COMPUTABLE POLISH GROUP ACTIONS

ALEXANDER MELNIKOV AND ANTONIO MONTALBÁN

Abstract. Using methods from computable analysis, we establish a new connection between two seemingly distant areas of logic: computable structure theory and invariant descriptive set theory. We extend several fundamental results of computable structure theory to the more general setting of topological group actions. As we will see, the usual action of S_∞ on the space of structures in a given language is *effective* in a certain algorithmic sense that we need, and S_∞ itself carries a natural computability strucutre (to be defined). Among other results, we give a sufficient condition for an orbit under effective $\mathcal{G}$-action of a computable Polish $\mathcal{G}$ to split into infinitely many disjoint effective orbits. Our results are not only more general than the respective results in computable structure theory, but they also tend to have proofs different from (and sometimes simpler than) the previously known proofs of the respective prototype results.

1. Introduction

We show how various results from computable structure theory generalize to the setting of computable Polish group actions. On the one hand, this allows us to obtain new results on the computability of Polish group actions. Polish group actions appear in different areas of mathematics, as for instance the action of GL_n on $\mathbb{R}^n$, or the action of the group of orientation-preserving self-homeomorphisms of $[0, 1]$ on $C[0, 1]$. In all these examples, notions like computable categoricity and degree spectrum are still meaningful, and so are the results we generalize. On the other hand, our results give us a better understanding of some of the classical results from computable structure theory: Being able to prove results about computable structures using only topological arguments and without using the the elements of the structures, allows us to understand the underlying reasons that make these results true.

For a given countable language $\mathcal{L}$, the set of $\mathcal{L}$-structures with domain ω, denoted $\mathcal{X}_{\mathcal{L}}$, has a natural topology associated to it which makes it a Polish space. The permutation group on ω, denoted S_∞, acts naturally on this space by permuting the domain of the structures. One could then apply general results about Polish group actions to get results about $\mathcal{L}$-structures and their isomorphisms. Descriptive set theory have been studying this

The first author was supported by the Packard Foundation. The second author was partially supported the Packard Fellowship and NSF grant DMS-1363310. We thank Slawomir Solecki who found an error in one of our proofs.

kind of interactions for a while. In this paper we look at this idea from the point of view of computable structure theory. To our surprise, we have been able to show how five known results on computable structure theory can be extended to the much more general setting of Polish group actions.

In the language of computable Polish actions, when we consider action of S_∞ into $\mathcal{X}_\mathcal{L}$, we can talk about the following basic notions: presentation of a structure, degree of the presentation, isomorphism between presentations, and degree of the isomorphism. Thus, any other notion or theorem purely based on these basic notions can be expressed in generality in the language of computable Polish actions. Let us start by looking at the notion of computable categoricity. A computable structure $\mathcal{A}$ is *computable categorical* if for every computable structure $\mathcal{B}$ isomorphic to $\mathcal{A}$ there is a computable isomorphisms between them. Re-writing this definition in terms of the group action of S_∞ on $\mathcal{X}_\mathcal{L}$ we get:

Definition 1. Let $(\mathcal{X}, \mathcal{G}, a)$ be a computable transformation group, that is, let a be a computable action of a computable Polish group on a computable Polish space (see Definition 15). A computable point $x \in \mathcal{X}$ is *computably categorical* if for every computable point $y \in \mathcal{X}$ in the orbit of x, there is a computable $g \in G$ such that $g \cdot x = y$. We say that x is *uniformly computably categorical* if there exists a computable operator Φ that given a presentation of a point y in the orbit of x, it outputs $g \in G$ with $g \cdot x = y$.

We say that an effective property P holds *on a cone* if there exists a Turing degree $\mathbf{a}$ such that for any $\mathbf{b} \geq \mathbf{a}$ the relativized property $P^{\mathbf{b}}$ holds. For example, the second author has showed in [Mon] that a structure $\mathcal{A}$ is uniformly computably categorical on a cone if and only if the set of copies of $\mathcal{A}$ is a $\mathbf{\Pi^0_2}$ set of reals. The proof used Goncharov's [Gon75] characterization of relative computable categoricity, Ventsov's [Ven92] analysis of uniform computable categoricity, Lopez-Escobar's [LE65] theorem connecting Borel complex with $\mathcal{L}_{\omega_1,\omega}$, and a finer version of the type-omitting theorem for infinitary logic. This statement can be re-phrased in terms of Polish group actions as follows: A point x is uniformly computably categorical on a cone if and only its orbit is G_δ. Analyzing what uniformly computably categorical on a cone means in this sense, we get that it is equivalent to saying that the map $g \mapsto g \cdot x \colon \mathcal{G} \to \mathcal{X}$ is open. In 1965, Effros [Eff65] (see also [Gao09, Theorem 3.2.4]) proved that the map $g \mapsto g \cdot x \colon \mathcal{G} \to \mathcal{X}$ is open if and only if the orbit of x is G_δ. We will see that Montalbán's result from [Mon] is a corollary of Effros's 50-year-old theorem. In contrast to the more complicated proof in [Mon], our new proof uses only Effros theorem and some basic computable analysis techniques.

Then we look at a well-known unpublished result of Knight and her group from the 90's. They showed that a non-trivial union of countably many upper cones can never be the degree spectrum of a structure. (By non-trivial we mean that the union is not equal to a single cone.) Their result is also true when we look at enumeration degree spectra.

Definition 2. Given a point x in a computable transformation group $(\mathcal{X}, \mathcal{G}, a)$, the $\mathcal{G}$-enumeration-degree spectrum of x is defined by:

$$DgSp^e_{\mathcal{G}}(x) = \{X \subseteq \omega : \\ \text{Every enumeration of } X \text{ computes a presentation of a point } y \equiv_{\mathcal{G}} x\}.$$

Note that a point $x \in \mathcal{X}$ (in a computable Polish space $\mathcal{X}$) is computable if, and only if, its name $N^x = \{B \text{ basic open } : x \in B\}$ is c.e. (all notions will be clarified in the preliminaries). Thus, the result below for enumeration degree spectra is more natural in our setting, and it also extends the classical result from computable structure theory.

Theorem 3. *The non-trivial union of at most countably many (≥ 2) incomparable cones cannot be realized as the $\mathcal{G}$-enumeration-degree spectrum of any $x \in \mathcal{X}$.*

We also generalise the theory of *computable dimension* (also called *autodimension*, see the book [EG00]) to computable group actions. Recall that a computable algebraic structure has computable dimension $\alpha \in \omega \cup \{\omega\}$ if the structure has α computable presentations, up to computable isomorphism. In the more general set up of computable group actions, this corresponds to splitting a $\mathcal{G}$-orbit further into sub-classes of effectively $\mathcal{G}$-equivalent elements. Then the number of such classes in $\mathrm{Orb}(x)$ will be the computable $\mathcal{G}$-dimension of x.

We look at Goncharov's [Gon75] theorem that if we have two computable copies of a structures which are Δ^0_2 isomorphic, but not computably isomorphic, then the structure has infinite computable dimension. We prove a version of Goncharv's result true for *all* computable effectively open $\mathcal{G}$-spaces, but we require an "isomorphism" (in this setting, $g \in \mathcal{G}$) to be computable from a non-high c.e. oracle.

Theorem 4. *Let $\mathcal{X}$ be a computable $\mathcal{G}$-space such that the action of $\mathcal{G}$ on $\mathcal{X}$ is also effectively open. Let $x, y \in \mathcal{X}$ be computable points that are not computably equivalent but that, for some $g \in \mathcal{G}$ computable from a c.e. non-high set, we have*

$$y = g \cdot x.$$

Then x has infinite computable $\mathcal{G}$-dimension.

Let us remark that group actions are always open as functions $\mathcal{G} \times \mathcal{X} \to \mathcal{X}$. Our proof is a finite injury construction that is implemented via the true stages method. We leave open whether our extra effectiveness conditions can be dropped or improved in general. It seems that Goncharov's original proof heavily uses special properties of the S_∞ action on $\mathcal{X}_{\mathcal{L}}$. Our effectiveness condition gives a rather different reason for the proof to work. Although it is not even clear how to extend the theorem beyond non-c.e. high degrees, the general result in its present form was somewhat unexpected.

2. Background

2.1. **Computable metric and topological spaces.**

We will be mostly interested in the case when both the group $\mathcal{G}$ and the space $\mathcal{X}$ are effectively metrizable. The following definition is standard (e.g., [BHW08]).

Definition 5. A *computable Polish metric space* $\mathcal{M}$ is a triple $(M, d, (x_i)_{i\in\omega})$, where (M, d) is a Polish metric space and $(x_i)_{i\in\omega}$ is a dense sequence in M such that, for any $i, j \in \omega$, $d(x_i, x_j)$ is a computable real uniformly in i, j. When we refer to a *computable* Polish space, we assume it comes with an associated metric making it a computable Polish metric space.

It will be typically more convenient to use the associated topology rather than the metric. Using topology allows us to eliminate repetitive uses of triangle inequality and concrete choices of distances in some proofs, thus clarifying the proofs. Computable topological spaces that are not necessarily metrizable have also been studied (e.g., [WG09]). A *computable topological space* is a pair $(\mathcal{X}, \nu)$, where $\nu : \mathbb{N} \to \tau$ is a numbering of a countable basis of the topological space $\mathcal{X}$, so that

$$\nu(i) \cap \nu(j) = \bigcup_{(i,j,k)\in R} \nu(k),$$

where $R \subseteq \mathbb{N}^3$ is c.e.

Every computable metric space is clearly a computable topological space, with the base consisting of the basic open balls with rational radii and having their centers in the dense computable set. What makes metric spaces much more convenient from the recursion-theoretic viewpoint is the notion of *fast convergence* which is behind the following definition.

Definition 6. Let $\mathcal{M}$ be a Polish space. Strengthening the approach from [Wei01], we say that a basic open ball $B_r(x)$ is *formally included* in $B_q(y)$, written $B_r(x) \ll B_q(y)$, if $r + d(x, y) < q$ and $r < q/2$.

We will use formal inclusion throughout the paper. We list below several useful properties of formal inclusion. For basic open sets A, B, B_i, C, we have:

(F1) $A \ll B$ implies $A \subseteq B$.

(F2) For any $x \in \mathcal{X}$ and $C \ni x$ there exists a $B \ni x$ such that $B \ll C$.

(F3) For any countably infinite $B_0 \gg B_1 \gg B_2 \gg \cdots$ there is an $x \in \mathcal{X}$ such that $\bigcap_{i\in\omega} B_i = \{x\}$. We denote this by
$$B_0 \gg B_1 \gg B_2 \gg \cdots \to x.$$

(F4) For any countably infinite $B_0 \gg B_1 \gg B_2 \gg \cdots$ and any $C \supseteq \bigcap_{i\in\omega} B_i$ there exists an i such that $C \gg B_i$.

Property (F3) is a consequence of $\mathcal{M}$ being complete, while the other three properties follow from the basic definitions and the triangle inequality. In (F3), we say that $(B_i)_{i\in\omega}$ converges to x *fast*. Observe that in a computable metric space, formal inclusion is c.e., while the usual inclusion need not be c.e. We will use these properties throughout the paper without explicit reference.

Remark 7. We note that some fundamental properties of computable Polish spaces can be proved using (F1)-(F4) without any reference to the metric. Apart from the technical and notational convenience of $\ll$, we also sometimes get a more general result for free (e.g., the lemmas in the next few subsections as well as the proof of Theorem 4 work in this more general setting). Indeed, there exist computable topological spaces that possess a countable basis and a c.e. relation $\ll$ satisfying (F1)-(F4) that are not even regular[1].

2.1.1. *Computable points and functions.* In this section we establish some useful characterizations of computable points and computable maps in computable Polish spaces. We also compare the approaches via metric and via topology, and we use $\ll$ to illustrate how we get equivalent notions in the context of computable Polish spaces.

Definition 8. Let $\mathcal{X}$ be a computable topological space. We call

$$N^x = \{i : x \in B_i\}$$

the name of x (in $\mathcal{X}$).

A *fast Cauchy name* (f.c.n.) for a point x in a Polish space is a sequence $(j_i)_{i\in\omega}$ such that $d(x_{j_i}, x) < 2^{-i}$ (for all i) and $x = \lim_i x_{j_i}$. Equivalently, a *fast Cauchy name* for a point x is a sequence of (indices of) basic open balls $\{B_i : i \in \omega\}$ such that $B_0 \gg B_1 \gg B_2 \gg \cdots \to x$. Recall also that a point in a computable Polish space is *computable* if it has a computable fast Cauchy name [BHW08]. We illustrate that this notion is consistent with Definition 8.

Fact 9. *Let x be a point in a computable Polish space. Then x is computable iff N^x is c.e.*

Proof. Indeed, if N^y is c.e. we can uniformly produce a sequence $B_0 \gg B_1 \gg \cdots$ such that $\bigcap_{i\in\omega} B_i = \{x\}$. Taking the centers of the B_i, we get the desired c.e. fast Cauchy sequence. Conversely, note that if there exists a c.e. fast Cauchy sequence then we can produce a c.e. sequence of basic open balls $B_0 \gg B_1 \gg \cdots$ such that $\bigcap_{i\in\omega} B_i = \{x\}$. Then $B \in N^y$ iff $(\exists i) B_i \ll B$. □

[1] For example, define $\mathcal{X}$ to be $\mathbb{R}^2$ without $\{(0,y) : y \text{ irrational}\}$. The basis of topology consists of standard Euclidean open discs $B^E_r(v,q)$ with rational parameters r, v, q, $|r| < |v|$, and also special "squashed" discs that have the form $B^E_r(0,q)$-$\{(0,k) : k \neq q\}]$, where q, r are rational. The space is not regular because the closed set $Y = \{(0,y) : y \in \mathbb{Q}, y \neq 0\}$ and $(0,0)$ cannot be separated by open sets. To define $\ll$ on the basis, follow Definition 6 literally, but in the case when both discs are special require that their centres are equal. The definition of $\ll$ is clearly effective upon the fixed base. Going through various cases, we may verify that (F1)-(F4) hold as well. For (F1) use that a standard disc cannot intersect the y-axis, and special discs are compatible under $\subseteq$ only if they are centred at the same point. To get (F2) shrink $C \ni x$. For (F3), note that either the whole sequence is special (and thus the discs converge to their common center) or it is eventually standard. Finally, (F4) can be derived from (F3) and the definitions of $\mathcal{X}$ and $\ll$.

Observe that the proof above is uniform. It follows that N^x has the least enumeration degree among all enumeration degrees of fast Cauchy names of x.

We can also use basic open balls to produce names of open sets, as follows. A *name* of an open set U in a computable topological space $\mathcal{X}$ is a set $W \subseteq \mathbb{N}$ such that $U = \bigcup_{i \in W} B_i$, where B_i stands for the i-th basic open set in the basis of $\mathcal{X}$. If an open U has a c.e. name, then we say that U is *effectively open*.

Definition 10. A function $f \colon \mathcal{X} \to \mathcal{Y}$ between two computable topological spaces is *effectively continuous* if there is a c.e. family $F \subseteq \mathcal{P}(\mathcal{X}) \times \mathcal{P}(\mathcal{Y})$ of pairs of (indices of) basic open sets in such that:

(C1): for every $(U, V) \in F$, $f(U) \subseteq V$;

(C2): for every $x \in \mathcal{X}$ and basic open $E \ni f(x)$ in $\mathcal{Y}$ there exists a basic open $D \ni x$ in $\mathcal{X}$ such that $(D, E) \in F$.

Note that a function is continuous if and only if it is effectively continuous relative to some oracle. The lemma below is well-known.

Lemma 11. *Let $f \colon \mathcal{X} \to \mathcal{Y}$ be a function between computable Polish spaces. The following are equivalent:*

(1) f is effectively continuous.

(2) There is an enumeration operator Φ that on input a name of an open set Y (in $\mathcal{Y}$), lists a name of $f^{-1}(Y)$ (in $\mathcal{X}$).

(3) There is an enumeration operator Ψ, that given the name of $x \in \mathcal{X}$, enumerates the name of $f(x)$ in $\mathcal{Y}$.

(4) There exists a uniformly effective procedure that on input a fast Cauchy name of $x \in \mathcal{M}$ lists a fast Cauchy name of $f(x)$ (note that the Cauchy names need not be computable).

2.1.2. *Effectively open maps and c.e. subspaces.* The properties of effectively open maps are dual to the properties of computable maps.

Definition 12. A function $f \colon \mathcal{X} \to \mathcal{Y}$ is *effectively open* if there is a c.e. family F of pairs of basic open sets such that

(O1): for every $(U, V) \in F$, $f(U) \supseteq V$;

(O2): for every $x \in \mathcal{X}$ and any basic open $E \ni x$ there exists a basic open $D \ni f(x)$ such that $(E, D) \in F$.

Lemma 13. *Lef $f \colon \mathcal{X} \to \mathcal{Y}$ be a function between computable Polish spaces. The following are equivalent:*

(1) f is effectively open.

(2) There is an enumeration operator that given a name of an open set A in $\mathcal{X}$, outputs a name of the open set $f(A)$ in $\mathcal{Y}$.

Proof. (1) $\Rightarrow$ (2). Suppose $E = \bigcup_{i \in W} B_i$. We have $f(B_i) = \bigcup \{D : (B_i, D) \in F\}$. The collection of all D such that for some $i \in W$, $(B_i, D) \in F$ is a name for $f(E)$. Clearly, given any enumeration of W we can uniformly produce an enumeration of all such D.

(2) $\Rightarrow$ (1). For each basic open ball B_i in $\mathcal{X}$, let V_i be a name for $f(B_i)$. Note that (2) implies that $\{V_i : i \in \omega\}$ is uniformly c.e. Let $F = \{(i,j) : i \in \omega, j \in V_i\}$. So F is a set of pairs of (indices of) open balls, and it is not hard to see that F witnesses that f is effectively open. $\square$

As far as we are concerned, there is no standard notion of an effectively open map between computable Polish metric spaces. However, the most natural "pointwise" definition would be equivalent to our definition (we omit details). The next simple lemma will be used later in the paper.

Lemma 14. *Suppose $f : \mathcal{X} \to \mathcal{Y}$ is a continuous effectively-open map between computable Polish spaces. Then there exists an enumeration operator that on input N^y of $y \in f(\mathcal{X})$ lists N^x of some $x \in f^{-1}(y)$.*

Proof. Given an enumeration of N^y, we use Fact 9 to get a sequence $U_0 \gg U_1 \gg \ldots \to y$. We build a sequence of basic open balls $\{B_i : i \in \omega\}$ such that for every i: $B_i \ll B_{i-1}$ and $f(B_i) \supseteq U_j$ for some j. Given B_{i-1}, it is not hard to see that such a B_i exists. We use the fact that f is effectively open and that $\ll$ is c.e. to find one such B_i. We then get a fast Cauchy name for a point x: $B_0 \gg B_1 \gg \ldots \to x$. Since f is continuous, $\bigcap_i f(B_i)$ contains only one point, and hence we must have $f(x) = y$. $\square$

2.2. **Computable topological groups and their actions.** A group $\mathcal{G}$ is a *computable Polish group* if it is a computable Polish space in which the operations $(x,y) \to xy$ and $x \to x^{-1}$ are computable maps (in the respective Cartesian power of $\mathcal{G}$).

Examples of effective topological groups include S_∞ with the usual base of clopen sets, the space of $n \times n$ invertible real or complex matrices[2], the group of orientation-preserving self-homeomorphisms of $[0,1]$, and many other examples. In this paper we are concerned with effective actions of such groups.

Definition 15. Suppose a Polish group $\mathcal{G}$ acts on a Polish space $\mathcal{M}$. We say that the action is *effective* if $\mathcal{G}$ is a computable Polish group, $\mathcal{M}$ is a computable Polish space, and the action is represented by a computable function $a\colon \mathcal{G} \times \mathcal{M} \to \mathcal{M}$. We call $(\mathcal{X}, \mathcal{G}, a)$ a *computable transformation group*.

The following lemma is a version of the well-known result that if a real is computable in every Cohen generic, then it is computable. We will use it a few times throughout the paper. We use quantifiers $\exists^* x$ and $\forall^* x$ that mean "for non-meager many x" and "comeager many x" respectively.

[2] In $M_{n\times n}(\mathbb{R})$, the compatible complete metric is given by the usual Euclidian distance d_E on $\mathbb{R}^{n^2}$ modified using the continuous determinant function: $d'(A,B) = d_E(A,B) + |\frac{1}{det(A)} - \frac{1}{det(B)}|$. The natural base of open balls centered in rational-valued invetible matrices makes the group effectively Polish.

Lemma 16. *Let $\mathcal{G}$ be a computable Polish group and let $A \subseteq \omega$ be such that*

$$(\exists^* h \in \mathcal{G})\ A \leq_e N^h.$$

Then A is c.e.

Proof. We have that for some basic open set V, for co-meager many $h \in V$, there is an enumeration operator Φ, such that $A = \Phi^{N^h}$. Since for each Φ, the set of such h is Borel, there must be an operator Ψ such that $(\exists^* h \in \mathcal{G})\ A = \Psi^{N^h}$. Let U be a basic open set such that $(\forall^* h \in U)\ A = \Psi^{N^h}$. We claim that

$$A = \{n \in \omega : \exists B_{k_1}, ..., B_{k_m} \left(\emptyset \neq \cap_{i \leq m} B_{k_i} \cap U \wedge n \in \Psi^{\{B_{k_1}, ..., B_{k_m}\}}\right)\}.$$

If $n \in A$, then $n \in \Psi^{N^h}$ for some $h \in U$, and Ψ enumerates n using only finitely many balls from N^h. Let $B_{k_1}, ..., B_{k_m}$ be those balls. If n is in the right-hand-side as witnessed by $B_{k_1}, ..., B_{k_m}$, we can extend this set of basic open balls to N^h for some $h \in U$ with $A = \Psi^{N^h}$. Thus $n \in A$.

Note that the right-hand-side is c.e. □

3. Categoricity

Let us start by looking at the notion of computable categoricity. Recall from Definition 1 that we call a computable point $x \in \mathcal{X}$ is *computable categorical* if for every computable point $y \in \mathcal{X}$ in the orbit of x, there is a computable $g \in G$ such that $g \cdot x = y$. Also, a computable point x is *uniformly computably categorical* if there is an enumeration operator that given an enumeration of N^y for a point y in the orbit of x, outputs the name of $g \in G$ with $g \cdot x = y$.

The following theorem is generalization of Montalbán's theorem that a structure is computably categorical on a cone if an only if it has a $\Pi_2^{\mathtt{in}}$ Scott Sentence. (The theorem we are referring to is the equivalence between (U2) and (U3) (and also (U4)) in Theorem 1.1 in [Mon] for $\alpha = 1$. Other equivalent characterizations are given in [Mon], and the proof there works for all ordinals α.)

Theorem 17. *Let $(\mathcal{X}, \mathcal{G}, a)$ be a transformation group, and $x \in \mathcal{X}$. The following are equivalent:*

(1) x is uniformly computably categorical on a cone;
(2) the orbit of x is $\mathbf{\Pi_2^0}$.

Proof. (1) ⇒ (2). For this direction we need to show that the orbit of x has a Π_2^0 definition. Let Φ be the enumeration functional that witnesses uniform categoricity of x. Then y is in the orbit of x if and only if Φ^{N^y} is the name of a element $g \in \mathcal{G}$ and $g \cdot x = y$. We now need to observe this is a Π_2^0 statement. To say that Φ^{N^y} is the name of a element $g \in \mathcal{G}$, we need to say that Φ^{N^y} is a filter, and that it contains basic open balls of arbitrary small radius — this is easily seen to be Π_2^0. To say that $\Phi^{N^y} \cdot x = y$, we need to say that for every $Y \in N^y$ there exist basic open $B \in \Phi^{N^y}$ and a basic open

$B' \in N^x$ with $BB' \subseteq Y$ — which again is easily seen to be Π^0_2 using the presentation of the action given by Definition 10.

(2) ⇒ (1) We make us of the following well-known result:

Theorem 18 (Effros [Eff65])**.** *Let $(\mathcal{X}, \mathcal{G}, a)$ be a transformation group, and $x \in \mathcal{X}$. The following are equivalent:*

(a) the map $g \mapsto g \cdot x \colon \mathcal{G} \to \mathcal{X}$ is open;
(b) the orbit of x is G_δ.

Now since the orbit of x is $\mathbf{\Pi^0_2}$, i.e. G_δ, we get that $g \mapsto g \cdot x \colon \mathcal{G} \to \mathcal{X}$ is open. Now, relative to some oracle, the map is effectively open. By Lemma 14, we have x is u.c.c. on a cone. □

4. Degree spectra

In this section, $\mathcal{G}$ is a computable Polish group that acts effectively on a computable Polish space M. We now prove Theorem 3 that states that the degree spectrum of a point $x \in \mathcal{X}$ can never be a non-trivial union of countably many e-cones.

Proof of Theorem 3. We prove the case of only two cones. It will be clear that almost literally the same proof works for any finite number of incomparable cones or countably many incomparable e-cones.

Aiming for a contradiction, suppose the degree spectrum of some x is the union of two e-cones,

$$DgSp^e_{\mathcal{G}}(x) = \{X \subseteq \omega : X \geq_e A\} \cup \{X \subseteq \omega : X \geq_e B\},$$

for some $A, B \subseteq \omega$ such that $A|_e B$.

Define $C_a = \{g \in G : A \leq_e N^{gx}\}$, and C_b is defined mutatis mutandis. These sets are Borel, so one of them must be non-meager; say C_a. (Note the same would be true if they were countably many.) If Z is an oracle such that N^x is Z-c.e., then for every $g \in C_a$, A is Z-enumerable from N^g. It follows from Lemma 16 that A is Z-c.e., and hence that $A \leq_e N^x$. But then $DgSp^e_{\mathcal{G}}(x) \subseteq \{X \subseteq \omega : X \geq_e A\}$ and hence $DgSp^e_{\mathcal{G}}(x) = \{X \subseteq \omega : X \geq_e A\}$, getting a trivial union of cones. □

5. Proof of Theorem 4

In this section we prove Theorem 4 that states that a point $x \in \mathcal{X}$ has infinite computable $\mathcal{G}$-dimension if there is a $y \in \mathrm{Orb}(x)$ which it not computably equivalent, but there is some $g \in \mathcal{G}$ computable in a c.e. non-high oracle, with $y = g \cdot x$.

5.1. Preliminary analysis.

We start this subsection by giving a rather informal idea of the proof. Then we establish two auxiliary technical facts about approximations of Δ^0_2-points.

5.1.1. *Informal idea.* In the construction, we will build infinitely many points $z^i \in \mathrm{Orb}(x)$ and diagonalize against all potential computable h_e in $\mathcal{G}$ for which $h_e \cdot z^i = z^j$. For this we start by "copying" x to z^i and y to z^j, $i \neq j$. To copy x to z^i we start defining a computable sequence of nested balls of $\mathcal{G}$ working towards a computable a_i such that $a_i x = z^i$. While copying, we wait for the first disagreement in $h_e z^i = z^j$. If h_e is total then we must eventually find a disagreement, for otherwise x can be mapped to y using a computable element of $\mathcal{G}$, contrary to our assumption. These diagonalization strategies are finitary in nature. Thus, once we act for them, we do not need to keep on defining a_i as a computable point. Some diagonalization requirements may require z^i to copy x and some may require it to copy y. Thus, we will have to "switch" from copying x to copying y in-between our actions using a Δ^0_2 element $g \in \mathcal{G}$ such that $y = g \cdot x$, simultaneously preserving parts of the copying procedures of higher priorities. Although this sounds a lot like the proof of the original theorem for structures, there is one crucial difference. More specifically, lower priority strategies may potentially injure a higher priority strategy. This would not be a problem in the case of the standard S_∞ action on a space of $\mathcal{L}$-structures. Very informally, this is because Δ^0_2-elements of S_∞ behave very nicely with respect to this action; that is, the action of a Δ^0_2 point on a basic open set of $X_\mathcal{L}$ eventually gets truly stable for any given precision ϵ. We omit a formal explanation of this phenomenon. However, in contrast to the standard S_∞-action, in an arbitrary $\mathcal{G}$-space a Δ^0_2 element g (and thus, its action) may introduce an infinite injury effect to the construction. Here the non-highness condition comes into play: it will allow us to produce a neater approximation to the Δ^0_2 element g. We will use Martin's theorem to produce approximations $(G_s)_{s\in\omega}$ and $(\tilde{G}_s)_{s\in\omega}$ to g and g^{-1} such that $(G_s^{-1}\tilde{G}_s)_{s\in\omega}$ is eventually close to $e_\mathcal{G}$, regardless of whether the approximations are correct or not. We will then argue that, in this case, small perturbations of g will no longer be harmful. Note this gives a different reason for the construction to work when compared with the original proof for structures. Unfortunately, there are several technical subtleties in implementing this ideas, and these cannot be explained informally.

5.2. **Δ^0_2 and non-high points.** Let us start by looking at computable approximations to Δ^0_2 points. There are various ways of characterizing the Δ^0_2 points of an effective Polish group. We chose to use the method of true stages.

Before that, let us recall the notion of a true stage introduced by Lachlan. We view countable sets as elements of 2^ω. Let X be a c.e. set with a computable enumeration $\{x_0, x_1, x_2, ...\}$. For each s, let $X_s = \{x_0, ..., x_s\} \upharpoonright x_s + 1$. We view X_s as a string in 2^{x_s+1}. So, $\{X_s : s \in \omega\}$ is a computable sequence of binary strings trying to approximate $X \in 2^\omega$. A stage t is said to be *a true stage (for the enumeration of X)* if its approximation to X is correct, in the sense that $X_t \subseteq X$ (where the inclusion is *as strings*). Equivalently, t is a true stage if and only if $(\forall s \geq t) X_s \supseteq X_t$, and also, if and only if $(\forall s \geq t) x_s \geq x_t$. It is not hard to show that there are infinitely many true stages. We say that t *looks true at a stage* $s \geq t$ *(for the enumeration of* X*)* if $X_t \subseteq X_s$. (If $t \geq s$, then we agree that t looks true at s.) Notice that if $t < r < s$ and t looks true at s, then t looks true at r too.

We want a uniform way of saying that a sequence of balls gets small fast. For that, fix in $\mathcal{G}$ a computable basis of open neighbourhoods of the identity $e_{\mathcal{G}}$ and produce a computable fast Cauchy name of $e_{\mathcal{G}}$:

$$E_0 \gg E_1 \gg E_2 \gg \cdots \to e_{\mathcal{G}}.$$

Lemma 19. *Let* X *be c.e. set. An element* $g \in \mathcal{G}$ *is* X*-computable if and only if there exist computable sequences of basic open balls* $\{F_s : s \in \omega\}$ *and* $\{\tilde{F}_s : s \in \omega\}$ *and an infinite sequence of stages* $t_0 < t_1 < t_2 < \dots$ *such that*

(1) $F_{t_0} \gg F_{t_1} \gg F_{t_2} \gg \cdots \to g$ *and* $\tilde{F}_{t_0} \gg \tilde{F}_{t_1} \gg \tilde{F}_{t_2} \gg \cdots \to g^{-1}$*;*
(2) for every n, $\tilde{F}_{t_n} \cdot F_{t_n} \subseteq E_n$ *and* $F_{t_n} \cdot \tilde{F}_{t_n} \subseteq E_n$*;*
(3) for all $s < t$*, if* s *looks true at* t *for the enumeration of* X*, then* $F_t \subseteq F_s$ *and* $\tilde{F}_t \subseteq \tilde{F}_s$*;*
(4) every t_n *is true for the enumeration of* X*.*

Proof. For the right-to-left direction notice that the sequence of true stages is computable in X. We can then compute a sub-sequence $l_0 < l_1 < l_2 < \dots.$ satisfying $F_{l_0} \gg F_{l_1} \gg F_{l_2} \gg \cdots \to g$. We thus get an X-computable fast Cauchy name for g.

For the left-to-right direction let $H_0 \gg H_1 \gg \cdots \to g$ and $\tilde{H}_0 \gg \tilde{H}_1 \gg \cdots \to g^{-1}$ be X-computable fast Cauchy names for g and g^{-1}. By taking a subsequence if necessary, assume also that $\tilde{H}_n \cdot H_n \subseteq E_n$ and $H_n \cdot \tilde{H}_n \subseteq E_n$. Let Φ and Ψ be the Turing functionals such that $\Phi^X(n)$ is an index for H_n and $\Psi^X(n)$ is an index for $\tilde{H}_n$.

For each $s \in \omega$, to define F_s and $\tilde{F}_s$, the idea is to look at the values of $\Phi^{X_s}(i)$ and $\Psi^{X_s}(i)$ which converge, and take the greatest one. We need to be a bit more careful. Let $i_s \leq s$ be the greatest such that for all $i \leq i_s$,

(1) $\Phi^{X_s}(i)$ converges to an index of a basic open ball J_i, and $\Psi^{X_s}(i)$ converges to an index of a basic open ball $\tilde{J}_i$;
(2) $\tilde{J}_i \cdot J_i \subseteq E_i$ and $J_i \cdot \tilde{J}_i \subseteq E_i$; and
(3) $J_0 \gg J_1 \gg \cdots \gg J_{i_s}$ and $\tilde{J}_0 \gg \tilde{J}_1 \gg \cdots \gg \tilde{J}_{i_s}$.

Then, let $F_s = J_{i_s}$ and $\tilde{F}_s = \tilde{J}_{i_s}$.

Notice that if $X_s \subseteq X_t$, then $\Phi^{X_t}(i)$ and $\Phi^{X_t}(i)$ converge on more values than they do with oracle X_s, and hence $F_s \supseteq F_t$ and $\tilde{F}_s \supseteq \tilde{F}_t$. Among the true stages, we can take an increasing subsequence $\{t_j : j \in \omega\}$ such that $\{i_{t_j} : j \in \omega\}$ is strictly increasing and hence the sequences $\{F_{t_j} : j \in \omega\}$ and $\{\tilde{F}_{t_j} : j \in \omega\}$ are sub-sequences of the original X-computable sequences $H_0 \gg H_1 \gg \cdots \to g$ and $\tilde{H}_0 \gg \tilde{H}_1 \gg \cdots \to g^{-1}$. □

When we have that g is not only Δ^0_2, but also computable from a non-high c.e. oracle X, we get a slightly better behaved approximation. (Recall that a

$X \in 2^\omega$ is *high* if $X' \geq_T 0''$.) This slight improvement to our approximation will be key at the very end of our construction. What we get from the fact that X is c.e. is that we are able to use a true-stage approximation exactly as we did above. What we get from the fact that X is non-high is that, for every H-computable function, there is a computable function not dominated by it. This is by Martin's theorem [Mar66] that states that a set is high if and only if it computes a function that dominates all total computable functions.

Lemma 20. *Let $g \in \mathcal{G}$ be computable from a non-high c.e. set X. Then there exist computable sequences of basic open balls $\{G_s : s \in \omega\}$ and $\{\tilde{G}_s : s \in \omega\}$, and an infinite sequence $t_0 < t_1 < t_2 < \dots$ such that*

1. *$G_{t_0} \gg G_{t_1} \gg G_{t_2} \gg \cdots \to g$ and $\tilde{G}_{t_0} \gg \tilde{G}_{t_1} \gg \tilde{G}_{t_2} \gg \cdots \to g^{-1}$;*
2. *for every n, $\tilde{G}_n \cdot G_n \subseteq E_n$ and $G_n \cdot \tilde{G}_n \subseteq E_n$.*
3. *for all $s < t$, if s looks true at t for the enumeration of X, then $G_t \subseteq G_s$ and $\tilde{G}_t \subseteq \tilde{G}_s$;*
4. *every t_n is true for the enumeration of X.*

Notice that the difference with the previous lemma is that now we get that for *every* n we have $\tilde{G}_n \cdot G_n \subseteq E_n$ and $G_n \cdot \tilde{G}_n \subseteq E_n$, and not just for the ones from the special sequence.

Proof. Let $\{F_s : s \in \omega\}$ and $\{\tilde{F}_s : s \in \omega\}$ be exactly as in the previous lemma. We will define the sequences $\{G_s : s \in \omega\}$ and $\{\tilde{G}_s : s \in \omega\}$ as subsequences of $\{F_s : s \in \omega\}$ and $\{\tilde{F}_s : s \in \omega\}$.

Observe that the sequence $t_0 < t_1 < \cdots$ from the previous lemma can be found computably in X. Let g^X be an X-computable function such that $g^X(n) \geq t_n$, but also that $g^X(n)$ bounds the witnesses for the inclusions $\tilde{F}_{t_n} \cdot F_{t_n} \subseteq E_n$ and $F_{t_n} \cdot \tilde{F}_{t_n} \subseteq E_n$. Since X is non-high, there is a computable function f not dominated by g^X.

For each $n \in \omega$, let s_n be the first stage s that looks true at $f(n)$ and for which we have witnessed $\tilde{F}_s \cdot F_s \subseteq E_n$ and $F_s \cdot \tilde{F}_s \subseteq E_n$. First notice that there is always such a stage s. Second, note that if $g^X(n) \leq f(n)$, then s_n is actually true.

Finally let $G_n = F_{s_n}$ and $\tilde{G}_n = \tilde{F}_{s_n}$. For the infinitely many n's with $g^X(n) \leq f(n)$ we have that s_n is true and hence G_n and $\tilde{G}_n$ are decreasing on those n, and they both have $\gg$-decreasing sub-sequences converging to the points g and g^{-1}. □

5.3. **The setup.** Let $\{G_s : s \in \omega\}$ and $\{\tilde{G}_s : s \in \omega\}$ be as in Lemma 20. We will build infinitely many points $z^i \in \mathrm{Orb}(x)$ that are pairwise not computably $\mathcal{G}$-equivalent. Together with each z^i, we will build a^i and $b^i \in \mathcal{G}$ such that $z^i = a^i \cdot x$ and $z^i = b^i \cdot y$ witnessing that $z^i \in \mathrm{Orb}(x)$. To get z^i and z^j not computably equivalent for $i \neq j$ we will consider each potentially computable $h \in \mathcal{G}$ (it could end up being partial), one at a time, and work towards getting $h \cdot z^i \neq z^j$.

Notation 21. For an index e, we say that *e is an index for an element of $\mathcal{G}$* if Φ_e is total and, for every n, $\Phi_e(n)$ is a code for a basic open ball $H_{e,n} \ll H_{e,n-1}$ (we may assume $H_{e,0} = \mathcal{G}$). When this is the case, we let h_e be the limit of these balls.

There will be infinitely many strategies $\mathbf{R}_0, \mathbf{R}_1, \ldots$ ordered by priority. There will be two types of strategies, but both types will share several common technical properties including the form of their input and output. This is a purely technical feature of the construction that will be convenient in showing that the points z^i are computable.

More specifically, for each $i \in \omega$, we build z^i by defining at each stage s, and for each strategy $\mathbf{R}_k$ that is active at s, a tuple $q^i_k[s]$ which is either of the form

$$q^i_k[s] = (Z^i_k, A^i_k, -, t^i_k)[s], \qquad\qquad \text{(copying } x\text{)}$$

or of the form

$$q^i_k[s] = (Z^i_k, -, B^i_k, t^i_k)[s], \qquad\qquad \text{(copying } y\text{)}$$

where

- $Z^i_k[s] \subseteq \mathcal{X}$ is a basic open ball trying to approximate z^i in the sense that we will have $z^i \in Z^i_k[s]$;
- $A^i_k[s] \subseteq \mathcal{G}$ is a basic open set trying to approximate $a^i \in \mathcal{G}$ in the sense that we will have $a^i \in A^i_k[s]$ unless $\mathbf{R}_k$ is *injured* (to be defined);
- $B^i_k[s] \subseteq \mathcal{G}$ is a basic open set trying to approximate $b^i \in \mathcal{G}$ in the sense that we will have $b^i \in B^i_k[s]$ unless $\mathbf{R}_k$ is injured; and
- $t^i_k[s] \in \omega$ is a stage that looks true (for both approximations) at stage s, stating that we currently believe in $G_{t^i_k[s]}$ and $\tilde{G}_{t^i_k[s]}$ in our approximations to g and g^{-1}.

In what follows, we omit the superscript ^{i} if it is clear from the context which i the tuple $q^i_k[s]$ corresponds to.

$$\begin{tikzcd}
x \arrow[r, bend left=10, "g\in G_{t_k}"] \arrow[d, "a\in A_k"'] & y \arrow[l, bend left=10, "g^{-1}\in\tilde{G}_{t_k}"] \arrow[dl, "b\in B_k"] \\
z \in Z_k &
\end{tikzcd}$$

Convention 22. When $q_k[s] = (Z_k, A_k, -, t_k)[s]$, we define $B_k[s] = A_k \cdot \tilde{G}_{t_k}$, and when $q_k[s] = (Z_k, -, B_k, t_k)[s]$, we define $A_k[s] = B_k \cdot G_{t_k}$. Recall that, by our hypothesis, the action is also effectively open. Thus, given indices for A_k and $\tilde{G}_{t_k}$ as effective open sets we can compute an index for $A_k \cdot \tilde{G}_{t_i}$, and similarly given indices for B_k and G_{t_k} we can compute an index for the open set $B_k \cdot G_{t_k}$. Thus, in both cases we will sometimes write

$$q_k[s] = (Z_k, A_k, B_k, t_k)[s],$$

where either A_k or B_k is in fact not basic open but merely effectively open.

At each stage s and for a fixed i, let k_s be (the number of) the last requirement active at stage s. We have a sequence $q_0[s]$, $q_1[s]$, ..., $q_{k_s}[s]$ which must satisfy for $k < k_s$:

(1) $Z_0[s] \gg Z_1[s] \gg \cdots \gg Z_{k_s}[s]$,
(2) $A_0[s] \supseteq A_1[s] \supseteq \cdots \supseteq A_{k_s}[s]$,
(3) $B_0[s] \supseteq B_1[s] \supseteq \cdots \supseteq B_{k_s}[s]$,
(4) $t_0[s] \leq t_1[s] \leq \cdots \leq t_{k_s}[s]$.

We define $Z^i[s]$ to be the least $Z_j[s]$ in the sequence, that is

$$Z^i[s] = Z^i_{k_s}[s].$$

We require that,

$$Z^i[0] \supseteq Z^i[1] \supseteq Z^i[2] \supseteq \cdots$$

as we need to make z^i computable. At the end, we will define z^i so that $\{z^i\} = \bigcap_s Z^i[s]$.

If none of the strategies below k_{s-1} require attention at s, then we will have $k_s = k_{s-1} + 1$ and $q^i_k[s] = q^i_k[s-1]$ for all $k < k_s$. Otherwise, $k_s \leq k_{s-1}$ is the index of the the highest priority strategy that requires attention at s. In this case, all the requirements $\mathbf{R}_k$ for $k > k_s$ will be re-initialized, while the values q^i_k for $k < k_s$ will stay unchanged, and $q^i_{k_s}$ will be re-defined.

This is a finite injury construction and we will show that each requirement $\mathbf{R}_k$ will eventually stop re-defining $q^i_k[s]$, which will stabilize at $(Z^i_k, A^i_k, B^i_k, t^i_k)$. We will have that for each i, $A^i_0 \supseteq A^i_1 \supseteq A^i_2 \supseteq \cdots$ and $B^i_0 \supseteq B^i_1 \supseteq B^i_2 \supseteq \cdots$, and that, for $a \in \bigcap_k A^i_k$, $z^i = a \cdot x$ and, for $b \in \bigcap_k B^i_k$, $z^i = b \cdot y$ witnessing that $z^i \in \mathrm{Orb}(x)$.

Each of the tuples $(Z, A, B, t) = (Z^i_k, A^i_k, B^i_k, t^i_k)[s]$ ever built in this construction must satisfy the following two properties:

(Ca) $A \cdot x \cap Z \neq \emptyset$.
(Cb) $B \cdot y \cap Z \neq \emptyset$.

Not only must the tuples satisfy these properties above, but we must have witnessed that they do. To witness that $A \cdot x \cap Z \neq \emptyset$ we must find basic open sets $\tilde{\mathcal{A}} \subseteq A$, $\tilde{X} \ni x$ and $\tilde{Z} \subseteq Z$ with $\tilde{\mathcal{A}} \cdot \tilde{X} \subseteq Z$ where $(\tilde{\mathcal{A}}, \tilde{X}, \tilde{Z})$ is listed in the effective representation of the group operation, and similarly for (Cb). Both conditions are clearly c.e.

5.4. **The strategies.** There will be two types of strategies denoted $\mathbf{CD}_{i,j,e}$, and $\mathbf{S}^x_i$ or $\mathbf{S}^y_i$. The strategy $\mathbf{CD}_{i,j,e}$ will be in charge of convergence and diagonalization, while $\mathbf{S}^x_i$ and $\mathbf{S}^y_i$ will be in charge of switching from copying y to x and from copying x to y, respectively.

Remark 23. The phrase *copying* x comes from the version of this proof for structures. Informally, we say that z^j is currently copying x if we have defined $A^i_{k+1} \ll A^i_k$, and thus we have a better current approximation to a_i such that $z^i = a_i \cdot x$. In this case, it refers to the fact that B is being defined out of A, so that any change in A will automatically produce a change in B. We note here that this definition of B through A is consistent with condition (Cb) (and symmetrically, our definition of A via B in the phase of copying y will be consistent with (Ca)). Indeed, if we assume that $g^{-1} \in \tilde{G}$, we have that if $\hat{A} \subseteq A$ satisfies $\hat{A} \cdot x \cap Z \neq \emptyset$, then $\hat{B}$, defined by $\hat{B} = \hat{A} \cdot \tilde{G}$, satisfies $\hat{B} \cdot y \cap Z \neq \emptyset$ because $\hat{B} \cdot y = \hat{A} \cdot \tilde{G} \cdot y \supseteq \hat{A} \cdot g \cdot y = \hat{A} \cdot x$ which intersects Z. This means that, while we are copying x, we can concentrate on defining the "A-side", and we know the "B-side" will be fine.

In the order of priorities, each strategy $\mathbf{CD}_{i,j,e}$ for $i, j, e \in \omega$ with $i \neq j$ will appear once. As we will see later, for it to work, $\mathbf{CD}_{i,j,e}$ needs that in its input z^i is copying x and z^j is copying y. Before $\mathbf{CD}_{i,j,e}$ we will have one occurrence of $\mathbf{S}^x_i$ and one occurrence of $\mathbf{S}^y_j$. The reason for that is rooted in the *preservation method* that we will use; as will become clear later, this technical assumption will also be convenient in showing that for each i the sequence $(A^i_k)_{k\in\omega}$ is a Δ^0_2-approximation to a point $a_i \in \mathcal{G}$, and thus we indeed have $z_i = a_i \cdot x \in \mathrm{Orb}(x)$.

5.4.1. *The convergence-diagonalization strategies* $\mathbf{CD}_{i,j,e}$. The strategy $\mathbf{CD}_{i,j,e}$ is split into three different substrategies $\mathbf{C}^x_i$, $\mathbf{C}^y_j$ and $\mathbf{D}_{i,j,e}$. The actions of the three different substrategies are different and independent, but we are grouping them because $\mathbf{D}_{i,j,k}$ needs to have the same priority strength as $\mathbf{C}^x_i$ and $\mathbf{C}^y_j$ to work.

The substrategy $\mathbf{C}^x_i$ is responsible for:

- Taking steps towards making z^i computable; and
- Taking steps towards making $A^i[s]$ an approximation of a point a^i.

The substrategy $\mathbf{C}^y_j$ is responsible for:

- Taking steps towards making z^j computable; and
- Taking steps towards making $B^j[s]$ an approximation to a point b^j.

Recall that h_e stands for the e-th potential computable point in $\mathcal{G}$. The requirement $\mathbf{D}_{i,j,k}$ is responsible for:

- Ensuring that if Φ_e is total, then $h_e \cdot z^i \neq z^j$.

Suppose that $\mathbf{CD}_{i,j,e}$ is the k-th requirement in the list of priorities, that is, that $\mathbf{CD}_{i,j,e} = \mathbf{R}_k$. It will receive from $\mathbf{R}_{k-1}$ two relevant inputs:

$$(Z^i_{k-1}, A^i_{k-1}, -, t^i_{k-1})[s] \quad \text{and} \quad (Z^j_{k-1}, -, B^j_{k-1}, t^j_{k-1})[s],$$

the former one copying x and the latter one copying y. The output of $\mathbf{CD}_{i,j,e}$ will not change who is copying what.

The requirement $\mathbf{D}_{i,j,k}$ is the one deciding when to require attention, and hence when to act and re-initialize weaker priority requirements, but that will be its only action. It will be $\mathbf{C}^x_i$ and $\mathbf{C}^y_j$ who will act defining the values of $(Z^i_k, A^i_k, -, t^i_k)[s]$ and $(Z^j_k, -, B^j_k, t^j_k)[s]$.

Action of $\mathbf{CD}_{i,j,e}$. The strategy $\mathbf{CD}_{i,j,e}$ will act the first time we reach it after it is initialized, and then whenever $\mathbf{D}_{i,j,e}$ requires attention (unless a higher priority strategy also requires attention at the same time). $\mathbf{D}_{i,j,e}$ will require attention whenever we see more evidence towards having $h_e \cdot z^i = z^j$. At each stage s, we define the *length of agreement* for $\mathbf{D}_{i,j,e}$ to be the largest n such that Φ_e converges on $0, 1, ..., n$ and is looking like an index for an element, h_e, in $\mathcal{G}$ so far, and we have witnessed that

$$H_{e,n} \cdot Z^i[s] \cap Z^j[s] \neq \emptyset,$$

where H_n is the basic open set with index $\Phi_e(n)$. We say that a stage s is an *expansionary stage* for $\mathbf{D}_{i,j,e}$ if the length of agreement reaches a new value higher than any value it had before. $\mathbf{CD}_{i,j,e}$ require attention at every expansionary stage for $\mathbf{D}_{i,j,e}$. If $\mathbf{CD}_{i,j,e}$ is the highest priority strategy requiring attention, it will act; otherwise some strategy of a higher priority will act and will re-initialize $\mathbf{CD}_{i,j,e}$.

When $\mathbf{CD}_{i,j,e}$ acts, it does the following: First, it re-initializes all strategies of lower priority. Second, for $\hat{k} < k$ and $h = i, j$, it sets $q^h_{\hat{k}}[s] = q^h_{\hat{k}}[s-1]$. Finally, it lets $\mathbf{C}^x_i$ and $\mathbf{C}^y_j$ act.

Let us now describe the action of $\mathbf{C}^x_i$. Let $(\tilde{Z}, \hat{A}, -, t)$ be, either the output of $\mathbf{CD}_{i,j,e}$ the last time it acted (i.e., $(Z_k[s-1], A_k[s-1], -, t_k[s-1])$) if it has not been initialized since, and let it be the output of $\mathbf{R}_{k-1}$ if this is the first time $\mathbf{CD}_{i,j,e}$ acts after initialization (i.e., $(Z_{k-1}[s-1], A_{k-1}[s-1], -, t_{k-1}[s-1])$. Note that $(\tilde{Z}, \hat{A}, -, t)$ satisfies condition (Ca). Let $\hat{Z} = Z_{k_{s-1}}[s-1]$. Since $(Z_{k_{s-1}}[s-1], A_{k_{s-1}}[s-1], -, t_{k_{s-1}})$ satisfies (Ca), and $\hat{A} \supseteq A_{k_{s-1}}[s-1]$, we have that $\hat{A} \cdot x \cap \hat{Z} \neq \emptyset$, and similarly for the B-side.

To define $A[s]$ and $Z[s]$, we search for basic open sets $A[s] \ll A[s-1]$ and $Z[s] \ll Z[s-1]$ with

$$A[s] \cdot x \cap Z[s] \neq \emptyset.$$

Recall that we then define $B[s] = A[s] \cdot \tilde{G}_t$ provided we see confirmation that

$$A[s] \cdot \tilde{G}_t \cdot x \cap Z[s] \neq \emptyset.$$

As we have already noted above, if $\tilde{G}_t$ is true and contains g^{-1}, we will eventually see this is the case because $B[s] \cdot y = A[s] \cdot \tilde{G}_t \cdot y \supseteq A[s] \cdot g^{-1} \cdot y = A[s] \cdot x$ which intersects $Z[s]$. However, while waiting for this confirmation we might discover that $\tilde{G}_t$ was not true. In that case, the strategy that defined $t_k[s]$, which has stronger priority, is now requiring attention and $\mathbf{D}_{i,j,e}$ will be re-initialized before even defining $(Z^i, A^i, B^i, t^i)[s]$.

Let us now show how $\mathbf{CD}_{i,j,e} = \mathbf{R}_k$ works. Suppose $\mathbf{R}_{k-1}$ eventually stops requiring attention. If there is a later stage when $\mathbf{CD}_{i,j,e}$ stops requiring attention then it is because we have reached a maximum in the length of agreement and hence $h_e \cdot z^i \neq z^j$. Suppose towards a contradiction that

$\mathbf{CD}_{i,j,e}$ acts infinitely often. This would mean that we have infinitely many expansionary stages and that the length of agreement goes to infinity, which would imply that $h_e \cdot z^i = z^j$. Since the requirement $\mathbf{C}^x_i$ is acting infinitely often, we have that the sequence $\{\mathcal{A}^i_k[s] : s \in \omega\}$ converges to a computable point a^i. The same way we get a computable point b^j. Thus, we will have $y = (b^j)^{-1} \cdot h_e \cdot a^i \cdot x$, with $(b^j)^{-1} \cdot h_e \cdot a^i$ being computable, contradicting that x and y are not computably equivalent. Therefore, if $\mathbf{D}_{i,j,e}$ eventually stops being re-initialized, it will eventually stop requiring attention and we will have that either $z^j \neq h_e \cdot z^i$ or that e is not an index for an element of $\mathcal{G}$.

5.4.2. *The "switch" requirement.* For each i, we will have infinitely many strategies $\mathbf{S}^x_i$ and $\mathbf{S}^y_i$. The job of $\mathbf{S}^x_i$ is to switch from z^i copying y to z^i copying x, and the one of $\mathbf{S}^y_i$ is to switch from z^i copying x to copying y. In other words, the job of $\mathbf{S}^x_i$ is to make sure its outcome is of the form

$$q^i_k[s] = (Z^i_k, A^i_k, -, t^i_k)[s], \qquad \text{(copying } x\text{)}$$

instead of of the form

$$q^i_k[s] = (Z^i_k, -, B^i_k, t^i_k)[s], \qquad \text{(copying } y\text{)},$$

while the job of $\mathbf{S}^y_i$ is the opposite. Since $\mathbf{D}_{i,j,e}$ needs z^i to be copying x and z^j to be copying y, before each strategy $\mathbf{CD}^{i,j,e}$ we put a strategy $\mathbf{S}^x_i$ and a strategy $\mathbf{S}^x_j$.

Let us describe the action of $\mathbf{S}^x_i$. Suppose $\mathbf{R}_k$ is an instance of $\mathbf{S}^x_i$. If the input for $\mathbf{S}^x_i$ is of the form $(Z, A, -, t)$, then there is nothing to switch and nothing for $\mathbf{S}^x_i$ to do. So, let us assume

$$\text{the input for } \mathbf{S}^x_i \text{ is } (Z, -, B, t).$$

By the input for $\mathbf{S}^x_i$ we mean the following: B and t are given to us by their higher priority requirement, i.e., $B = B^i_{k-1}[s]$ and $t = t^i_{k-1}[s]$; and Z is given to us by the outcome at the previous stage, i.e., $Z = Z^i[s-1] = Z^i_{k_{s-1}}[s-1]$. Recall that even if $k_{s-1} > k-1$, we still have that Z and B satisfy (Cb). As before, we let $A = A^i_{k-1}[s] = B \cdot \tilde{G}_t$.

We want to find t' true, and $\hat{A} \subseteq A$ such that, if we let $\hat{B} = \hat{A} \cdot G_{t'}$ then $(\hat{Z}, \hat{A}, \hat{B}, t')$ satisfies (Ca) and $\hat{B} \subseteq B$. Note we can only search for t' which look true, and if we later notice we chose a t' that was not really true, will have to search again.

Recall that $(E_t)_{t\in\omega}$ is the computable fast Cauchy name of $e_{\mathcal{G}}$ that we fixed at the beginning of the construction. When $\mathbf{S}^x_i$ acts it choses a member of the original dense subset $b' \in B$, and a number $t' \geq t$ such that

(S1) $b' \cdot y \in Z$,
(S2) $b' \cdot E_{t'} \subset B$, and
(S3) t' looks true at the current stage.

We remark that b' is only useful at this step of the construction and is going to be forgotten later. Then define

$$\hat{A} = b' \cdot \tilde{G}_{t'} = \bigcup \{U : \exists W \text{ a basic open ball around } (b')^{-1} \ (W \cdot U \subseteq \tilde{G}_{t'})\}.$$

We then have that

$$\hat{B} = \hat{A} \cdot G_{t'} = b' \cdot \tilde{G}_{t'} \cdot G_{t'} \subseteq b' \cdot E_{t'} \subseteq B.$$

Note that t looks true at t', as otherwise it would not look true at the current stage and some higher priority requirement would have required attention. We remark here that, above, we have used that $\tilde{G}_{t'} \cdot G_{t'} \subseteq E_{t'}$ and this is why we could state (S2) in terms of $E_{t'}$ and not in terms of $\tilde{G}_{t'} \cdot G_{t'}$ — this is one of the key uses of having a non-high point g.

We have that $\tilde{G}_t \supseteq \tilde{G}_{t'}$, and hence

$$\hat{A} = b' \cdot \tilde{G}_{t'} \subseteq B \cdot \tilde{G}_t = A.$$

Since $b' \cdot y \in Z$, we have that unless $g \notin G_{t'}$ or $g^{-1} \notin \tilde{G}_{t'}$, (Ca) and (Cb) hold. Thus, either we will eventually find confirmation of (Ca) and (Cb), or we will find out t' is not true in which cases we need new t' and b'.

At a later stage, if we ever see that t' is not true, then $\mathbf{S}_i^x$ *requires attention* to redefine its outcome. At that later stage when we act again, the input is going to be of the form $(\tilde{Z}, -, B, t)$, where B and t are the same as the first time $\mathbf{S}_i^x$ acted after initialization, but $\tilde{Z}$ is new and smaller than the previous Z (recall that the sequence $\{Z^i[s] : s \in \omega\}$ needs to be decreasing). This means that we might need to change our choice of b' to b'' so that $b'' \cdot y \in \tilde{Z}$, and then we might have to change t' to a larger t'' to get $b'' \cdot E_{t''} \subseteq B$. If this new t'' is not true, at some later stage we will have to change it again to t''', etc. We need to make sure that eventually we will hit an actual true stage. In fact, what we need to make sure is that if there is a real true stage between t' and t'', we do not skip it.

The next technical feature is crucial in the construction. After the first time $\mathbf{S}_i^x$ acts after initialization, it silently keeps track of the new changes in Z to make sure that if it needs to act again, it will not skip a true stage. It does it as follows. Suppose that at a later stage $\tilde{s}$ some lower priority strategy wants to define $Z^i[\tilde{s}]$ to be some set $\tilde{Z}$. Silently, and without interfering with the rest of the construction, $\mathbf{S}_i^x$ will look for a new $\tilde{b}$ and its corresponding $\tilde{t}$ so that $\tilde{b} \cdot y \in \tilde{Z}$, $\tilde{b} \cdot E_{\tilde{t}} \subseteq B$ and $\tilde{t}$ looks true. Such $\tilde{b}$ and $\tilde{t}$ must exist, so $\mathbf{S}_i^x$ will eventually find them. At that point, $\mathbf{S}_i^x$ will then check that its current t' is still true up to $\tilde{t}$ before allowing $Z^i[s]$ to be re-defined. If t' still looks true at $\tilde{t}$, there is nothing for $\mathbf{S}_i^x$ to worry about and it can let the other requirement do its job. If t' does not look true anymore, $\mathbf{S}_i^x$ requires attention and acts before Z is re-defined.

Claim 1. After $\mathbf{S}_i^x$ is re-initialized for the last time, $\mathbf{S}_i^x$ eventually will output a t' which is a true stage and it will stop requiring attention thereafter.

Proof of the Claim. Suppose $\mathbf{S}_i^x$ acts at a stage s' getting output t' which is not true, and that $\hat{\mathbf{t}}$ is the next true stage after t'. At some stage, say $\tilde{s}$, we will notice t' is not true and $\mathbf{S}_i^x$ will act again. We will prove that, at that time, $\mathbf{S}_i^x$ will output $\tilde{t} \leq \hat{\mathbf{t}}$. Therefore, either it will output the true $\hat{\mathbf{t}}$ and it will be done, or it will output $\tilde{t} < \hat{\mathbf{t}}$ but we will be closer to finding $\hat{\mathbf{t}}$ at a later stage.

At $\tilde{s} - 1$ we had not yet noticed t' is not true. This implies two things. First, we had a potential pair b'', t'' that worked for $Z^i[\tilde{s} - 1]$, i.e., they satisfied $b'' \cdot y \in Z^i[\tilde{s} - 1]$, $b'' \cdot E_{t''} \subseteq B$. Second, we must have $\tilde{s} \leq \hat{\mathbf{t}}$. When $\mathbf{S}_i^x$ acts at $\tilde{s}$, Z does not change, that is, $Z^i[\tilde{s}] = Z^i[\tilde{s} - 1]$. That means that the pair $b'', \hat{\mathbf{t}}$ also satisfies (S1)-(S3). Therefore, if $\mathbf{S}_i^x$ does not choose this pair at $\tilde{s}$, it choses $t'' < \hat{\mathbf{t}}$. □

5.5. **Finalising the proof.** At stage s we let the strategies act according to their instructions. We have already argued above that the injury is merely finite (which was the main meat of the proof), thus the verification is just a straightforward inductive argument.

References

[BHW08] Vasco Brattka, Peter Hertling, and Klaus Weihrauch. A tutorial on computable analysis. In *New computational paradigms*, pages 425–491. Springer, New York, 2008.

[Eff65] Edward G. Effros. Transformation groups and C^*-algebras. *Ann. of Math. (2)*, 81:38–55, 1965.

[EG00] Yuri L. Ershov and Sergei S. Goncharov. *Constructive models*. Siberian School of Algebra and Logic. Consultants Bureau, New York, 2000.

[Gao09] Su Gao. *Invariant descriptive set theory*, volume 293 of *Pure and Applied Mathematics (Boca Raton)*. CRC Press, Boca Raton, FL, 2009.

[Gon75] S. S. Gončarov. Selfstability, and computable families of constructivizations. *Algebra i Logika*, 14(6):647–680, 727, 1975.

[Gon80] S. S. Gončarov. The problem of the number of nonautoequivalent constructivizations. *Algebra i Logika*, 19(6):621–639, 745, 1980.

[Kec95] Alexander S. Kechris. *Classical descriptive set theory*, volume 156 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1995.

[LE65] E. G. K. Lopez-Escobar. An interpolation theorem for denumerably long formulas. *Fund. Math.*, 57:253–272, 1965.

[Mar66] Donald A. Martin. Classes of recursively enumerable sets and degrees of unsolvability. *Z. Math. Logik Grundlagen Math.*, 12:295–310, 1966.

[McC02] Charles F. D. McCoy. Finite computable dimension does not relativize. *Arch. Math. Logic*, 41(4):309–320, 2002.

[Mon] Antonio Montalbán. A robuster scott rank. To appear in Proc.Amer.Math.Soc.

[RN64] C. Ryll-Nardzewski. On Borel measurabilty of orbits. *Fund. Math.*, 56:129–130, 1964.

[Ven92] Yu. G. Ventsov. The effective choice problem for relations and reducibilities in classes of constructive and positive models. *Algebra i Logika*, 31(2):101–118, 220, 1992.

[Wei01] Klaus Weihrauch. On computable metric spaces Tietze-Urysohn extension is computable. In *Computability and complexity in analysis (Swansea, 2000)*, volume 2064 of *Lecture Notes in Comput. Sci.*, pages 357–368. Springer, Berlin, 2001.

[WG09] Klaus Weihrauch and Tanja Grubba. Elementary computable topology. *J.UCS*, 15(6):1381–1422, 2009.

The Institute of Natural and Mathematical Sciences, Massey University, Auckland, New Zealand

Department of Mathematics, University of California, Berkeley, USA
Email address: antonio@math.berkeley.edu
URL: www.math.berkeley.edu/∼antonio