is defined;
- $\mathcal{M}_i \models S_{n_\sigma(t+1)}(f_\tau(\sigma))$; and
- For every j with $\sigma^\frown j \in B_\tau(s)$, $\mathcal{M}_i \models P(f_\tau(\sigma), f_\tau(\sigma^\frown j))$,

If the above holds for every $\sigma \in B_\tau(s)$, we will have an infinite outcome at stage s, but it remains to determine which. First, let $D_\tau(s)$ consist of those strings $\sigma \in B_\tau(s)$ which have not been chosen by a strategy extending $\tau^\frown \infty_\infty$. For each $\sigma \in C_\tau$, we let $x_\tau(\sigma, s)$ be the oldest element x of $\mathcal{M}_i$ such that $\mathcal{M}_i \models W_\sigma(x) \wedge P(x, f_\tau(\sigma^\frown j))$ for every $\sigma^\frown j \in D_\tau(s)$, if we see such an x at stage s, and we leave $x_\tau(\sigma, s)$ undefined if there is no such x.

If, for every $\sigma \in C_\tau$, both $x_\tau(\sigma, s)$ and $x_\tau(\sigma, t)$ are defined, and $x_\tau(\sigma, s) = x_\tau(\sigma, t)$, then we take outcome ∞_{k_1}. Otherwise, we take outcome ∞_∞.

Running the construction. At stage s, we begin by visiting the strategy $\langle \rangle$. After having visited a strategy τ at stage s with $|\tau| < s$, we let k be the outcome taken by τ at this stage. We next visit the strategy $\tau^\frown k$. After having visited a strategy τ with $|\tau| = s$, we run the global strategy and then end the stage.

We put the lexicographic ordering on the tree of strategies, where we order the outcomes of M_i strategies as $\infty_\infty < \cdots < \infty_2 < \infty_1 < \infty_0 < \cdots < 2 < 1 < 0$. Observe that our construction has the property that if a strategy τ is visited, and then at a later stage a strategy ρ which is lexicographically to the left of τ is visited, τ can never again be visited.

The true path. We define the true path recursively, maintaining the inductive hypothesis that every strategy on the true path is visited infinitely often. $\langle \rangle$ is on the true path. If τ is on the true path and τ is an N_π-strategy, then $\tau^\frown$outcome is on the true path.

If τ is on the true path and τ is an M_i-strategy, let α be the leftmost outcome which τ takes infinitely often. Observe that this necessarily exists by our action for τ. Then $\tau^\frown \alpha$ is on the true path. We call α the *true outcome* of τ.

By the earlier observation, no strategy off the true path is visited infinitely often. Observe that the true path is computable from $\emptyset''$.

Verification. We begin by defining Q and an isomorphism $\varphi : T \to Q$. For $\pi \in T$, let τ be the unique N_π-strategy along the true path. We define $\varphi(\pi)$ to be the string which τ declared to be the image of π. As the true path is Δ^0_3, φ is arithmetic. We then define Q to be the image of T under φ. It is a straightforward induction to show that Q is a tree and φ is an isomorphism.

Claim 2.4. *If $\sigma \in Q$, then $\mathcal{A} \models S_n(F, \sigma)$ for every $F \in [\omega]^{<\omega}$.*
If $\sigma \notin Q$, then $n_\sigma = \lim_s n_\sigma(s)$ exists, and $\mathcal{A} \models S_{n_\sigma}(F, \sigma)$ iff $F = \emptyset$.

Proof. If $\sigma \in Q$, then σ was selected by some N_π-strategy τ along the true path. As τ is visited infinitely often, our action for τ ensures that $\mathcal{A} \models S_n(F, \sigma)$ for every $F \in [\omega]^{<\omega}$.

If $\sigma \notin Q$, then either σ was never chosen by an N_π-strategy, or it was chosen by one off the true path. In the former case, $n_\sigma = 0$, and we never declared $S_0((F, \sigma))$ for any $F \neq \emptyset$. In the latter case, the strategy was only visited finitely often, so n_σ exists. By construction, we have not declared $S_{n_\sigma}((F, \sigma))$ for any $F \neq \emptyset$. $\square$

Next, we must show that $\mathcal{A}$ is computably categorical.

Claim 2.5. *Fix τ an M_i-strategy.*

If τ is visited at stages $s_0 < s_1$, then $B_\tau(s_0) \subseteq B_\tau(s_1)$.

Further, if there is no stage $t \in [s_0, s_1)$ at which τ takes an infinite outcome, then $B_\tau(s_0) = B_\tau(s_1)$.

Proof. For $i < 2$, let k_i be the number of times τ took an infinite outcome before stage s_i, and let t_i be the last stage before s_i at which τ took an infinite outcome ($t_i = 0$ if there is no such stage). By definition, if there were some $\sigma \in B_\tau(s_0) - B_\tau(s_1)$, then it would have to be the case that $\sigma \in t_0^{<t_0}$ (and so $t_0 > 0$), σ was selected by some strategy extending $\tau^\frown k_1$, and $k_0 \neq k_1$. But no strategy extending $\tau^\frown k_1$ is accessible before stage s_0, and σ cannot be selected after stage t_0, and $t_0 < s_0$.

By definition, if there is $\sigma \in B_\tau(s_1) - B_\tau(s_0)$ then either $t_1 > t_0$ or σ was selected by a strategy extending $\tau^\frown k_0$ which does not extend $\tau^\frown k_1$, and thus $k_0 \neq k_1$ (which again implies $t_1 > t_0$). In either case, we see that there is some stage $t \in [s_0, s_1)$ at which τ took an infinite outcome. □

Claim 2.6. *Fix τ the M_i-strategy along the true path. If τ's true outcome is some $k \in \omega$, then $\mathcal{A} \not\cong \mathcal{M}_i$.*

Proof. Note that k is the number of times τ ever takes an infinite outcome. Let t be the last stage at which τ takes an infinite outcome (or $t = 0$ if there is no such stage). Let α be the infinite outcome taken at stage t, if there is one. Let s_0 be the first stage after t at which τ is visited. Then for every stage $s > s_0$ at which τ is visited, $B_\tau(s) = B_\tau(s_0)$.

Observe that for every $\sigma \in B_\tau(s_0)$, and every $s > s_0$, $n_\sigma(s) = n_\sigma(s_0)$. For if $\sigma \in t^{<t}$ was not chosen by any strategy by stage t, then it will never be chosen, and our action for G makes $n_\sigma(s) = 0$ for every $s > t$. If instead σ is chosen by a strategy by stage t, it must be a strategy which is incomparable with $\tau^\frown k$ by the definition of $B_\tau(s_0)$. Such a strategy can never be visited at or after stage s_0, and so $n_\sigma(s)$ will not change after s_0.

For $\sigma \in B_\tau(s_0)$, we write n_σ for $\lim_s n_\sigma(s) = n_\sigma(s_0)$. By construction, for every $\sigma \in B_\tau(s_0)$, and every $F \neq \emptyset$, $\mathcal{A} \models \neg S_{n_\sigma}((F, \sigma)) \wedge \neg S_{n_\sigma - 1}((F, \sigma))$. There is some $\sigma \in B_\tau(s_0)$ which prevents τ from taking an infinite outcome after stage t. We consider the various cases.

If $f_\tau(\sigma)$ is never defined, then there is no $x \in \mathcal{M}_i$ with $W_\sigma(x)$ and $S_{n_\sigma}(x)$. On the other hand, $\mathcal{A} \models W_\sigma((\emptyset, \sigma)) \wedge S_{n_\sigma}((\emptyset, \sigma))$, so $\mathcal{A} \not\cong \mathcal{M}_i$.

If $f_\tau(\sigma)$ is defined but $\mathcal{M}_i \models \neg S_{n_\sigma}(f_\tau(\sigma))$, then it must be that $f_\tau(\sigma)$ was already defined by stage t, and thus $\sigma \in B_\tau(t)$. For if not, then $f_\tau(\sigma)$ was defined at some stage $s \geqslant s_0$, and by construction we would have $\mathcal{M}_i \models S_{n_\sigma}(f_\tau(\sigma))$. By the fact that $f_\tau(\sigma)$ is defined, we know that $\mathcal{M}_i \models W_\sigma(f_\tau(\sigma))$. As $B_\tau(0)$ is empty, we know that $t > 0$ and thus τ had infinite outcome at stage t.

$\mathcal{M}_i \models S_{n_\sigma(t)}(f_\tau(\sigma))$ by the fact that τ had an infinite outcome at stage t. Since $n_\sigma \neq n_\sigma(t)$, σ must have been chosen by a strategy ρ which is visited in the interval $[t, s)$. By the fact that $\sigma \in B_\tau(t) \subseteq t^{<t}$, we know that ρ chose σ, and thus was visited, at or before stage t. Since it is also visited at or after stage t, and τ has infinite outcome α at stage t, ρ must be comparable with $\tau^\frown \alpha$. By definition of $B_\tau(t)$, $\rho \not\subseteq \tau$. Thus $\rho \supseteq \tau^\frown \alpha$. So ρ is visited at stage t and at no stage after t.

By our action for G, it follows that $n_\sigma = n_\sigma(t) + 1$. Further, the only element $y \in \mathcal{A}$ with $\mathcal{A} \models W_\sigma(y) \wedge S_{n_\sigma(t)}(y)$ is $(\emptyset, \sigma)$, and $\mathcal{A} \models S_{n_\sigma}((\emptyset, \sigma))$. So there is no element $y \in \mathcal{A}$ with $\mathcal{A} \models W_\sigma(y) \wedge S_{n_\sigma(t)}(y) \wedge \neg S_{n_\sigma}(y)$, and so $\mathcal{A} \not\cong \mathcal{M}_i$.

Finally, suppose there is $\sigma{}^\frown j \in B_\tau(s_0)$ with $f_\tau(\sigma), f_\tau(\sigma{}^\frown j)$ both defined, and

$$\mathcal{M}_i \models W_\sigma(f_\tau(\sigma)) \wedge W_{\sigma^\frown j}(f_\tau(\sigma{}^\frown j)) \wedge S_{n_\sigma}(f_\tau(\sigma)) \wedge S_{n_{\sigma^\frown j}}(f_\tau(\sigma{}^\frown j)) \wedge \neg P(f_\tau(\sigma), f_\tau(\sigma{}^\frown j)).$$

As $\sigma{}^\frown j \in B_\tau(s_0)$, $n_{\sigma^\frown j} = n_{\sigma^\frown j}(s_0)$. So $(\emptyset, \sigma)$ and $(\emptyset, \sigma{}^\frown j)$ are the only $y_0, y_1 \in \mathcal{A}$ satisfying

$$\mathcal{A} \models W_\sigma(y_0) \wedge W_{\sigma^\frown j}(y_1) \wedge S_{n_\sigma}(y_0) \wedge S_{n_{\sigma^\frown j}}(y_1),$$

and $\mathcal{A} \models P((\emptyset, \sigma), (\emptyset, \sigma{}^\frown j))$. Thus $\mathcal{A} \not\cong \mathcal{M}_i$. $\square$

Claim 2.7. *Fix τ the M_i-strategy along the true path. If $h : \mathcal{M}_i \cong \mathcal{A}$, then for all $\sigma \notin Q$, $h \circ f_\tau(\sigma) = (\emptyset, \sigma)$.*

Proof. As $\mathcal{M}_i \cong \mathcal{A}$, τ's true outcome must be an infinite outcome. Thus every $\sigma \notin C_\tau$ is in the domain of f_τ, and in particular every $\sigma \notin Q$ is in the domain of f_τ.

Since τ takes infinite outcomes at infinitely many stages, for every $\sigma \notin Q$, $\mathcal{M}_i \models W_\sigma(f_\tau(\sigma)) \wedge S_{n_\sigma}(f_\tau(\sigma))$. For $\sigma \notin Q$, $(\emptyset, \sigma)$ is the only element of $\mathcal{A}$ satisfying both W_σ and S_{n_σ}, so for h to be an isomorphism, it must be that $h \circ f_\tau(\sigma) = (\emptyset, \sigma)$. $\square$

Claim 2.8. *Fix τ the M_i-strategy along the true path. If τ's true outcome is ∞_∞, then $\mathcal{A} \not\cong \mathcal{M}_i$.*

Proof. Let $t_0 < t_1 < \dots$ be the stages at which τ takes an infinite outcome. Then it must be that for some $\sigma \in C_\tau$, $\lim_s x_\tau(\sigma, t_s)$ does not exist. Fix such a σ. Note that if $s_0 < s_1$ and $x_\tau(\sigma, t_{s_0}), x_\tau(\sigma, t_{s_1})$ are both defined but unequal, then $x_\tau(\sigma, t_{s_0}) < x_\tau(\sigma, t_{s_1})$. This is because $x_\tau(\sigma, t_{s_1})$ is an x of the sort sought for $x_\tau(\sigma, t_{s_0})$, and if it were older it would have been chosen at stage t_{s_0}.

Let $D_\tau = \bigcup_s D_\tau(t_s)$. There is no $x \in \mathcal{M}_i$ such that $\mathcal{M}_i \models P(x, f_\tau(\sigma{}^\frown j))$ for every $\sigma{}^\frown j \in D_\tau$ – if there were, $x_\tau(\sigma, t_s)$ would converge to the least such. Observe that $D_\tau \cap Q = \emptyset$, for if a string is in Q, then it is selected by a strategy which is an initial segment of τ or which extends $\tau{}^\frown \infty_\infty$, and these strings are excluded from D_τ.

If there were an isomorphism $h : \mathcal{M}_i \cong \mathcal{A}$, then $h \circ f_\tau(\sigma{}^\frown j) = (\emptyset, \sigma{}^\frown j)$ for every $\sigma{}^\frown j \notin Q$, and in particular every $\sigma{}^\frown j \in D_\tau$, and $P((\emptyset, \sigma), (\emptyset, \sigma{}^\frown j))$ for every j. So $x = h^{-1}((\emptyset, \sigma))$ would satisfy $\mathcal{M}_i \models P(x, f_\tau(\sigma{}^\frown j))$ for every $\sigma{}^\frown j \in D_\tau$, contradicting the previous paragraph. $\square$

Claim 2.9. *If $\mathcal{A} \cong \mathcal{M}_i$, then there is a computable isomorphism between $\mathcal{A}$ and $\mathcal{M}_i$.*

Proof. Fix τ the M_i-strategy along the true path. Then τ's true outcome is ∞_k for some $k \in \omega$, and so the domain of f_τ is all strings $\sigma \in \omega^{<\omega}$ except those chosen by some strategy $\rho \subset \tau$, which is to say $\mathrm{dom}\, f_\tau = \omega^{<\omega} - C_\tau$. Note that the domain of f_τ is thus a cofinite, upwards-closed set. By assumption, there is an isomorphism $h : \mathcal{M}_i \to \mathcal{A}$. We begin by non-uniformly extending f_τ to all of $\omega^{<\omega}$. We do so recursively along C_τ.

For $\sigma \in C_\tau$, assume $f_\tau(\sigma{}^\frown j)$ is already defined for every $j \in \omega$. Let $t_0 < t_1 < \dots$ be the stages at which τ has an infinite outcome. As τ's true outcome is ∞_k, $x = \lim_s x_\tau(\sigma, t_s)$ exists. So for every $\sigma{}^\frown j \in D_\tau = \bigcup_s D_\tau(t_s)$, $\mathcal{M}_i \models P(x, f_\tau(\sigma{}^\frown j))$. Since C_τ is finite and $\tau{}^\frown \infty_\infty$ is visited only finitely many times, there are only

finitely many j with $\sigma^\frown j \notin D_\tau$. Let $J = \{j : \mathcal{M}_i \models \neg P(x, f_\tau(\sigma^\frown j))\}$, and note that J is finite. Informally, we let $f_\tau(\sigma) = x \triangle J$. More precisely, if $h(x) = (F, \sigma)$, we let $f_\tau(\sigma) = h^{-1}((F \triangle J, \sigma))$.

Having extended f_τ to a total computable function (albeit non-uniformly), let H_σ be such that $h(f_\tau(\sigma)) = (H_\sigma, \sigma)$, for each $\sigma \in \omega^{<\omega}$.

Claim 2.9.1. *For all σ and j, $\mathcal{M}_i \models P(f_\tau(\sigma), f_\tau(\sigma^\frown j))$.*

Proof. If $\sigma \notin C_\tau$, then $\sigma^\frown j \notin C_\tau$, and so $f_\tau(\sigma), f_\tau(\sigma^\frown j)$ were defined by our action for τ, and since τ's true outcome is an infinite outcome, $\mathcal{M}_i \models P(f_\tau(\sigma), f_\tau(\sigma^\frown j))$.

If instead $\sigma \in C_\tau$, then this is by construction. Consider the $x = \lim_s x_\tau(\sigma, t_s)$ and the set J used in the definition of $f_\tau(\sigma)$. If $j \notin J$, then $\mathcal{M}_i \models P(x, f_\tau(\sigma^\frown j))$. So $\mathcal{A} \models P(h(x), (H_{\sigma^\frown j}, \sigma^\frown j))$. By construction, $h(x) = (H_\sigma \triangle J, \sigma)$, and since $j \notin J$, $j \in H_\sigma$ iff $j \in H_\sigma \triangle J$, so $\mathcal{A} \models P((H_\sigma, \sigma), (H_{\sigma^\frown j}, \sigma^\frown j))$, and thus $\mathcal{M}_i \models P(f_\tau(\sigma), f_\tau(\sigma^\frown j))$.

If instead $j \in J$, then $\mathcal{M}_i \models \neg P(x, f_\tau(\sigma^\frown j))$. So $\mathcal{A} \models \neg P(h(x), (H_{\sigma^\frown j}, \sigma^\frown j))$. Since $j \in J$, $\mathcal{A} \models P((H_\sigma, \sigma), (H_{\sigma^\frown j}, \sigma^\frown j))$, and thus $\mathcal{M}_i \models P(f_\tau(\sigma), f_\tau(\sigma^\frown j))$. $\square$

It follows that for every σ and j, $j \in H_\sigma$ iff $|H_{\sigma^\frown j}|$ is odd.

We now define a computable isomorphism $g : \mathcal{A} \to \mathcal{M}_i$. We begin by defining $g((\emptyset, \sigma)) = f_\tau(\sigma)$ for every $\sigma \in \omega^{<\omega}$. We then extend recursively. For each $F \in [\omega]^{<\omega}$ with $|F| > 0$, we search for G, j and y with $F = G \cup \{j\}$, $g((G, \sigma))$ already defined, and $\mathcal{M}_i \models E_j(g((G, \sigma)), y)$, and define $g((F, \sigma)) = y$ for the first such found.

To show that g is an isomorphism, we will show that $h \circ g$ is an automorphism of $\mathcal{A}$.

Claim 2.9.2. *For each $(F, \sigma) \in \mathcal{A}$, $h \circ g((F, \sigma)) = (F \triangle H_\sigma, \sigma)$.*

Proof. We argue by induction on $|F|$. The base case $|F| = 0$ is immediate.

For $|F| > 0$, there was some $j \in F$ such that $g((F, \sigma))$ was defined based on $G = F - \{j\}$. Then $\mathcal{M}_i \models E_j(g((G, \sigma)), g((F, \sigma)))$, and so

$$\mathcal{A} \models E_j(h \circ g((G, \sigma)), h \circ g((F, \sigma))) \implies h \circ g((G, \sigma)) \triangle h \circ g((F, \sigma)) = \{j\}$$
$$\implies (G \triangle H_\sigma) \triangle h \circ g((F, \sigma)) = \{j\}$$
$$\implies (F \triangle \{j\}) \triangle H_\sigma \triangle h \circ g((F, \sigma)) = \{j\}$$
$$\implies h \circ g((F, \sigma)) = F \triangle H_\sigma. \qquad \square$$

Claim 2.9.3. *$h \circ g$ is an automorphism of $\mathcal{A}$.*

Proof. It follows from the previous claim that $h \circ g$ respects all the relations E_j.

As $j \in H_\sigma$ iff $|H_{\sigma^\frown j}|$ is odd, $h \circ g$ respects the P relation.

If $\sigma \in Q$, then $S_n((F, \sigma))$ for every $F \in [\omega]^{<\omega}$ and $n \in \omega$, and so $h \circ g$ trivially respects the S_n relations on elements of the form (F, σ).

If $\sigma \notin Q$, then by construction, $\mathcal{M}_i \models W_\sigma(f_\tau(\sigma)) \wedge S_{n_\sigma}(f_\tau(\sigma))$. Further, $(\emptyset, \sigma)$ is the only element of $\mathcal{A}$ satisfying this formula, so it must be that $h \circ f_\tau(\sigma) = (\emptyset, \sigma)$. Thus $H_\sigma = \emptyset$, and so $h \circ g$ fixes elements of the form (F, σ), and thus respects the S_n relations on them. $\square$

So $h \circ g$ is an automorphism of $\mathcal{A}$, and thus g witnesses that $\mathcal{A}$ and $\mathcal{M}_i$ are computably isomorphic. $\square$

This completes the proof of Theorem 2.

The computably categorical structure $\mathcal{A}$ constructed in the proof of Theorem 2 has the property that it is no longer computably categorical after a constant is added naming $(\emptyset, \langle\rangle)$. For fix any i such that $[i] \cap [Q] \neq \emptyset$. Then, as shown in claim 2.2, $(\mathcal{A}, (\emptyset, \langle\rangle)) \cong (\mathcal{A}, (\{i\}, \langle\rangle))$, but computing an isomorphism requires computing a path through Q, and Q has no hyperarithmetic paths.

Indeed, for any distinct i_0, i_1, there is no computable isomorphism between $(\mathcal{A}, (\{i_0\}, \langle\rangle))$ and $(\mathcal{A}, (\{i_1\}, \langle\rangle))$. Thus $(\mathcal{A}, (\emptyset, \langle\rangle))$ has infinite computable dimension. A structure with this behavior was first constructed by Hirschfeldt, Khoussainov and Shore [HKS03].

Corollary 3. *There is a structure $\mathcal{B}_0$ of computable dimension 2 with two computable copies which are not hyperarithmetically isomorphic.*

Proof. Let $\mathcal{A}$ be the structure from the proof of Theorem 2. Our structure $\mathcal{B}_0$ will consist of $\mathcal{A}$ along with two new elements, a_{even} and a_{odd}, and a new constant symbol c. We make the following additional definitions:

- $\mathcal{B}_0 \models P(a_{\mathrm{even}}, (F, \langle\rangle))$ precisely when $|F|$ is even;
- $\mathcal{B}_0 \models P(a_{\mathrm{odd}}, (F, \langle\rangle))$ precisely when $|F|$ is odd;
- $\mathcal{B}_0 \models c = a_{\mathrm{even}}$;

and no other relations hold involving a_{even} or a_{odd}.

Let $\mathcal{B}_1$ be defined identically to $\mathcal{B}_0$, excepting that $\mathcal{B}_1 \models c = a_{\mathrm{odd}}$. Let $\hat{\mathcal{B}}$ be the structure obtained from $\mathcal{B}_0$ (equivalently, from $\mathcal{B}_1$) by removing the constant symbol c from the language.

For any computable $\hat{\mathcal{C}} \cong \hat{\mathcal{B}}$, if we nonuniformly remove the copies of a_{even} and a_{odd}, then we have a computable copy of $\mathcal{A}$. As $\mathcal{A}$ is computably categorical, there is a computable isomorphism f from this smaller structure to $\mathcal{A}$, and this can be extended to a computable isomorphism from $\hat{\mathcal{C}}$ to $\hat{\mathcal{B}}$.

So for any $\mathcal{C} \cong \mathcal{B}_0$, the computable isomorphism from $\hat{\mathcal{C}}$ to $\hat{B}$ is either an isomorphism from $\mathcal{C}$ to $\mathcal{B}_0$ or from $\mathcal{C}$ to $\mathcal{B}_1$. However, any isomorphism from $\mathcal{B}_0$ to $\mathcal{B}_1$ must restrict to an automorphism of $\mathcal{A}$ sending $(\emptyset, \langle\rangle)$ to some $(F, \langle\rangle)$ with $|F|$ odd, and by Claim 2.2 and choice of computable tree T, such an isomorphism must not be hyperarithmetic.   □

We are grateful to Rod Downey for asking whether our methods could address the previous question.

3. A structure with degree of categoricity and infinite spectral dimension

This section is devoted to the proof of the following theorem.

Theorem 4. *There is a computable structure with degree of categoricity $\mathbf{0}''$, but with infinite spectral dimension.*

We will first build a computable structure $\mathcal{A}$ and closed sets $Q, R \subseteq \omega^\omega$ satisfying all of the following:

- $\mathcal{A}$ is computably categorical.
- $X = \{i : [i] \cap Q \neq \emptyset\}$ is infinite.
- For all $i \in X$, $|[i] \cap Q| = 1$. We refer to the unique element of $[i] \cap Q$ as f_i.
- $Y = \{j : [j] \cap R \neq \emptyset\}$ is infinite.

- For all $j \in Y$, $|[j] \cap R| = 1$. We refer to the unique element of $[j] \cap R$ as g_j.
- Each f_i and g_j is Δ_3^0.
- For any $i \in X$ and any $j \in Y$ with $i < j$, $f_i + g_j$ computes $\emptyset''$.
- For any $j_0, \ldots, j_k \in Y$ and any $i \in X$ with $i > \max\{j_0, \ldots, j_k\}$, $f_i \oplus g_{j_0} \oplus \cdots \oplus g_{j_k}$ does not compute $\emptyset'$.
- There are elements $\{u_0, u_1\}$ which comprise an orbit of $\mathcal{A}$. Computing an automorphism sending u_0 to u_1 exactly corresponds to computing an element of Q (i.e. the tasks are Muchnik-equivalent).
- There are elements $\{v_F : F \in [Y]^{<\omega}\}$ which comprise an orbit of $\mathcal{A}$. Computing an automorphism sending v_F to v_G (and fixing u_0) exactly corresponds to computing $\bigoplus_{j \in F \triangle G} g_j$.

Before we proceed with the construction, we explain why this will suffice. Our final structure will be $(\mathcal{A}, c, d)$, where c and d are constants naming u_0 and $v_\emptyset$. Suppose $\mathcal{B}$ is another computable presentation of $(\mathcal{A}, c, d)$. As $\mathcal{A}$ is computably categorical, we may assume that $\mathcal{B}$ has the form $(\mathcal{A}, u_\ell, v_F)$ for $\ell \in \{0, 1\}$, $F \in [Y]^{<\omega}$. Then, for any $i \in X$, $f_i \oplus \bigoplus_{j \in F} g_j$ computes an isomorphism from $(\mathcal{A}, c, d)$ to $\mathcal{B}$, and in particular $\mathbf{0}''$ does.

If $\mathcal{B}_0, \ldots, \mathcal{B}_{n-1}$ are all computable presentations of $(\mathcal{A}, c, d)$, then we may assume that $\mathcal{B}_k$ is of the form $(\mathcal{A}, u_{\ell_k}, v_{F_k})$. Then, for any $i \in X$, $f_i \oplus \bigoplus_{j \in \bigcup_{k<n} F_k} g_j$ computes isomorphisms between all of these structures, and if i is chosen sufficiently largely, it does not compute $\emptyset'$.

On the other hand, suppose $\mathbf{d}$ can compute an isomorphism between any two computable presentations of $(\mathcal{A}, c, d)$. Then it computes an isomorphism between $(\mathcal{A}, c, d)$ and $(\mathcal{A}, u_1, d)$, and thus must compute some $f_i \in Q$. Fix $j > i$. Then $\mathbf{d}$ must also compute an isomorphism between $(\mathcal{A}, c, d)$ and $(\mathcal{A}, c, \{j\})$, and thus must compute g_j. Since $f_i + g_j$ computes $\emptyset''$, $\mathbf{d} \geqslant \mathbf{0}''$.

The structure. Our structure will have $(W_\sigma^Q)_{\sigma \in \omega^{<\omega}}, (W_\sigma^R)_{\sigma \in \omega^{<\omega}}$ all unary relations, $(E_i)_{i \in \omega}$ all binary relations, P a binary relation, and $(S_n)_{n \in \omega}$ all unary relations. As in the previous construction, the S_n will only be declared in a c.e. fashion. Our structure's universe will be $([\omega]^{<\omega} \times \omega^{<\omega} \times \{0, 1\}) \sqcup \{u_0, u_1\}$.

With the exception of the S_n, we can give a complete description of our structure immediately. For $(F, \tau, a), (G, \rho, b) \in [\omega]^{<\omega} \times \omega^{<\omega} \times \{0, 1\}$:

- $W_\sigma^Q((F, \tau, a))$ holds iff $\sigma = \tau$ and $a = 0$.
- $W_\sigma^R((F, \tau, a))$ holds iff $\sigma = \tau$ and $a = 1$.
- $E_i((F, \tau, a), (G, \rho, b))$ holds iff $\tau = \rho$, $a = b$ and $F \triangle G = \{i\}$.
- $P((F, \tau, a), (G, \rho, b))$ holds iff $a = b$, $\rho = \tau^\frown i$ for some i, and one of the following holds:
    - $i \notin F$ and $|G|$ is even; or
    - $i \in F$ and $|G|$ is odd.
- For $k < 2$, $P(u_k, (F, \langle\rangle, 0))$ holds iff $k + |F|$ is even.
- No other relations involving u_0 or u_1 hold.

As before, we have the affine space $[\omega]^{<\omega}$, but this time we have two copies of $\omega^{<\omega}$, and we have associated a copy of the space to each string in either copy. The W_σ^Q and W_σ^R let us identify which copy a given element belongs to.

As mentioned, during the course of the construction we will build closed sets $Q, R \subseteq \omega^\omega$. Let $T_0, T_1 \subseteq \omega^{<\omega}$ be the minimal trees such that $[T_0] = Q$, $[T_1] = R$, i.e. $T_0 = \{\sigma \in \omega^{<\omega} : [\sigma] \cap Q \neq \emptyset\}$, and similarly for T_1. For $a < 2$ and $\sigma \in T_a$, $S_n(F, \sigma, a)$ will hold for every n and every $F \in [\omega]^{<\omega}$. For $\sigma \notin T_a$, there will be an n such that $S_n(F, \sigma, a)$ holds iff $F = \emptyset$. The purpose of this setup is the following.

Claim 4.1. *The automorphisms of $\mathcal{A}$ which send $(\emptyset, \sigma, 1)$ to $(\{i\}, \sigma, 1)$ are Muchnik equivalent to $[\sigma^\frown i] \cap R$. More generally, the automorphisms of $\mathcal{A}$ which send $(\emptyset, \sigma, 1)$ to $(F, \sigma, 1)$ are Muchnik equivalent to the join $\bigoplus_{i \in F}([\sigma^\frown i] \cap R)$.*

The automorphisms which send u_0 to u_1 are Muchnik equivalent to Q.

In particular, such an automorphism exists iff the appropriate set is nonempty.

Proof. As the proof of claim 2.2. $\qquad\square$

Let $v_F = (F, \langle\rangle, 1)$ for $F \in [Y]^{<\omega}$. These, along with u_0, u_1, are the previously promised elements.

Construction. Fix $(\mathcal{M}_i)_{i \in \omega}$ a listing of all partial computable structures in our language. Fix also Z a set which is complete for Π^0_2, and with the property that any enumeration of Z computes Z. For example, $Z = \overline{\emptyset'} \oplus \overline{\emptyset''}$. Fix a computable predicate $\varphi(n, s)$ such that $n \in Z \iff \exists^\infty s\, \varphi(n, s)$. For $i < j$, our intention is that $f_i + g_j$ is a "modulus for Σ^0_2-ness":

$$\text{for } n > j,\ \exists^\infty s\, \varphi(n, s) \iff \exists s > (f_i + g_j)(n)\, \varphi(n, s).$$

This will ensure that $f_i + g_j$ enumerates Z and thus computes Z.

Again, we perform a Π^0_2 priority construction on a tree of strategies. We have strategies of type N^a_r, for $a < 2$ and $r \in \omega$, which are mother strategies which begin an f_i or g_i with $i > r$, depending on whether $a = 0$ or 1. We have strategies of type $N^a_{r,n}$ for $a < 2$, $r \in \omega$ and $n > 0$, which are daughter strategies responsible for defining $f_i(n)$ or $g_j(n)$ and ensuring that $\exists^\infty s\, \varphi(n, s) \iff \exists s > (f_i + g_j)(n)\, \varphi(n, s)$, where either f_i or g_j is the path begun by the mother node. We have strategies of type $U_{i,e}$, for $i, e \in \omega$, which are responsible for ensuring that $f_i \oplus g_0 \oplus \cdots \oplus g_{i-1}$ does not compute $\emptyset'$ via functional Φ_e. We have strategies of type M_i, for $i \in \omega$, which are responsible for constructing a computable isomorphism between $\mathcal{A}$ and $\mathcal{M}_i$ when $\mathcal{A} \cong \mathcal{M}_i$. Finally, we have a global strategy G which is responsible for ensuring that the S_n are as described.

We arrange the N^a_r, $N^a_{r,n}$, $U_{i,e}$ and M_i in some priority ordering of type ω, such that N^a_r occurs in the ordering before any $N^a_{r,n}$, and each $N^a_{r,n}$ occurs before $N^a_{r,n+1}$. The set of possible outcomes for an M_i-strategy is again $\{k, \infty_k : k \in \omega\} \cup \{\infty_\infty\}$. N^a_r-strategies have only a single outcome: `outcome`. The set of possible outcomes for an $N^a_{r,n}$-strategy is $\{k : k \in \omega\} \cup \{\infty\}$. The set of possible outcomes for a $U_{i,e}$-strategy is $\{0, 1\}$.

We will construct a priority tree of these strategies, but this tree will not be the simplest tree, where each level is devoted to a given strategy type according to the priority ordering. Instead, we will need to dynamically construct our tree during the construction. We will say more about this later.

Auxiliary functions. During the construction, we will be defining various f_i and g_j. Each will be begun by a mother node of some N^a_r-type, and then will be continued by nodes of $N^a_{r,n}$-type but also by nodes of $U_{i,e}$-type. The $N^a_{r,n}$-nodes will inherit a string from an ancestor strategy which they will then extend, while

the $U_{i,e}$-nodes may steal strings from descendant strategies. To track these strings, we will define an auxiliary function σ.

For τ an N_r^a-strategy, $\sigma(\tau) = \langle i \rangle$ will be the beginning of the f_i or g_i that τ creates. For τ an $N_{r,n}^a$-strategy and β an outcome of τ, $\sigma(\tau, \beta)$ will be the string which τ provides to strategies extending $\tau^\frown \beta$.

Finally, for τ a $U_{i,e}$-strategy, if τ takes outcome 1, we will define $\sigma(\tau, \theta)$ for each $\theta \subset \tau$ such that θ is a mother node beginning f_i, or θ is a mother node beginning g_j with $j < i$. These will be the strings which $N_{r,n}^a$-strategies extending $\tau^\frown 1$ inherit, if they are daughters of such a θ.

Also, for τ an N_r^a-strategy, we will define $v(\tau) = i$ for the i such that τ is beginning f_i or g_i.

Finally, if a $U_{i,e}$-strategy τ steals strings from its descendants, we will define $\ell(\tau)$ to be the length of the longest stolen string. This will be used in building the priority tree, later.

Choosing a pair. During the construction, pairs $(\sigma, a) \in \omega^{<\omega} \times \{0, 1\}$ will be chosen by strategies; we will say that τ *chooses* (σ, a) when it declares that (σ, a) is growing (see below). This is similar to strategies choosing strings in the previous construction. However, unlike the previous construction, a strategy may choose more than one pair. Also, a pair may be chosen by a strategy and then later chosen by another strategy. When this happens, the first strategy will never again be visited, and so we may think of it as the pair being transferred from the first strategy to the second. As in the previous strategy, if a pair (σ, a) with $\sigma \in s^{<s}$ has not been chosen by any strategy by the start of stage s, it will never be chosen by any strategy. Unlike the previous construction, the pairs $(\langle \rangle, a)$ are never chosen, and so we will handle them explicitly in G.

Growing a pair. At a stage s, a strategy may declare that the pair (σ, a) is growing. When this happens, we let m be largest such that we have already declared $S_m((\emptyset, \sigma, a))$ (or $m = -1$ if no such declarations have been made). We declare $S_k((\emptyset, \sigma, a))$ for $0 \leqslant k < m + 2$. We also declare $S_k((F, \sigma, a))$ for every $F \subseteq s$ and $0 \leqslant k < m$.

Strategy for G. Our action is much the same as the action for the global strategy in Theorem 2. At the end of stage s, for every $\sigma \in s^{<s}$ and every $a < 2$, declare $S_0((\emptyset, \sigma))$. Also, at the end of every stage, grow the pairs $(\langle \rangle, 0)$ and $(\langle \rangle, 1)$.

Strategy for N_r^a. Suppose τ is an N_r^a-strategy. At the first stage when τ is visited, an $i \in \omega$ larger than any number previously mentioned in the construction is chosen. τ makes the definitions $\sigma(\tau) = \langle i \rangle$, $v(\tau) = i$.

At every stage when τ is visited, grow the pair $(\langle i \rangle, a)$.

τ always takes the outcome `outcome`.

Strategy for $N_{r,n}^a$. Suppose τ is an $N_{r,n}^a$-strategy visited at stage s. Let θ be the mother node for τ, i.e. the unique $\theta \subset \tau$ which is an N_r^a-strategy. Let $\rho \subset \tau$ be largest such that one of the following holds:

- $\rho = \theta$. In this case, let $\gamma = \sigma(\theta)$.
- ρ is an $N_{r,n-1}^a$-strategy. In this case, let α be the outcome such that $\rho^\frown \alpha \subseteq \tau$, and let $\gamma = \sigma(\rho, \alpha)$.
- $a = 0$ and ρ is a $U_{i,e}$-strategy with $i = v(\theta)$. In this case, let $\gamma = \sigma(\rho, \theta)$.
- $a = 1$ and ρ is a $U_{i,e}$-strategy with $i > v(\theta)$. In this case, let $\gamma = \sigma(\rho, \theta)$.

We will argue in the verification that $|\gamma| = n$.

Let k be the number of times τ has previously taken outcome ∞, and let $t < s$ be the last stage at which τ took outcome ∞ (or $t = 0$ if there is no such stage). If there is a $q \in [t, s)$ such that $\varphi(n, q)$ holds, we will take outcome ∞ at stage s; otherwise, we will take outcome k.

Regardless, let β be the outcome we are taking at stage s. If $\sigma(\tau, \beta)$ is not already defined, we define $\sigma(\tau, \beta) = \gamma^\frown s$. We grow the pair $(\sigma(\tau, \beta), a)$ and then take outcome β.

Strategy for $U_{i,e}$. Suppose τ is a $U_{i,e}$-strategy visited at stage s. If there is no $\theta \subset \tau$ an N_r^0-strategy with $v(\theta) = i$, then τ never acts and simply takes outcome 0 at every stage.

Suppose instead there is such a θ. Let

$$C_\tau = \{\theta\} \cup \{\psi \subset \tau : \psi \text{ is an } N_r^1\text{-strategy with } v(\psi) < i\}.$$

By the recursion theorem, there is some $x_\tau \in \omega$ such that we can control the enumeration of x_τ into $\emptyset'$. We initially keep x_τ out of $\emptyset'$.

At stage s, we check if there is a set $\{\pi_\psi, \alpha_\psi : \psi \in C_\tau\}$ such that:

- Each π_ψ extends $\tau^\frown 0$ and is a daughter of ψ;
- Each α_ψ is an outcome of π_ψ;
- $\sigma(\pi_\psi, \alpha_\psi)$ is defined for every ψ; and
- $\Phi_{e,s}(\bigoplus_{\psi \in C_\tau} \sigma(\pi_\psi, \alpha_\psi); x_\tau) \downarrow = 0$.

If there is no such set, τ takes outcome 0.

If there is, we fix such a set $\{\pi_\psi, \alpha_\psi : \psi \in C_\tau\}$. We will henceforth always take outcome 1, and we will always use this set. We enumerate x_τ into $\emptyset'$ and define $\sigma(\tau, \psi) = \sigma(\pi_\psi, \alpha_\psi)$ for each $\psi \in C_\tau$. We also define $\ell(\tau) = \max\{|\sigma(\tau, \psi)| : \psi \in C_\tau\}$.

If τ is being visited at stage s after having fixed a set, we grow $(\sigma(\tau, \theta), 0)$ and $(\sigma(\tau, \psi), 1)$ for $\psi \in C_\tau - \{\theta\}$, and then take outcome 1.

Notation. For notational convenience, if τ is an N_r^a-strategy, we define $a(\tau) = a$.

Strategy for M_i. This is mostly as in the proof of Theorem 2, with the obvious changes for pairs $(\sigma, a) \in \omega^{<\omega} \times \{0, 1\}$ rather than strings $\sigma \in \omega^{<\omega}$.

C_τ is defined to consist of those pairs (σ, a) such that there is a $\rho \subset \tau$ for which one of the following holds:

- ρ is an N_r^a-strategy and $\sigma = \sigma(\tau)$.
- ρ is an $N_{r,n}^a$-strategy, $\rho^\frown \alpha \subseteq \tau$ and $\sigma = \sigma(\rho, \alpha)$.
- ρ is a $U_{i,e}$-strategy with $\rho^\frown 1 \subseteq \tau$, and there is $\psi \in C_\rho$ with $a = a(\psi)$ and $\sigma = \sigma(\rho, \psi)$.

It is important to note that this is finite and is completely determined by the first stage at which τ is visited.

If $t < s$ is the last stage at which τ took an infinite outcome and k_0 is the number of times τ has taken an infinite outcome, $B_\tau(s)$ consists of every pair $(\sigma, a) \in t^{<t} \times \{0, 1\}$ except for those in C_τ and those chosen by a strategy extending $\tau^\frown k_0$.

Otherwise, this is as in the proof of Theorem 2.

Some auxiliary definitions. For a mother strategy ψ and a $\tau \supset \psi$, we let $n(\tau, \psi)$ be the largest n such that ψ has an $N_{r,n}^a$-daughter strategy along τ, or $n(\tau, \psi) = 0$ if there is no such n.

Suppose ψ is an N_r^a-strategy, and $\tau \supset \psi$ is a $U_{i,e}$-strategy with $\psi \in C_\tau$. For every n with $n(\tau, \psi) < n < |\sigma(\tau, \psi)|$, τ has defined $f_i(n)$ or $g_j(n)$ for the f_i or g_j started by ψ. In this case, we say that τ *blocks* $N_{r,n}^a$. Observe that a given τ can only block finitely many $N_{r,n}^a$.

Running the construction and the priority tree. As mentioned, our priority tree will be constructed dynamically. Our reason for doing this is that a $U_{i,e}$-strategy along the true path may define f_i on some interval I. To maintain the desired property of $f_i + g_j$, we must ensure that $g_j(x)$ is defined by N-strategies, for every $j > i$ with $j < x$ and $x \in I$. So the appropriate N-strategies must occur on the true path before the next U-strategy. Similar concerns about the definition of f_i apply. Also, if a $U_{i,e}$-strategy defines $f_i(n)$, then we cannot have an N-strategy which seeks to define $f_i(n)$.

At stage s, we begin by visiting the strategy $\langle \rangle$. After having visited a strategy τ at stage s with $|\tau| < s$, we let α be the outcome taken by τ at this stage. We next visit the strategy $\tau^\frown \alpha$. After having visited a strategy τ with $|\tau| = s$, we run the global strategy and end the stage.

The first time a given node on the priority tree is visited, we will determine what type of strategy it is. This will determine what outcomes it has, and thus what children it has on the priority tree. But we will not decide what type each of the children is until they are visited.

Suppose we are visiting node τ for the first time. We make τ a ζ-strategy, where ζ is least in the priority ordering subject to the following constraints:

- There can be no $\rho \subset \tau$ which is a ζ-strategy;
- τ cannot be an $N_{r,n}^a$-strategy if there is a $U_{i,e}$-strategy ρ with $\rho^\frown 1 \subseteq \tau$ and ρ blocks $N_{r,n}^a$.
- If there is a $U_{i,e}$-strategy ρ with $\rho^\frown 1 \subseteq \tau$, and $\psi \subset \rho$ is a mother node, then τ cannot be a $U_{i',e'}$-strategy if this would make $n(\tau, \psi) \leqslant \ell(\rho)$.

We put the lexicographic ordering on the tree of strategies, where we order the outcomes of M_i as $\infty_\infty < \cdots < \infty_2 < \infty_1 < \infty_0 < \cdots < 2 < 1 < 0$, we order the outcomes of $N_{r,n}^a$ as $\infty < \cdots < 2 < 1 < 0$, and we order the outcomes of $U_{i,e}$ as $1 < 0$. Observe that our construction has the property that if a strategy τ is visited, and then at a later stage a strategy ρ which is lexicographically to the left of τ is visited, τ can never again be visited.

The true path. We define the true path recursively, maintaining the inductive hypothesis that every strategy on the true path is visited infinitely often. $\langle \rangle$ is on the true path. If τ is on the true path, let α be the leftmost outcome which τ takes infinitely often. By our actions for the various strategies, this necessarily exists. Then $\tau^\frown \alpha$ is on the true path. We call α the *true outcome* of τ.

We let TP denote the true path.

By our earlier observation, no strategy off the true path is visited infinitely often. Observe that the TP is computable from $\mathbf{0}''$.

Verification. We begin by showing that our dynamic construction did not omit any strategies we did not intend to. We say that $N_{r,n}^a$ is *blocked on the true path* if there is a $U_{i,e}$-strategy ρ with $\rho^\frown 1 \in TP$ and ρ blocks $N_{r,n}^a$.

Claim 4.2. *For every strategy type ζ which is not blocked on the true path, there is a ζ-strategy on the true path.*

Proof. Towards a contradiction, let ζ be the least type in the priority ordering which does not occur on the true path and is not blocked. Fix ρ on the true path such that all types before ζ occur along ρ. If ζ is not of type $U_{i,e}$, then since ζ is not blocked, the next strategy along the true path must be of type ζ.

If instead ζ is of type $U_{i,e}$, then since there is no ζ-strategy on the true path, there can be no $U_{i',e'}$-strategy on the true path for any $U_{i',e'}$ later in the priority ordering than ζ. So no strategy along the true path extending ρ will have type $U_{i',e'}$. So only finitely many $N_{r,n}^a$ are blocked. So for sufficiently long τ on the true path, the relevant $n(\tau, \psi)$ will have grown sufficiently large that ζ is no longer constrained from being chosen for the next strategy, contrary to our choice of ζ. $\square$

Claim 4.3. *If τ is an $N_{r,n}^a$-strategy on the tree of strategies, then the γ defined in the action for τ has length n.*

Proof. By induction on n and our action for the $N_{r,n}^a$-strategies, as well as our priority ordering and the dynamic construction of the tree of strategies. $\square$

Claim 4.4. *If ψ is a mother node on the true path, then ψ has infinitely many daughter nodes on the true path.*

Proof. By our dynamic construction of the tree of strategies. Note that we are not allowed to select a $U_{i,e}$-strategy until $n(\tau, \psi) > \ell(\rho)$ for every mother node $\psi \subset \rho$. $\square$

Let X be the set of $v(\psi)$, for ψ an N_r^0-strategy along the true path. Let Y be the set of $v(\psi)$, for ψ an N_r^1-strategy along the true path. For ψ an N_r^0-strategy along the true path with $v(\psi) = i$, define $f_i = \bigcup \sigma(\tau, \alpha)$, where the union ranges over $\tau \widehat{\ } \alpha \in TP$ with τ a daughter of ψ. Similarly, for ψ an N_r^1-strategy along the true path with $v(\psi) = j$, define $g_j = \bigcup \sigma(\tau, \alpha)$, where the union ranges over $\tau \widehat{\ } \alpha \in TP$ with τ a daughter of ψ. By the previous claim, these are elements of ω^ω. Further, as TP is $\mathbf{0}''$-computable, so are f_i and g_j, for $i \in X$, $j \in Y$.

Define $T_0 = \{\sigma : \exists i \in X \; \sigma \subset f_i\}$ and $T_1 = \{\sigma : \exists j \in Y \; \sigma \subset g_j\}$. The argument that the relationship of the S_n to T_0 and T_1 is as previously described is the same as the corresponding argument in the proof of Theorem 2.

The argument that $\mathcal{A}$ is computably categorical is also as in the proof of Theorem 2. The reader might worry that the possibility for a string σ to be chosen by multiple strategies could cause trouble. However, note that the only way in which σ can be chosen by multiple strategies is if it is chosen first by some strategy π, and then later chosen by a $U_{i,e}$-strategy ρ with $\rho \widehat{\ } 0 \subseteq \pi$. If τ is an $\mathcal{M}_i$-strategy, then τ's action does not depend on precisely which strategy has chosen σ—rather, it only depends on which outcome of τ's is extended by the strategy that has chosen σ. If τ is on the true path, and some strategy π extending $\tau \widehat{\ } \alpha$ has chosen σ, then every strategy which chooses σ extends $\tau \widehat{\ } \alpha$, and so τ is untroubled by this behavior.

It remains only to show that the f_i and g_j behave as we would like them to.

Claim 4.5. *For $i < j < n$ with $i \in X$ and $j \in Y$, at most one of $f_i(n)$ and $g_j(n)$ is defined by some $U_{i,e}$-strategy ρ with $\rho \widehat{\ } 1$ along the true path.*

Proof. Suppose $f_i(n)$ is defined before $g_j(n)$ on the true path, and some ρ a $U_{i,e}$-strategy with $\rho \widehat{\ } 1$ along the true path defines $f_i(n)$. Let ψ be the N_r^1-strategy with $v(\psi) = j$ and $\psi \in TP$. It cannot be that $\rho \widehat{\ } 1 \subseteq \psi$, because then ψ would have

chosen $j > n$. So $\psi \subset \rho$, and thus a daughter $N^1_{r,n}$-strategy is guaranteed to occur on the true path by our dynamic construction of the tree of strategies.

The argument for the case where $g_j(n)$ is first defined is the same.   □

To show that $Z \leqslant_T f_i + g_j$ for each $i < j$, we need only show the following:

Claim 4.6. *For all $i < j < n$ with $i \in X$ and $j \in Y$, if $n \notin Z$, then $\neg \exists s > (f_i + g_j)(n)\, \varphi(n,s)$.*

Proof. By the previous claim, without loss of generality $f_i(n)$ is defined by some $N^0_{r,n}$-strategy. Let τ be the strategy along the true path defining $f_i(n)$. Let s_0 be largest such that $\varphi(n,s_0)$. Then τ has outcome k for some $k \in \omega$, and it first takes outcome k at some stage $s_1 > s_0$. So $f_i(n)$ is defined to be some $s \geqslant s_1$, by construction.   □

Claim 4.7. *For all $e \in \omega$ and $i \in X$, if $j_0 < j_1 < \ldots j_{k-1}$ are the elements of Y less than i, then $\emptyset' \neq \Phi_e(f_i \oplus g_{j_0} \oplus \cdots \oplus g_{j_{k-1}})$.*

Proof. Fix τ the $U_{i,e}$-strategy along the true path. Since N^a_r-strategies choose their number large, $i \in X$ implies there is a $\theta \in C_\tau$ with $a(\theta) = 0$, $v(\theta) = i$. Similarly, for each $m < k$ there is a $\psi_m \in C_\tau$ with $a(\psi_m) = 1$, $v(\psi_m) = j_m$.

If τ has true outcome 0, then $x_\tau \notin \emptyset'$, and

$$\neg[\Phi_e(f_i \oplus g_{j_0} \oplus \cdots \oplus g_{j_{k-1}}; x_\tau)\!\downarrow = 0],$$

as otherwise there would be some collection of sufficiently large $\pi_\psi \supseteq \tau{}^\frown 0$ and α_ψ for $\psi \in C_\tau$ with $\pi_\psi{}^\frown \alpha_\psi$ along the true path such that $\Phi_e(\bigoplus_{\psi \in C_\tau} \sigma(\pi_\psi, \alpha_\psi); x_\tau)\!\downarrow = 0$, and τ would instead have true outcome 1.

If τ has true outcome 1, then $x_\tau \in \emptyset'$, $f_i \supset \sigma(\tau, \theta)$ and $g_{j_m} \supset \sigma(\tau, \psi_m)$ for $m < k$, and

$$\Phi_e(\sigma(\tau, \theta) \oplus \sigma(\tau, \psi_0) \oplus \cdots \oplus \sigma(\tau, \psi_{k-1}); x_\tau)\!\downarrow = 0$$

by construction.   □

This completes the proof of Theorem 4.

4. Closing thoughts

In the construction for Theorem 2, given a computable tree T, we constructed a computably categorical structure and a tree Q which is Δ^0_3-isomorphic to T such that $[Q]$ was coded into the automorphism group of our structure in a particular fashion. The reason for T and Q being Δ^0_3-isomorphic rather than computably isomorphic is because constructing the isomorphism requires the true path of the construction, and the computable categoricity machinery we used requires a Δ^0_3 true path. If there were some way to avoid this obstacle and make T and Q computably isomorphic (or even equal), then these methods would be applicable to a wide range of questions. For example, we would immediately be able to conclude that lowness for categoricity and lowness for isomorphism are the same.

Similarly, the reason $\mathbf{0}''$ was chosen as the degree of categoricity in Theorem 4 was because we again required a Δ^0_3 true path for the computable categoricity machinery, and the true path was necessary to compute the various f_i and g_j. We suspect that, using standard techniques, this could be changed to any Σ^0_2 degree strictly above $\mathbf{0}'$. We do not know, however, whether the same result could be repeated with $\mathbf{0}'$.

References

[AC16]   Bernard Anderson and Barbara Csima. Degrees that are not degrees of categoricity.
            Notre Dame J. Form. Log., 57(3):389–398, 2016.
[AK00]   C. J. Ash and J. Knight. *Computable structures and the hyperarithmetical hierarchy*,
            volume 144 of *Studies in Logic and the Foundations of Mathematics*. North-Holland
            Publishing Co., Amsterdam, 2000.
[Ash87]   C. J. Ash. Categoricity in hyperarithmetical degrees. *Ann. Pure Appl. Logic*, 34(1):1–
            14, 1987.
[BKY18]   Nikolay A. Bazhenov, Iskander Sh. Kalimullin, and Mars M. Yamaleev. Degrees of
            categoricity and spectral dimension. *J. Symb. Log.*, 83(1):103–116, 2018.
[CFS13]   Barbara F. Csima, Johanna N. Y. Franklin, and Richard A. Shore. Degrees of cate-
            goricity and the hyperarithmetic hierarchy. *Notre Dame J. Form. Log.*, 54(2):215–231,
            2013.
[CGK99]   Peter Cholak, Sergey Goncharov, and Bakhadyr Khoussainov. Computably categorical
            structures and expansions by constants. *J. Symb. Log.*, 64(1):13–37, 1999.
[CHT17]   Barbara F. Csima and Matthew Harrison-Trainor. Degrees of categoricity on a cone
            via -systems. *J. Symb. Log.*, 82(1):325–346, 2017.
[CS19]   Barbara F. Csima and Jonathan Stephenson. Finite computable dimension and degrees
            of categoricity. *Ann. Pure Appl. Logic*, 170(1):58–94, 2019.
[DKL$^+$15] Rodney G. Downey, Asher M. Kach, Steffen Lempp, Andrew E. M. Lewis-Pye, Antonio
            Montalbán, and Daniel D. Turetsky. The complexity of computable categoricity. *Adv.
            Math.*, 268:423–466, 2015.
[FKM10]   Ekaterina B. Fokina, Iskander Kalimullin, and Russell Miller. Degrees of categoricity
            of computable structures. *Arch. Math. Logic*, 49(1):51–67, 2010.
[Fra17]   Johanna N. Y. Franklin. Strength and weakness in computable structure theory. In
            Computability and complexity, volume 10010 of *Lecture Notes in Comput. Sci.*, pages
            302–323. Springer, Cham, 2017.
[FS14]   Johanna N. Y. Franklin and Reed Solomon. Degrees that are low for isomorphism.
            Computability, 3(2):73–89, 2014.
[FT]   Johanna N. Y. Franklin and Dan Turetsky. Taking the path computably travelled.
            Submitted.
[FT18]   Johanna N. Y. Franklin and Dan Turetsky. Lowness for isomorphism and degrees of
            genericity. *Computability*, 7(1):1–6, 2018.
[GM17]   Noam Greenberg and Benoit Monin. Higher randomness and genericity. *Forum Math.
            Sigma*, 5:e31, 41, 2017.
[Gon77]   S. S. Gončarov. The number of nonautoequivalent constructivizations. *Algebra i Logika*,
            16(3):257–282, 377, 1977.
[HKS03]   Denis R. Hirschfeldt, Bakhadyr Khoussainov, and Richard A. Shore. A computably
            categorical structure whose expansion by a constant has infinite computable dimension.
            J. Symbolic Logic, 68(4):1199–1241, 2003.
[MS18]   Timothy H. McNicholl and Don Stull. The isometry degree of a computable copy of
            ℓ^p. In *Sailing routes in the world of computation*, volume 10936 of *Lecture Notes in
            Comput. Sci.*, pages 277–286. Springer, Cham, 2018.

School of Mathematics and Statistics, Victoria University of Wellington, Welling-
ton, New Zealand
   E-mail address: dan.turetsky@vuw.ac.nz