COMPUTABLE SCOTT SENTENCES AND THE FRIEDMAN–STANLEY EMBEDDING

DAVID GONZALEZ, AND JULIA KNIGHT

Abstract. Friedman and Stanley [9] developed the notion of *Borel reducibility* and illustrated its use in comparing classification problems for some familiar classes of countable structures. For many embeddings, the fact that the embedding is 1–1 on isomorphism types is explained by the existence of simple formulas that, uniformly, interpret the input structure in the output structure. For the embeddings of graphs in trees, and in linear orderings, there is no uniform interpretation [16, 20]. We focus on a version of the Friedman–Stanley embedding from [16] that takes each structure $\mathcal{A}$ for the language of graphs to a labeled tree $T_{\mathcal{A}}$. Gonzalez and Rossegger [13] showed that this embedding preserves Scott complexity. We refine this result, showing that for an X-computable ordinal, if one of $\mathcal{A}$, $T_{\mathcal{A}}$ has a computable infinitary Scott sentence, then so does the other, and the complexities match. Let $\mathbb{T}$ be the class of labeled trees isomorphic to those in the range of the embedding, and let $\mathbb{T}^{\alpha}$ be the subclass consisting of structures of Scott rank at most α. It follows from results of Gao [10] that $\mathbb{T}$ is not Borel. We show that for each α, $\mathbb{T}^{\alpha}$ is Borel. In fact, if α is an X-computable ordinal, then $\mathbb{T}^{\alpha}$ is complete X-effective $\Pi_{2\alpha+2}$.

§1. Introduction. A good deal of work, in different branches of mathematics, involves trying to "classify" the objects in some class up to some notion of sameness, using relatively simple invariants. For a countable language L, $Mod(L)$ denotes the class of L-structures with universe ω. Our objects are elements of $Mod(L)$, for some countable language L. Our classes are subclasses of $Mod(L)$, always closed under isomorphism. Our notion of sameness is isomorphism. The class of $\mathbb{Q}$-vector spaces is nicely classified by dimension. Nobody would try to classify directed graphs. Of course, there are only countably many isomorphism types of $\mathbb{Q}$-vector spaces, while there are $2^{\aleph_0}$ isomorphism types of countable directed graphs. Friedman and Stanley [8] developed the notion of "Borel reducibility" to compare classification problems for classes of countable structures, using more than just cardinality.

Definition 1. For classes $K \subseteq Mod(L)$ and $K' \subseteq Mod(L')$, we say that K is *Borel reducible* to K', and we write $K \leq_B K'$, if there is a Borel function $\Phi : K \to K'$ such that for $\mathcal{A}, \mathcal{B} \in K$, $\mathcal{A} \cong \mathcal{B}$ iff $\Phi(\mathcal{A}) \cong \Phi(\mathcal{B})$.

In [8], Friedman and Stanley located various familiar classes of structures under Borel reducibility, showing that undirected graphs, fields, 2-step nilpotent groups, and linear orderings lie on top, while $\mathbb{Q}$-vector spaces and Abelian p-groups lie

Received May 6, 2026 UTC.
2020 *Mathematics Subject Classification.* Primary 03E15, 03D55, 03C57.
Key words and phrases. Borel reduction, Scott sentence, computable structure, infinitary logic.

© The Author(s), 2026. Published by Cambridge University Press on behalf of The Association for Symbolic Logic. This is an Open Access article, distributed under the terms of the Creative Commons Attribution licence (https://creativecommons.org/licenses/by/4.0/), which permits unrestricted re-use, distribution, and reproduction in any medium, provided the original work is properly cited.

0022-4812/00/0000-0000
DOI:10.1017/jsl.2026.10247

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

strictly below the top. There has been a great deal of work locating further interesting classes. In particular, for the class $TFAb_n$ of torsion-free Abelian groups of rank n, Hjorth [16] and Thomas [26] showed that $TFAb_n <_B TFAb_{n+1}$. Paolini and Shelah [24] showed that torsion-free Abelian groups of infinite rank lie on top. For some of their results, Friedman and Stanley used known embeddings. To show that linear orderings and trees lie on top, they gave new embeddings. We focus on a version of the Friedman–Stanley embedding, described in [15], that, for a finite relational language L, takes each $\mathcal{A} \in Mod(L)$ to a labeled tree $T_{\mathcal{A}}$.

Let L be a fixed finite relational language, such as the language of graphs. Let $\mathcal{A} \in Mod(L)$, and let $T_{\mathcal{A}}$ be the corresponding tree structure. In [12] it is shown that $\mathcal{A}$ and $T_{\mathcal{A}}$ have the same Scott rank; hence, if one of the structures has a $\Pi_{\alpha+1}$ Scott sentence, then so does the other. Gonzalez and Rossegger asked whether the same is true for computable infinitary Scott sentences. In Section 2, we give a positive answer.

THEOREM A. *For all $\mathcal{A} \in Mod(L)$, if one of $\mathcal{A}$, $T_{\mathcal{A}}$ has a computable $\Pi_{\alpha+1}$ Scott sentence, then so does the other.*

REMARK. Our proof shows more. By analogy with the work in [12], we see that, if α is a computable limit ordinal and $\mathcal{A}$ has a computable Π_α Scott sentence, then so does $T_{\mathcal{A}}$.

Let $\mathbb{T}$ be the class of labeled trees isomorphic to one of the form $T_{\mathcal{A}}$ for $\mathcal{A} \in Mod(L)$. Let $\mathbb{T}^\alpha$ consist of the elements of $\mathbb{T}$ of rank at most α. The full class $\mathbb{T}$ is not Borel. Gao [9] gave results implying that for any embedding of graphs in linear orderings, the class of orderings isomorphic to those in the range is not Borel. The same reasoning shows that $\mathbb{T}$ is not Borel (see [11] or [12]). In Section 3, we show that for all countable ordinals α, $\mathbb{T}^\alpha$ *is* Borel, and we give the precise complexity. In Section 4, we show that our given complexity is the best possible.

THEOREM B. *For all* X*-computable ordinals $\alpha \geq 1$, $\mathbb{T}^\alpha$ is complete* X*-effective* $\Pi_{2\alpha+2}$.

1.1. Background. In the remainder of the introduction, we give some background. We describe the embedding $\mathcal{A} \to T_{\mathcal{A}}$ precisely. We recall infinitary formulas, $L_{\omega_1\omega}$-formulas, and computable infinitary formulas. We state basic results of Scott [25] and Montalbán [21] on Scott sentences and Scott rank. We define the standard back-and-forth relations. We indicate how $Mod(L)$ may be identified with Cantor space. Finally, we describe variants of Borel reducibility for which a Pullback Theorem holds [18].

1.1.1. Tree of tuples. In this section, we describe (for a fixed finite relational language L) the embedding $\mathcal{A} \to T_{\mathcal{A}}$ in a precise way, following [15] (see also [12, p. 4]). Let X be the set of variables $x_1, x_2, \ldots$ We assign to each L-structure $\mathcal{A}$, with universe ω, a tree structure $T_{\mathcal{A}}$. The language of the tree structures, L^*, consists of a unary function symbol p (the predecessor function) and unary relation symbols U_D, where, for some n, D is a finitary quantifier-free L-formula assigning an atomic type to the tuple of variables $(x_1, \ldots, x_n)$. For each relation symbol $R \in L$, say of

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

arity m, and each m-tuple of variables $\bar{z}$ chosen from $x_1, \ldots, x_n$, D has a conjunct $\pm R(\bar{z})$. In addition, D has conjuncts $x_i \neq x_j$ for $1 \leq i < j \leq n$.

For each tuple $\bar{a} = (a_1, \ldots, a_n)$ of distinct elements of $\mathcal{A}$, let $\bar{x} = (x_1, \ldots, x_n)$ be the corresponding tuple $(x_1, \ldots, x_n)$ of variables from X, and let $D_{\bar{a}}(\bar{x})$ be the formula (of the form described above) that gives the atomic type of $\bar{a}$. Each node σ at level n of $T_{\mathcal{A}}$ represents some n-tuple $\bar{a} = (a_1, \ldots, a_n)$. For each σ representing this $\bar{a}$, the label is $U_{D_{\bar{a}}}$, and $p(\sigma)$ represents the tuple $(a_1, \ldots, a_n - 1)$. We define $T_{\mathcal{A}}$ more precisely below.

Definition 2 (Tree of tuples $T_{\mathcal{A}}$). At level 0, $T_{\mathcal{A}}$ has a unique node λ, representing $\emptyset$. The nodes at level n represent tuples $\bar{a}$ of length n. A node representing a tuple $\bar{a}$ is labeled with a code for the atomic diagram of $\bar{a}$. For each node σ representing an n-tuple $\bar{a}$, and each further element a', σ has infinitely many extensions representing the tuple $(\bar{a}, a')$.

1.1.2. Formulas of $L_{\omega_1\omega}$. The infinitary logic $L_{\omega_1\omega}$ allows countably infinite disjunctions and conjunctions, but only finite strings of quantifiers. Bringing negations inside, we get a normal form in which $\bigvee \exists$ alternates with $\bigwedge \forall$. We classify formulas in this normal form as Σ_α or Π_α for countable ordinals α.

1. $\varphi(\bar{x})$ is Σ_0 and Π_0 if it is finitary quantifier-free.
2. For $\alpha \geq 1$,
   (a) $\varphi(\bar{x})$ is Σ_α if it has form $\bigvee_i \exists \bar{u}_i \psi_i(\bar{x}, \bar{u}_i)$, where each ψ_i is Π_{β_i} for some $\beta_i < \alpha$;
   (b) $\varphi(\bar{x})$ is Π_α if it has form $\bigwedge_i \forall \bar{u}_i \psi_i(\bar{x}, \bar{u}_i)$, where each ψ_i is Σ_{β_i} for some $\beta_i < \alpha$.

1.1.3. Computable infinitary formulas. Computable infinitary formulas are $L_{\omega_1\omega}$ formulas in which the infinite disjunctions and conjunctions are over c.e. sets. To make this precise, we need to assign indices to the formulas, as in [4] or [23]. We classify computable infinitary formulas as computable Σ_α or computable Π_α for computable ordinals α.

1.1.4. Scott complexity. In this section, we recall the Scott Isomorphism Theorem [25] and define *Scott sentences*. We also state a result of Montalbán [21], which yields the notion of Scott rank we will use.

Theorem 1.1 (Scott Isomorphism Theorem). *For each countable structure $\mathcal{A}$ for a countable language* L, *there is a sentence φ of $L_{\omega_1\omega}$ whose countable models are just the isomorphic copies of $\mathcal{A}$.*

Definition 3. A *Scott sentence* for $\mathcal{A}$ is a sentence φ that characterizes $\mathcal{A}$ up to isomorphism, as in Scott's Theorem.

Theorem 1.2 (Montalbán). *Let $\alpha \geq 1$ be a countable ordinal. For $\mathcal{A} \in Mod(L)$, the following are equivalent*:

1. *$\mathcal{A}$ has a $\Pi_{\alpha+1}$ Scott sentence*;
2. *for each tuple $\bar{a}$, the* orbit; *i.e., the set of tuples $\bar{b}$ automorphic to $\bar{a}$, is defined by a Σ_α formula*;
3. *for each tuple $\bar{a}$, there is a Σ_α-formula that implies (in $\mathcal{A}$) all Π_α formulas satisfied by $\bar{a}$.*

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

The definition of "Scott rank" given below is not due to Scott.[1] It comes from Montalbán's paper [21].

DEFINITION 4. The *Scott rank* of $\mathcal{A}$, denoted by $SR(\mathcal{A})$, is the least α such that the orbits of all tuples are defined by Σ_α formulas. If α is a limit ordinal, then for a structure $\mathcal{A}$ of Scott rank α, the orbit of each tuple is defined by a Σ_β formula for some $\beta < \alpha$. The structure $\mathcal{A}$ will have a $\Pi_{\alpha+1}$ Scott sentence. It may or may not have a Scott sentence that is Π_α.

Alvir, Knight, and McCoy [1] gave the following partial effective version of Montalbán's result.

THEOREM 1.3. *If $\mathcal{A}$ has a computable Π_α Scott sentence, then the orbit of each tuple is defined by a computable Σ_β formula for some $\beta < \alpha$.*

1.1.5. Back-and-forth relations. We define the standard back-and-forth relations $\leq_\alpha$. We are mainly interested in tuples from a single structure, although in the definition, we allow tuples from two different structures.

DEFINITION 5. Let $\mathcal{A}, \mathcal{B}$ be structures for the same countable language. First, suppose that $\bar{a}$ in $\mathcal{A}$ and $\bar{b}$ in $\mathcal{B}$ are tuples of the same length.

1. $(\mathcal{A}, \bar{a}) \leq_1 (\mathcal{B}, \bar{b})$ if all existential formulas satisfied by $\bar{b}$ in $\mathcal{B}$ are satisfied by $\bar{a}$ in $\mathcal{A}$.
2. For $\alpha > 1$, $(\mathcal{A}, \bar{a}) \leq_\alpha (\mathcal{B}, \bar{b})$ if for each $\bar{d}$ and each $1 \leq \beta < \alpha$, there exists $\bar{c}$ such that $(\mathcal{B}, \bar{b}, \bar{d}) \leq_\beta (\mathcal{A}, \bar{a}, \bar{c})$.

If $\bar{b}$ is longer than $\bar{a}$, where $\bar{b}'$ is the initial segment of $\bar{b}$ of the same length as $\bar{a}$, then $(\mathcal{A}, \bar{a}) \leq_\alpha (\mathcal{B}, \bar{b})$ if $(\mathcal{A}, \bar{a}) \leq_\alpha (\mathcal{B}, \bar{b}')$.

Carol Karp [17] proved the following.

THEOREM 1.4. *Let $\bar{a}$ in $\mathcal{A}$ and $\bar{b}$ in $\mathcal{A}$ be tuples of the same length. For all countable ordinals $\alpha \geq 1$, the following are equivalent*:

1. $(\mathcal{A}, \bar{a}) \leq_\alpha (\mathcal{B}, \bar{b})$;
2. *all Π_α formulas satisfied by $\bar{a}$ in $\mathcal{A}$ are satisfied by $\bar{b}$ in $\mathcal{B}$*;
3. *all Σ_α formulas satisfied by $\bar{b}$ in $\mathcal{B}$ are satisfied by $\bar{a}$ in $\mathcal{A}$*;

The following is well-known (see [3]).

LEMMA 1.5. *Let* L *be a countable language, and let $\mathcal{A}$ be a countable* L*-structure. For each countable ordinal α and each tuple $\bar{a}$ in $\mathcal{A}$, there is a Π_α-formula $\varphi(\bar{x})$ that defines in $\mathcal{A}$ the set of $\bar{b}$ such that $(\mathcal{A}, \bar{a}) \leq_\alpha (\mathcal{A}, \bar{b})$.*

PROOF SKETCH. For each tuple $\bar{b}$ such that $(\mathcal{A}, \bar{a}) \not\leq_\alpha (\mathcal{A}, \bar{b})$, choose a Π_α formula true of $\bar{a}$ and not true of $\bar{b}$. The conjunction of the chosen formulas is the desired $\varphi(\bar{x})$. ⊣

[1] In [25], Scott defined a nested family of equivalence relations on tuples $\sim^\alpha$ for countable ordinals α. He showed that for each countable structure $\mathcal{A}$, the $\sim^\alpha$-equivalence classes are definable, and there is a countable ordinal α such that $\sim^\alpha$ implies automorphism.

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

We add one more to the list of conditions in Montalbán's Theorem.

LEMMA 1.6. *Let $\mathcal{A}$ be a countable structure for a countable language* L. *For a countable ordinal α, the following are equivalent*:

1. *$\mathcal{A}$ has Scott rank at most α*;
2. *for each tuple $\bar{a}$, there exist $\beta < \alpha$ and a tuple $\bar{b}$ such that for all tuples $\bar{a}'$ of the same length as $\bar{a}$ and $\bar{b}'$ of the same length as $\bar{b}$, $(\mathcal{A}, \bar{a}, \bar{b}) \leq_\beta (\mathcal{A}, \bar{a}'\bar{b}')$ implies $(\mathcal{A}, \bar{a}) \leq_\alpha (\mathcal{A}, \bar{a}')$.*

PROOF. (1) $\Rightarrow$ (2): For each tuple in $\mathcal{A}$, there is a Σ_α formula $\varphi(\bar{x})$ that defines the orbit of $\bar{a}$. We may suppose that $\varphi(\bar{x})$ has form $(\exists \bar{u})\psi(\bar{x}, \bar{u})$, where $\psi(\bar{x}, \bar{u})$ is Π_β for some $\beta < \alpha$. Take $\bar{b}$ such that $\mathcal{A} \models \psi(\bar{a}, \bar{b})$. If $(\mathcal{A}, \bar{a}, \bar{b}) \leq_\beta (\mathcal{A}, \bar{a}', \bar{b}')$, then $\mathcal{A} \models \psi(\bar{a}', \bar{b}')$. Then $(\mathcal{A}, \bar{a}) \leq_\alpha (\mathcal{A}, \bar{a}')$ and the two tuples are automorphic.

(2) $\Rightarrow$ (1): Take $\bar{a}$ in $\mathcal{A}$. By hypothesis, we have $\beta < \alpha$ and $\bar{b}$ such that $(\mathcal{A}, \bar{a}, \bar{b}) \leq_\beta (\mathcal{A}, \bar{a}', \bar{b}')$ implies $(\mathcal{A}, \bar{a}) \leq_\alpha (\mathcal{A}, \bar{a}')$. By Lemma 1.5, there is a Π_α-formula $\psi(\bar{x}, \bar{u})$ such that $\mathcal{A} \models \psi(\bar{a}', \bar{b}')$ iff $(\mathcal{A}, \bar{a}, \bar{b}) \leq_\beta (\mathcal{A}, \bar{a}', \bar{b}')$. Then the Σ_α-formula $(\exists \bar{u})\psi(\bar{x}, \bar{u})$ implies all Π_α formulas true of $\bar{a}$, so it defines the orbit. ⊣

1.1.6. Identifying $Mod(L)$ with $\mathcal{C}$. For a countable language L, we may identify $Mod(L)$ with Cantor space $\mathcal{C} = 2^\omega$, showing that $Mod(L)$ is a Polish space. There are different ways to do this. Below we describe the one that seems most natural to us. We treat elements of ω as constants. For a countable relational language L, let $(d_n)_{n\in\omega}$ be a *Friedberg* list (no repetitions) of all sentences of form $R(\bar{a})$, where $R \in L$ and $\bar{a}$ is a tuple of elements of ω of length equal to the arity of R. If L is computable, we may take this list to be computable. For each $f \in \mathcal{C}$, let $\mathcal{A}_f$ be the structure in $Mod(L)$ such that

$$\mathcal{A}_f \models \begin{cases} d_n & \text{if } f(n) = 1 \\ \neg d_n & \text{if } f(n) = 0 \end{cases}.$$

On $\mathcal{C}$, there is a natural topology, where for $\sigma \in 2^{<\omega}$, the set of extensions N_σ is a basic clopen neighborhood. Moreover, $\mathcal{C}$ is a Polish space, since the natural topology is obtained from the metric $d(f, g) = \frac{1}{2^n}$, where n is least such that $f(n) \neq g(n)$. There is a natural topology on $Mod(L)$, generated by the open neighborhoods $Mod(\varphi)$, where φ is a finitary quantifier-free sentence. The identification above yields the same topology. Of course, under the identification, the metric also transfers to $Mod(L)$, witnessing that it is a Polish space.

1.1.7. Embeddings that preserve complexity. We consider variants of the notion of Borel embedding that satisfy a "Pullback" theorem. The following definition is from [5].

DEFINITION 6 (Turing computable embedding). Let L and L' be computable languages and take classes $K \subseteq Mod(L)$ and $K' \subseteq Mod(L')$. A *Turing computable embedding* of K in K' is a Turing operator $\Phi : K \to K'$ such that $\mathcal{A} \cong \mathcal{B}$ iff $\Phi(\mathcal{A}) \cong \Phi(\mathcal{B})$. We write $K \leq_{tc} K'$ if there is such an embedding.

The result below is from [18].

THEOREM 1.7 (Pullback Theorem). *Suppose $K \leq_{tc} K'$, where $K \subseteq Mod(L)$ and $K' \subseteq Mod(L')$. For any computable infinitary L'-sentence φ, we can effectively find a*

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

computable infinitary L*-sentence* φ^**, of the same complexity, such that for all* $\mathcal{A} \in K$, $\Phi(\mathcal{A}) \models \varphi$ *iff* $\mathcal{A} \models \varphi^*$.

The notion of Turing computable embedding relativizes.

FACT. *An embedding is continuous iff it is* X*-computable for some* X.

The Pullback Theorem relativizes to give the following.

COROLLARY 1.8. *Let* Φ *be a continuous embedding of* K *in* K'*, where* $K \subseteq Mod(L)$ *and* $K' \subseteq Mod(L')$*. Then for any* $L_{\omega_1\omega}$ *sentence* φ*, there is an* $L_{\omega_1\omega}$ *sentence* φ^**, of the same complexity, such that* $\Phi(\mathcal{A}) \models \varphi$ *iff* $\mathcal{A} \models \varphi^*$.

The following is well-known (see, e.g., [22, Theorem VI.27 and Lemma VI.29]).

THEOREM 1.9. *Let* L *be a computable relational language. There is a Turing computable embedding* Φ *of* $Mod(L)$ *in the class of undirected graphs. Moreover,*

1. *the class* K *consisting of graphs* G *such that for some* $\mathcal{A} \in Mod(L)$, $G \cong \Phi(\mathcal{A})$ *is effective* Π_2;
2. *there are existential formulas that for all* $\mathcal{A} \in Mod(L)$*, define a copy of* $\mathcal{A}$ *in* $\Phi(\mathcal{A})$*; we have formulas* $d_U(x)$*, defining the universe of the copy, and, for* $R \in L$, $d_R(\bar{x})$ *defining the interpretation of* R*, and* $d_{\neg R}(\bar{x})$ *defining the interpretation of* $\neg R$.

Hence, there is a Turing computable embedding Ψ *of* K *in* $Mod(L)$ *such that for all* $\mathcal{A} \in Mod(L)$, $\Psi(\Phi(\mathcal{A})) \cong \mathcal{A}$.

Using this theorem, we easily obtain the following.

PROPOSITION 1.10. *Let* L *be a computable relational language. There is a Turing computable embedding* Φ *of* $Mod(L)$ *in the class of graphs such that for* $\mathcal{A} \in Mod(L)$*, the Scott sentences for* $\mathcal{A}$ *and* $\Phi(\mathcal{A})$ *have the same complexity.*

PROOF. Let Φ, K, and Ψ be as in the previous theorem. Take $\mathcal{A} \in Mod(L)$, and let $G = \Phi(\mathcal{A})$. First, let φ be a Scott sentence for G. Let φ^* be the Φ-pullback of φ, of the same complexity as φ. For $\mathcal{A}' \in Mod(L)$, we have $\mathcal{A}' \models \varphi^*$ iff $\Phi(\mathcal{A}') \models \varphi$ iff $\Phi(\mathcal{A}') \cong G$ iff $\mathcal{A}' \cong \mathcal{A}$. Therefore, φ^* is a Scott sentence for $\mathcal{A}$.

Now, let ψ be a Scott sentence for $\mathcal{A}$. Let ψ^* be the Ψ-pullback of ψ. Adding a conjunct, if necessary, we may suppose that ψ^* implies the computable Π_2 sentence characterizing the class K. Then we can show, by a sequence of equivalences just like the one in the previous paragraph, that ψ^* is a Scott sentence for G. ⊣

§2. Transfer of computable Scott sentences. In this section, we prove the first of our two main results. In [12], Gonzalez and Rossegger proved that for a structure $\mathcal{A} \in Mod(L)$, $\mathcal{A}$ and $T_{\mathcal{A}}$ have the same Scott rank. By Montalbán's Theorem, if one of the structures has a Scott sentence that is $\Pi_{\alpha+1}$, then so does the other. Gonzalez and Rossegger asked whether the same is true for computable infinitary Scott sentences. We give a positive answer.

THEOREM A. *Let* L *be a finite relational language. If one of the structures* $\mathcal{A}$ *or* $T_{\mathcal{A}}$ *has a computable* $\Pi_{\alpha+1}$ *Scott sentence, then so does the other. Moreover, we can pass effectively from a Scott sentence for one to a Scott sentence for the other.*

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

REMARK. We can vary Theorem A, letting the language L be computable, but not necessarily finite.

We split the proof of Theorem A in two. The result below gives the easier implication.

PROPOSITION 2.1. *Given a computable infinitary Scott sentence φ for $T_{\mathcal{A}}$, we can effectively pass to a computable infinitary Scott sentence φ^* for $\mathcal{A}$, where the complexity of φ^* matches that of φ.*

PROOF. The embedding $\mathcal{A} \to T_{\mathcal{A}}$ is Turing computable. Applying the Pullback Theorem, we take φ^* to be the pullback of φ. The two sentences have the same complexity, and for $\mathcal{B} \in Mod(L)$, $\mathcal{A} \cong \mathcal{B}$ iff $T_{\mathcal{A}} \cong T_{\mathcal{B}}$. Since φ is a Scott sentence for $T_{\mathcal{A}}$, we have that

$$\mathcal{B} \models \varphi^* \iff T_{\mathcal{B}} \models \varphi \iff T_{\mathcal{B}} \cong T_{\mathcal{A}} \iff \mathcal{A} \cong \mathcal{B},$$

and so φ^* is a Scott sentence for $\mathcal{A}$. ⊣

Every $L_{\omega_1\omega}$-sentence is X-computable for some X. Relativizing Proposition 2.1, we get the result, already proved by Gonzalez and Rossegger [12], saying that if $T_{\mathcal{A}}$ has a Scott sentence that is Π_α, then so does $\mathcal{A}$.

The result below gives the harder implication in Theorem A.

PROPOSITION 2.2. *For a computable ordinal α, if $\mathcal{A}$ has a computable Π_α Scott sentence φ, then so does $T_{\mathcal{A}}$.*

2.1. Outline for the proof of Proposition 2.2. We begin by isolating some simple properties of trees $T_{\mathcal{A}}$ for $\mathcal{A} \in Mod(L)$. We write $\preceq$ for the tree ordering; i.e., $\sigma \preceq \tau$ if $\sigma = p^k(\tau)$ for some k. For a labeled tree T satisfying the first two properties below, any path p yields a structure $\mathcal{A}^p$. Next, we define forcing and prove the basic lemmas as in Cohen [7]. Suppose $T \cong T_{\mathcal{A}}$, where $\mathcal{A}$ has computable Scott rank α. We show that if p is a generic path, then $\mathcal{A}^p \cong \mathcal{A}$. If $\mathcal{A}$ has a computable infinitary Scott sentence φ, then the lemma on definability of forcing lets us pass from φ to a computable infinitary Scott sentence for $T_{\mathcal{A}}$.

2.2. Simple properties of trees $T_{\mathcal{A}}$. The basic properties listed below are simple to state, and they are clearly true of the tree structures $T_{\mathcal{A}}$. We will give further properties after we have defined forcing.

Levels: Each node satisfies exactly one U_D. Moreover, if σ is at level n, then σ satisfies U_D for some D assigning a complete atomic type to the variables $(x_1, \ldots, x_n)$.

Consistency: If $\sigma \preceq \tau$, then the atomic type assigned to τ implies the one assigned to σ.

Replication: If σ has at least one successor satisfying U_D, then it has infinitely many.

2.3. The definition of forcing. In this section, we define the appropriate kind of forcing to produce a generic path. Let T be a tree satisfying the levels and consistency properties. Each path p yields an L-structure $\mathcal{A}^p$ with universe $X = \{x_1, x_2, x_3, \ldots\}$. We think of the variables in X as constants. The atomic diagram of $\mathcal{A}^p$ is the set of conjuncts $\pm\alpha(\bar{x})$ of formulas $D = D(\bar{x})$ such that some $\sigma \prec p$ has label U_D. The

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

forcing conditions are the nodes of T. The partial ordering is the tree ordering $\preceq$. The forcing language describes the structure $\mathcal{A}^p$ obtained from a generic path p. The language consists of computable infinitary sentences $\psi(\bar{x})$ in the language $L \cup X$. We define forcing for these sentences by induction on complexity.

DEFINITION 7.

1. Suppose $\alpha(\bar{x})$ is finitary quantifier-free. For σ with label U_D, $\sigma \Vdash \alpha$ if the variables $\bar{x}$ are among those of σ and D logically implies $\alpha(\bar{x})$.
2. Suppose $\theta(\bar{x})$ is computable Σ_β, of form $\bigvee_i (\exists \bar{u}_i)\psi_i(\bar{x}, \bar{u}_i)$, where ψ_i is computable Π_{γ_i} for some $\gamma_i < \beta$, Then $\sigma \Vdash \theta(\bar{x})$ if for some i and some assignment t mapping $\bar{u}_i$ to variables of σ, we have $\sigma \Vdash \psi_i(\bar{x}, t(\bar{u}_i))$.
3. Suppose $\theta(\bar{x})$ is computable Π_β, of form $\bigwedge_i (\forall \bar{u}_i)\psi_i(\bar{x}, \bar{u}_i)$. Then $\sigma \Vdash \theta(\bar{x})$ if the variables $\bar{x}$ are among those of σ and for all $\sigma' \succeq \sigma$, $\sigma' \nVdash neg(\theta(\bar{x}))$—recall that $neg(\theta)$ is the (computable Σ_β) formula that is equivalent to $\neg\theta$, but with the negations brought inside.

2.4. The forcing lemmas. We state the usual forcing lemmas. We show that they hold for all T satisfying the levels and consistency properties.

LEMMA 2.3 (Extension). *If $\sigma \Vdash \psi(\bar{x})$ and $\sigma \preceq \sigma'$, then $\sigma' \Vdash \psi(\bar{x})$.*

PROOF. First, suppose $\psi(\bar{x})$ is finitary quantifier-free. If $\sigma \Vdash \psi(\bar{x})$, then the variables of σ include $\bar{x}$, and the label on σ assigns an atomic type D that logically implies $\psi(\bar{x})$. If $\sigma \preceq \sigma'$, then by Consistency, the label on σ', assigning an atomic type D' to a larger tuple of variables, implies D, so it implies $\psi(\bar{x})$. Next, suppose $\psi(\bar{x}) = \bigvee_i (\exists \bar{u}_i)\psi_i(\bar{x}, \bar{u}_i)$, where the statement holds for all $\psi_i(\bar{x}, \bar{z})$. If $\sigma \Vdash \psi(\bar{x})$, then for some i and some assignment fixing $\bar{x}$ and taking $\bar{u}_i$ to some $\bar{z}$, $\sigma \Vdash \psi_i(\bar{x}, \bar{z})$. If $\sigma \preceq \sigma'$, then by the induction hypothesis, $\sigma' \Vdash \psi_i(\bar{x}, \bar{z})$, so $\sigma \Vdash \psi(\bar{x})$. Finally, suppose $\psi(\bar{x}) = \bigwedge_i (\forall \bar{u}_i)\psi_i(\bar{x}, \bar{u}_i)$, where the statement holds for all $\psi_i(\bar{x}, \bar{z})$. If $\sigma \Vdash \psi(\bar{x})$, then by the definition of forcing, no extension of σ forces $neg(\psi(\bar{x}))$. For $\sigma \preceq \sigma'$, no extension of σ' forces $neg(\psi(\bar{x}))$, so $\sigma' \Vdash \psi(\bar{x})$. ⊣

LEMMA 2.4 (Consistency). *For all σ and $\psi(\bar{x})$, σ cannot force both $\psi(\bar{x})$ and $neg(\psi(\bar{x}))$.*

PROOF. First, suppose $\psi(\bar{x})$ is finitary quantifier-free. Then $neg(\psi(\bar{x})) = \neg\psi(\bar{x})$. If σ assigns the atomic type D to variables including $\bar{x}$, D does not imply both $\psi(\bar{x})$ and $\neg\psi(\bar{x})$, so σ does not force both. Next let $\psi(\bar{x}) = \bigvee_i (\exists \bar{u}_i)\psi_i(\bar{x}, \bar{u}_i)$. If $\sigma \Vdash neg(\psi(\bar{x}))$, then by definition, σ does not force $\psi(\bar{x})$. Similarly, for $\psi(\bar{x}) = \bigwedge_i (\forall \bar{u}_i)\psi_i(\bar{x}, \bar{u}_i)$, if $\sigma \Vdash \psi(\bar{x})$, then by definition, σ does not force $neg(\psi(\bar{x}))$. ⊣

LEMMA 2.5 (Density). *For all σ and $\psi(\bar{x})$, some $\sigma' \succeq \sigma$ such that σ' forces either $\psi(\bar{x})$ or $neg(\psi(\bar{x}))$.*

PROOF. First, suppose $\psi(\bar{x})$ is finitary quantifier-free. Take $\sigma \preceq \sigma'$ such that $\bar{x}$ is included among the variables of σ'. Then σ' forces one of $\psi(\bar{x})$, $\neg(\psi(\bar{x}))$, as the label of σ' describes the full quantifier-free type of $\bar{x}$. Next, suppose $\psi(\bar{x}) = \bigvee_i (\exists \bar{u}_i)\psi_i(\bar{x}, \bar{u}_i)$, where the statement holds for all $\psi_i(\bar{x}, \bar{z})$. If there is no $\sigma' \succeq \sigma$ such that σ' forces $\psi(\bar{x})$, then by definition, σ forces $neg(\psi(\bar{x}))$. Similarly, if $\psi(\bar{x}) = \bigwedge_i (\forall \bar{u}_i)\psi_i(\bar{x}, \bar{u}_i)$, and σ does not force $\psi(\bar{x})$, then by definition, there is some $\sigma' \succeq \sigma$ such that σ' forces $neg(\psi(\bar{x}))$. ⊣

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

Using the density and extension lemmas, we get the following.

LEMMA 2.6 (Existence of c.f.s.). *Any σ extends to a path with nodes deciding all computable infinitary sentences of complexity at most that of φ.*

PROOF. Let $(\psi_i)_{i\in\omega}$ be a list of the sentences in the forcing language. Starting with $\sigma_0 = \sigma$, we build a *complete forcing sequence*, or *c.f.s.*, where this is a sequence of nodes $\sigma_0 \preceq \sigma_1 \preceq \cdots$ such that σ_{i+1} decides ψ_i; i.e., σ_{i+1} forces either ψ_i or $neg(\psi_i)$. We choose σ_i at level at least i, so the sequence determines a path; i.e., p is the unique path containing all the nodes σ_i. ⊣

DEFINITION 8. A path p through T is *generic* if there is a c.f.s. (σ_i) that determines p.

LEMMA 2.7 (Truth-and-forcing). *Let $(\sigma_i)_{i\in\omega}$ be a c.f.s. giving the generic path* p, *and let $\mathcal{A}^p$ be the resulting structure. Then for all $\psi(\bar{x})$ in the forcing language, $\mathcal{A}^p \models \psi(\bar{x})$ iff there is some* i *such that $\sigma_i \Vdash \psi(\bar{x})$.*

PROOF. First, suppose $\psi(\bar{x})$ is finitary quantifier-free. Suppose the variables $\bar{x}$ are among the first i. Then $\bar{x}$ is among the variables of σ_i. If $\mathcal{A}^p \models \psi(\bar{x})$, then $\sigma_i \Vdash \psi(\bar{x})$. If $\mathcal{A}^p \models \neg\psi(\bar{x})$, then $\sigma_i \Vdash \neg\psi(\bar{x})$, where $\neg\psi(\bar{x}) = neg(\psi(\bar{x}))$. No $\sigma_{i'}$ can force $\psi(\bar{x})$. Next, let $\psi(\bar{x}) = \bigvee_j (\exists \bar{u}_j)\psi_j(\bar{x}, \bar{u}_j)$, where the statement holds for each $\psi_j(\bar{x}, \bar{z})$. If $\mathcal{A}^p \models \psi(\bar{x})$, then for some j and $\bar{z}$, $\mathcal{A}^p \models \psi_j(\bar{x}, \bar{z})$. Some σ_i forces $\psi_j(\bar{x}, \bar{z})$, so it forces $\psi(\bar{x})$. If $\mathcal{A}^p \models neg(\psi(\bar{x}))$, then for all j and all $\bar{z}$, $\mathcal{A}^p \models neg(\psi_j(\bar{x}, \bar{z}))$. Then no σ_i forces $\psi(\bar{x})$. Therefore, by the definition of c.f.s., some σ_i forces $neg(\psi(\bar{x}))$. We cannot have $\psi(\bar{x})$ and $neg(\bar{x})$ each forced by some term in the c.f.s., for then a single σ_i would force both. Finally, let $\psi(\bar{x}) = \bigwedge_j (\forall \bar{u}_j)\psi_j(\bar{x}, \bar{u}_j)$. We have already seen that whichever of $\psi(\bar{x})$, $neg(\psi(\bar{x}))$ is true in $\mathcal{A}^p$ is forced by some σ_i. ⊣

REMARK. Suppose $(\sigma_i)_{i\in\omega}$ is a chain of forcing conditions that decides all computable Σ_β sentences for $\beta < \alpha$. Let p be the resulting path, and let $\mathcal{A}^p$ be the corresponding structure. Then truth-and-forcing holds for computable Σ_β and computable Π_β sentences $\psi(\bar{x})$.

LEMMA 2.8 (Definability of forcing). *For each sentence $\theta(\bar{x})$ in the forcing language, we can find a formula $Force_{\theta(\bar{x})}(y)$ that defines in* T *the set of $\sigma \in T$ such that $\sigma \Vdash \theta$. The formula has the same complexity as $\theta(\bar{x})$, except that if θ is finitary quantifier-free, we may take $Force_\theta(y)$ to be either computable Σ_1 or computable Π_1.*

PROOF. First, suppose $\alpha(\bar{x})$ is finitary quantifier-free. We have $\sigma \Vdash \alpha(\bar{x})$ if σ satisfies the disjunction of $U_D(y)$ for D such that the variables include those of α and D implies α. This is computable Σ_1. We may also say that no extension of σ forces $\neg\alpha$. This is computable Π_1. Next, consider a computable Σ_β sentence $\theta(\bar{x}) = \bigvee_i (\exists \bar{u}_i)\psi_i(\bar{x}, \bar{u}_i)$. We have $\sigma \Vdash \theta(\bar{x})$ if the variables of σ include $\bar{x}$ and $t(\bar{u}_i)$ for some i and some assignment t taking $\bar{u}_i$ to X, and $\sigma \Vdash \psi_i(\bar{x}, t(\bar{u}_i))$. Using the induction hypothesis, we get a computable Σ_β formula saying that $\sigma \Vdash \theta(\bar{x})$. Finally, consider a computable Π_β sentence $\theta(\bar{x}) = \bigwedge_i (\forall \bar{u}_i)\psi_i(\bar{x}, \bar{u}_i)$. We have $\sigma \Vdash \theta(\bar{x})$ if no extension of σ forces the Σ_β sentence $neg(\theta(\bar{x}))$. This is computable Π_β, again by induction. ⊣

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

LEMMA 2.9. *Let $\mathcal{A}$ be an* L*-structure with a computable* Π_α *Scott sentence* φ*. Let* p *be a generic path through* $T_{\mathcal{A}}$*. Then the structure* $\mathcal{A}^p$ *given by* p *is isomorphic to* $\mathcal{A}$*.*

PROOF. Take $\sigma \prec p$ deciding φ. We can extend σ to a c.f.s. $(\sigma_i)_{i\in\omega}$ such that the tuple $\bar{a}_i$ represented by σ_i includes the first i elements of $\mathcal{A}$. Let q be the path with $\sigma_i \prec q$ for all i. Then $\mathcal{A}^q \cong \mathcal{A}$, so $\mathcal{A}^q \models \varphi$, By truth-and-forcing, $\sigma \Vdash \varphi$, so $\mathcal{A}^p \models \varphi$. ⊣

DEFINITION 9 (Weak forcing). For $\sigma \in T$ and $\psi(\bar{x})$ a sentence in our forcing language, σ *weakly forces* $\psi(\bar{x})$ if no extension of σ forces $neg(\psi)$.

Remark: The lemma above shows that if φ is a computable Π_α Scott sentence for $\mathcal{A}$, then the base node of $T_{\mathcal{A}}$ weakly forces φ. For computable Π_α formulas, weak forcing is the same as forcing.

LEMMA 2.10. *Suppose $\sigma \in T_{\mathcal{A}}$, where σ has variables $\bar{x}$ and σ represents the tuple $\bar{a}$. For any computable infinitary formula $\psi(\bar{x})$ satisfied by $\bar{a}$, σ weakly forces $\psi(\bar{x})$.*

PROOF. Suppose τ is an extension of σ that decides $\psi(\bar{x})$. There is a c.f.s. $(\sigma_i)_{i\in\omega}$, extending τ, and such that σ_i, with variables $\bar{x}_i$, represents a tuple $\bar{a}_i$ including the first i elements of $\mathcal{A}$. Let q be the generic path with $\sigma_i \prec q$ for all i. Then $\mathcal{A}^q \cong \mathcal{A}$, with an isomorphism F that maps $\bar{x}_i$ to $\bar{a}_i$. Since $\mathcal{A} \models \psi(\bar{a})$, $\mathcal{A}^q \models \psi(\bar{x})$, so $\tau \Vdash \psi(\bar{x})$. Then σ weakly forces $\psi(\bar{x})$. ⊣

So far, we have used only the simplest properties of structures in $\mathbb{T}$. Below, we use forcing to state some further properties.

α-agreement: Let $\psi(\bar{x})$ be a computable Σ_α formula, where $\bar{x}$ is among the first n variables. For σ at level n, if some extension τ of σ forces $\tau \Vdash \psi(\bar{x}, \bar{u})$, then σ weakly forces $\psi(\bar{x}, \bar{u})$.

If α is a computable limit ordinal, then we consider just the disjuncts of $\psi(\bar{x})$, each of which is computable Σ_β for some $\beta < \alpha$. We see that T satisfies α-agreement iff it satisfies β-agreement for all $\beta < \alpha$.

α-permutation: Let σ be a node with variables $\bar{x}$. Let $\psi(\bar{x}, z)$ be a computable Σ_α formula, where z is not in $\bar{x}$. If some $\tau \succeq \sigma$ forces $\psi(\bar{x}, z)$, and x' is the first variable not in $\bar{x}$, then some $\tau' \succeq \sigma$ forces $\psi(\bar{x}, x')$. (With α-agreement, this means that the node $v \preceq \tau'$ with variables $\bar{x}, x'$ weakly forces $\psi(\bar{x}, x')$.)

Again, if α is a computable limit ordinal, then we consider the disjuncts, each of which is computable Σ_β for some $\beta < \alpha$. We see that T satisfies α-permutation iff it satisfies β-permutation for all $\beta < \alpha$.

Below, we give computable infinitary axioms for all of the basic properties.

LEMMA 2.11. *For a computable ordinal $\alpha \geq 2$, there is a computable Π_α-sentence stating the properties levels, consistency, replication, plus β-agreement, and β-permutation for $\beta < \alpha$.*

PROOF. *Levels*: Recall that L is a finite relational language. For each n, there are only finitely many possible atomic types for the tuple $\bar{x} = (x_1, \ldots, x_n)$. We add two sentences to ensure the Levels property.

(1) First, taking into account all n, we say that each x satisfies U_D for exactly one D. This is computable Π_2. To say that $x = \lambda$ (the base node), we write $p(x) = x$. To say that x is at level $n \geq 1$, we say $p^{n+1}(x) = p^n(x)$ & $p^n(x) \neq p^{n-1}(x)$. Treating

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

p as a function symbol, this is finitary quantifier-free. Treating p as a relation symbol, it would be $\exists_1$ and $\forall_1$.

(2) We say that for each n, for all x at level n, x satisfies U_D for one of the finitely many D assigning an atomic type to $(x_1, \ldots, x_n)$. This is computable Π_2.

Altogether, it is computable Π_2 to say that T satisfies (1) and (2).

Consistency: Recall that T has the Consistency Property if for $\sigma \subseteq \tau$, the atomic type D' assigned by τ implies the atomic type D assigned by σ. For a given n, let P_n be the set of pairs (D, D') such that D is an atomic type in the first n variables, D' is an atomic type in the first $n+1$ variables, and D' logically implies D. We say for all x, y if y is at level $n+1$ and $p(y) = x$, then $\bigvee_{(D,D')\in P_n}(U_D x \ \& \ U_{D'} y)$. This is computable Π_2.

Replication: Recall that T has the Replication Property if for all σ if some successor satisfies U_D, then infinitely many do. This is computable Π_2.

β-agreement: Recall that T has the β-agreement Property if for all n, for all computable Σ_β formulas $\psi(\bar{x})$ in variables among the first n, and all σ at level n, if there is some τ such that $\sigma \preceq \tau$ and $\tau \Vdash \psi(\bar{x})$, then there is no τ' such that $\sigma \preceq \tau'$ and τ' forces $neg(\psi(\bar{x}))$. It is computable Σ_β to say of τ that it extends σ and forces $\psi(\bar{x})$. It is computable Π_β to say of τ such that $\sigma \preceq \tau$ that it forces $neg(\psi(\bar{x}))$. To say that τ does not force $neg(\psi(\bar{x}))$ is computable Σ_β. We have a computable $\Pi_{\beta+1}$ sentence saying that for all n, for all $\psi(\bar{x})$ with variables among the first n and σ at level n, if some extension of σ forces $\psi(\bar{x})$, then no extension forces $neg(\psi(\bar{x}))$.

If $\alpha = \beta + 1$, then it is enough to say that T has the β-agreement property, which is $\Pi_{\beta+1}$. If α is a computable limit ordinal, then we say that T has the β-agreement property for all $\beta < \alpha$. This is computable Π_α.

β-permutation: Recall that T has the β-permutation property if for ψ a computable Σ_β formula, and $\bar{x}$ consisting of the first n variables, if σ, at level n, has an extension forcing $\psi(\bar{x}, z)$, where z is a later variable, then some extension of σ forces $\psi(\bar{x}, x')$, where x' is the $(n+1)^{st}$ variable. For fixed n, ψ and σ at level n, it is computable Σ_β to say that some extension of σ forces $\psi(\bar{x}, z)$ for some later variable z, and it is computable Σ_β to say that some immediate extension forces $\psi(\bar{x}, x')$. Then it is computable $\Pi_{\beta+1}$ to say that for all n, ψ, and σ at level n, if some extension of σ forces $\psi(\bar{x}, z)$ for some z, then some immediate extension of σ forces $\psi(\bar{x}, x')$.

If $\alpha = \beta + 1$, it is enough to say that T has the β-permutation property, which is computable $\Pi_{\beta+1}$. If α is a computable limit ordinal, then we say that T has the β-permutation property for all $\beta < \alpha$. This is computable Π_α. ⊣

2.5. Completing the proof of Proposition 2.2. We suppose that φ is a computable Π_α Scott sentence for $\mathcal{A}$. Necessarily, $\alpha \geq 2$. By Lemma 2.11, we have a computable Π_α sentence θ characterizing the labeled trees T that satisfy the basic properties levels, consistency, replication, plus β-agreement, and β-permutation, for all $\beta < \alpha$. Let φ_* be the natural computable Π_α-sentence saying that the root node weakly forces φ. Then $\theta \ \& \ \varphi_*$ is a computable Π_α sentence. The result below says that this is a Scott sentence for $T_{\mathcal{A}}$. This is all we need to complete the proof of Proposition 2.2.

PROPOSITION 2.12. *Suppose that $\mathcal{A}$ has a computable Π_α Scott sentence. Let* T *be a labeled tree satisfying $\theta \ \& \ \varphi_*$. Then $T_{\mathcal{A}} \cong T$.*

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

PROOF. By Theorem 1.3, the fact that $\mathcal{A}$ has a computable Π_α Scott sentence implies that the orbit of each tuple $\bar{a}$ in $\mathcal{A}$ is defined by a computable Σ_β formula for some $\beta < \alpha$. For each $\bar{a}$, we choose (non-effectively) such a formula $d_{\bar{a}}(\bar{x})$. Let $D_{\bar{a}}(\bar{x})$ be the formula giving the atomic type of $\bar{a}$. To show that $T \cong T_{\mathcal{A}}$, we define a back-and-forth family. We use a preliminary notion.

DEFINITION 10. Let σ be a node at level n in $T_{\mathcal{A}}$, representing a tuple $\bar{a}$ in $\mathcal{A}$. We say that τ, at level n of T, is a *good match for* σ if τ satisfies $U_{D_{\bar{a}}}$ and weakly forces $d_{\bar{a}}(\bar{x})$.

DEFINITION 11. Let F be the set of partial isomorphisms f that take a finite labeled subtree S of $T_{\mathcal{A}}$ to a finite labeled subtree S' of T such that for each node $\sigma \in S$, $f(\sigma)$, in S' is a good match for σ.

We prove that F has the back-and-forth property.

F $\neq \emptyset$. The base node λ in $T_{\mathcal{A}}$ represents $\emptyset$ in $\mathcal{A}$. We may take $d_\emptyset$ to be φ, the Scott sentence for $\mathcal{A}$. The base node λ in T is the unique good match, as by assumption $T \models \varphi_*$ so λ weakly forces φ. The function f taking the base node λ of $T_{\mathcal{A}}$ to the base node λ of T is in F.

Forth. Take $f \in F$, with domain S and range S'. Let $\sigma \in T_{\mathcal{A}}$ be a terminal node in S, with variables $\bar{x}$, and let σ' be a successor of σ, with variables $\bar{x}, x'$. Say that σ represents $\bar{a}$, and σ' represents $\bar{a}, a'$. Suppose that $f(\sigma) = \tau$, where τ is a good match for σ. We need a successor τ' of τ that is a good match for σ'. Since τ weakly forces φ & $d_{\bar{a}}(\bar{x})$, some $\nu \succeq \tau$ forces this. Any generic path p extending ν gives a structure $\mathcal{A}^p$ satisfying φ & $d_{\bar{a}}(\bar{x})$, so $(\mathcal{A}^p, \bar{x}) \cong (\mathcal{A}, \bar{a})$. The isomorphism takes some y to a'. Take ν' such that $\nu \prec \nu' \prec p$ and y is among the variables of ν'. Note that ν' weakly forces $d_{\bar{a},a'}(\bar{x}, y)$. By the Permutation property, there is a successor τ' of τ such that τ' weakly forces $d_{\bar{a},a'}(\bar{x}, x')$, where x' is the next variable after $\bar{x}$. We extend f, taking σ' to τ'.

Back. Take f, σ, τ $\bar{a}$, and $\bar{x}$ be as in the previous paragraph. Suppose τ' is a successor of τ. There is a generic path p extending τ'. We have an isomorphism g from $\mathcal{A}$ to $\mathcal{A}^p$ that takes $\bar{a}$ to $\bar{x}$. Say that x' is the next variable after $\bar{x}$. Then τ' has variables $\bar{x}, x'$. Let a' be the g-pre-image of x'. For the formula $d_{\bar{a},a'}$ that defines the orbit of $\bar{a}, a'$ in $\mathcal{A}$, some node of p must force $d_{\bar{a},a'}(\bar{x}, x')$, and τ' must weakly force it. Let σ' be the successor of σ that represents $\bar{a}, a'$. Then τ' is a good match for σ'. We extend f, taking σ' to τ'. ⊣

REMARK. If $\mathcal{A}$ has a Scott sentence φ that is computable Σ_α, or computable d- Σ_α, the sentence $\theta \& \varphi_*$ is still $\Pi_{\alpha+1}$ because θ is $\Pi_{\alpha+1}$. This is consistent with the results of Gonzalez and Rossegger [12]. They show, in the boldface setting, that if $\mathcal{A}$ has Scott complexity Σ_α and d- Σ_α, then $T_{\mathcal{A}}$ has Scott complexity $\Pi_{\alpha+1}$. This means that the above result is the best possible. For example, consider $\omega^\alpha \cdot 2$, where α is a computable ordinal. This structure has a computable d- $\Sigma_{\alpha+1}$ Scott sentence and no $\Pi_{\alpha+1}$ Scott sentence (see [23, Chapter 2]). Because it has no $\Pi_{\alpha+1}$ Scott sentence, $T_{\omega^\alpha \cdot 2}$ has Scott complexity $\Pi_{\alpha+2}$. Therefore, the argument above produces a computable Scott sentence of the ideal complexity, $\Pi_{\alpha+2}$. Note that this precludes the possibility of a simpler set of tree properties θ. If θ were d- Σ_α, then θ & φ_* would be a d- Σ_α Scott sentence for $T_{\omega^\alpha \cdot 2}$, which is a contradiction.

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

§3. Describing $\mathbb{T}^\alpha$. In this section, as in the previous one, L is a fixed finite relational language, and for a countable ordinal α, $\mathbb{T}^\alpha$ is the class of labeled trees T such that for some $\mathcal{A}$ in $Mod(L)$ having Scott rank at most α, $T \cong T_{\mathcal{A}}$. We will show that if α is a computable ordinal, then $\mathbb{T}^\alpha$ is effective $\Pi_{2\alpha+2}$. This relativizes, and we get the fact that for all countable ordinals α, $\mathbb{T}^\alpha$ is $\mathbf{\Pi}_{2\alpha+2}$ in the Borel hierarchy. We will see that a labeled tree T is in $\mathbb{T}^\alpha$ iff it satisfies certain properties, all of which are given by computable (or X-computable) $\Pi_{2\alpha+2}$-sentences. More specifically, we will show that a labeled tree T is in $\mathbb{T}^\alpha$ iff it satisfies the following properties:

1. levels;
2. consistency;
3. replication;
4. α-agreement;
5. α-permutation;
6. α-support.

In the previous section, we stated properties characterizing the labeled trees T isomorphic to $T_{\mathcal{A}}$, for $\mathcal{A}$ having a specific computable infinitary Scott sentence. In that section, we used forcing. The first three properties—levels, consistency, and replication—were stated in a way that did not involve forcing. They were given by computable Π_2-sentences. The other properties were defined in terms of forcing. In this section, we shall give new definitions, which do not involve forcing. We begin with some back-and-forth relations.

Definition 12. Let T be a labeled tree satisfying levels, consistency, and replication. Suppose σ and τ are nodes in T, and let $\bar{x}$ and $\bar{y}$ be tuples of variables, of the same length, where $\bar{x}$ is among the variables of σ and $\bar{y}$ is among the variables of τ.

1. $(\sigma, \bar{x}) \leq_0 (\tau, \bar{y})$ if the atomic type given to $\bar{x}$ by the label on σ matches that given to $\bar{y}$ by the label on τ;
2. for $\alpha > 0$, $(\sigma, \bar{x}) \leq_\alpha (\tau, \bar{y})$ if for all $\beta < \alpha$, all τ' extending τ, with variables including $\bar{y}$ and $\bar{y}'$, there exist σ' extending σ and $\bar{x}'$ such that $(\tau', \bar{y}, \bar{y}) \leq_\beta (\sigma', \bar{x}, \bar{x}')$.

In the previous section, we described the trees T isomorphic to $T_{\mathcal{A}}$ for a specific $\mathcal{A}$, assuming that $\mathcal{A}$ has a computable Π_α Scott sentence. In this case, the orbit of each tuple is defined by a computable Σ_β formula for some $\beta < \alpha$. We stated the properties of α-agreement and α-permutation in terms of forcing, which depended on the computability of these orbit-defining formulas. Here we focus on structures of Scott rank at most α. We do not require that the orbits be defined by computable infinitary formulas. For this reason, we define α-agreement and α-permutation in a way that does not involve forcing. We also add a final property, α-support.

α-agreement: A labeled tree T satisfies *α-agreement* provided that for all nodes ν, σ, and τ, and all $\beta < \alpha$, if $\nu \prec \tau$ and $\nu \prec \sigma$, where ν has variables $\bar{x}$ and τ has variables $\bar{x}, \bar{y}$, then there exists τ', with variables including $\bar{x}$ and some $\bar{z}$ such that $\sigma \prec \tau'$ and $(\tau, \bar{x}, \bar{y}) \leq_\beta (\tau', \bar{x}, \bar{z})$.

α-permutation: A labeled tree T satisfies *α-permutation* provided that for all nodes σ, with variables $\bar{x}$, all $\beta < \alpha$, if for some τ, with variables $\bar{x}, \bar{y}, z$, we have $\sigma \prec \tau$, then for x' the next variable after $\bar{x}$, there exists τ', with variables $\bar{x}$, x' and some $\bar{u}$, such that $(\tau, \bar{x}, \bar{y}, z) \leq_\beta (\tau', \bar{x}, \bar{u}, x')$.

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

α-*support*: A labeled tree T satisfies α-*support* provided that for all σ with variables $\bar{x}$, there exist $\beta < \alpha$ and $\tau \succ \sigma$, with variables $\bar{x}$ and some $\bar{y}$, such that for all τ', with variables including $\bar{x}'$ of the same length as $\bar{x}$ and $\bar{y}'$ of the same length as $\bar{y}$, $(\tau, \bar{x}, \bar{y}) \leq_\beta (\tau', \bar{x}', \bar{y}')$ implies $(\sigma, \bar{x}) \leq_\alpha (\tau', \bar{x}')$.

It is easy to see that all $T \in \mathbb{T}^\alpha$ satisfy α-agreement, α-permutation, and α-support. It is also easy to see the following.

LEMMA 3.1. *For each computable ordinal α, there is a computable $\Pi_{2\alpha+2}$ sentence characterizing the labeled trees* T *that satisfy α-agreement and α-permutation* (*without forcing*), *and α-support*.

We want to prove the following.

THEOREM 3.2. *If* T *is a labeled tree satisfying length, consistency, replication, the new versions of α-agreement and α-permutation, and α-support, then* T *is in $\mathbb{T}^\alpha$*.

Here is an outline of the proof.

1. We say what it means for a path to be "α-generic" in a way that does not involve forcing.
2. We show that T has an α-generic path p.
3. We show that the resulting structure $\mathcal{B} = \mathcal{A}^p$ has rank at most α.
4. Finally, we show that $T \cong T_{\mathcal{B}}$.

Having fixed a countable ordinal α, we say that a labeled tree T has *all the properties*, meaning that it has all six properties listed in the statement of Theorem 3.2.

DEFINITION 13. Let T be a labeled tree satisfying all the properties. A path p is α-*generic* if for all $\beta < \alpha$ and all pairs of nodes (v, τ), where v has variables $\bar{x}$, τ has variables $\bar{x}, \bar{y}$, $v \prec p$, and $v \prec \tau$, there exists τ', with variables including $\bar{x}$ and some $\bar{z}$, such that $v \prec \tau' \prec p$, and $(\tau, \bar{x}, \bar{y}) \leq_\beta (\tau', \bar{x}, \bar{z})$.

LEMMA 3.3. *Let* T *be a labeled tree satisfying all the properties. For each $\sigma \in T$, there is an α-generic path with $\sigma \prec p$*.

PROOF. We build p satisfying the following witnessing requirements.

$W_{(v,\tau,\beta)}$: If $v \prec p$ and $v \prec \tau$, where v has variables $\bar{x}$ and τ has variables $\bar{x}, \bar{y}$, then there is some $\tau' \prec p$ with variables including $\bar{x}$ and some $\bar{z}$, such that $(\tau, \bar{x}, \bar{y}) \leq_\beta (\tau', \bar{x}, \bar{z})$.

Fix a list of the requirements. To build p, we form an increasing chain of nodes $(\sigma_s)_{s\in\omega}$. Let $\sigma_0 = \sigma$. Given σ_s, we take the first requirement $W_{(v,\tau,\beta)}$ needing attention, where this means that $v \prec \sigma_s$, and there does not exist $\tau' \prec \sigma_s$ as above. The α-agreement property gives an extension σ_{s+1} of σ_s satisfying the requirement. The resulting path p is α-generic. ⊣

LEMMA 3.4. *Let* T *be a tree satisfying all the properties, and let* P *be the set of pairs $(\sigma, \bar{x})$ such that $\sigma \in T$ and $\bar{x}$ is among the variables of σ. For all $\beta \leq \alpha$, for all $(\sigma, \bar{x})$ and $(\tau, \bar{y})$ in* P, *if $(\sigma, \bar{x}) \leq_\beta (\tau, \bar{y})$, then for all α-generic paths* p *extending σ and* q *extending τ, $(\mathcal{A}^p, \bar{x}) \leq_\beta (\mathcal{A}^q, \bar{y})$*.

PROOF. We proceed by induction on β. For $\beta = 0$, the label on σ determines the atomic type of $\bar{x}$ in $\mathcal{A}^p$, and the label on τ determines the atomic type of $\mathcal{A}^q$, so the

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

statement holds. Supposing that the statement holds for $\delta < \beta$, we show that it holds for β. Take $(\sigma, \bar{x})$ and $(\tau, \bar{y})$ in P, where $(\sigma, \bar{x}) \leq_\beta (\tau, \bar{y})$. Let p and q be α-generic paths, where $\sigma \prec p$ and $\tau \prec q$. We must show that $(\mathcal{A}^p, \bar{x}) \leq_\beta (\mathcal{A}^q, \bar{y})$. Take $\delta < \beta$ and $\bar{v}$, and let $\tau' \prec q$, with variables including $\bar{y}, \bar{v}$. We need $\sigma' \prec p$ with variables including $\bar{x}$ and some $\bar{u}$, such that $(\tau', \bar{y}, \bar{v}) \leq_\delta (\sigma', \bar{x}, \bar{u})$. By the definition of the back-and-forth relations, there exists $\sigma*$, with variables including $\bar{x}$ and some $\bar{u}*$, such that $\sigma \prec \sigma*$ and $(\tau, \bar{y}, \bar{v}) \leq_\delta (\sigma*, \bar{x}, \bar{u}*)$. By α-genericity, there exists $\sigma' \prec p$, with variables including $\bar{x}$ and some $\bar{u}$, such that $(\sigma*, \bar{x}, \bar{u}*) \leq_\delta (\sigma', \bar{x}, \bar{u}')$. Then $(\tau', \bar{y}, \bar{v}) \leq_\delta (\sigma', \bar{x}, \bar{u}')$. By the induction hypothesis, $(\mathcal{A}^q, \bar{y}, \bar{v}) \leq_\delta (\mathcal{A}^p, \bar{x}, \bar{u}')$. Then $(\mathcal{A}^p, \bar{x}) \leq_\beta (\mathcal{A}^q, \bar{y})$, as required. ⊣

LEMMA 3.5. *Suppose* T *satisfies all the properties, and let* p *be an* α*-generic path. Then the resulting structure* $\mathcal{A}^p$ *has Scott rank at most* α.

PROOF. Recall that $\mathcal{A}^p$ has universe equal to X. By Lemma 1.6, to show that $\mathcal{A}^p$ has Scott rank at most α, it is enough to show that for each $\bar{x}$ in X, there exist $\bar{u}$ and $\beta < \alpha$ such that for all $\bar{x}'$ of the same length as $\bar{x}$ and $\bar{u}'$ of the same length as $\bar{u}$, $(\mathcal{A}^p, \bar{x}, \bar{u}) \leq_\beta (\mathcal{A}^p, \bar{x}', \bar{u}')$ implies $(\mathcal{A}^p, \bar{x}) \leq_\alpha (\mathcal{A}^p, \bar{x}')$. Take $\sigma \prec p$ with variables including $\bar{x}$. By α-support, there exist $\beta < \alpha$, and τ, with variables including $\bar{x}$ and some $\bar{u}$, such that for all τ' with variables including $\bar{x}'$ the same length as $\bar{x}$ and $\bar{u}'$ the same length as $\bar{u}$, $(\tau, \bar{x}, \bar{u}) \leq_\beta (\tau', \bar{x}', \bar{u}')$ implies $(\tau, \bar{x}) \leq_\alpha (\tau', \bar{x}')$. By α-genericity, there exists $\tau*$, with variables $\bar{x}$ and some $\bar{u}^*$ such that $\nu \prec \tau* \prec p$ and $(\tau, \bar{x}, \bar{u}) \leq_\beta (\tau*, \bar{x}, \bar{u}^*)$. Then $(\tau*, \bar{x}, \bar{u}^*) \leq_\beta (\tau', \bar{x}', \bar{u}')$ implies $(\tau*, \bar{x}) \leq_\alpha (\tau', \bar{x}')$. By Lemma 3.4, $(\mathcal{A}^p, \bar{x}) \leq_\alpha (\mathcal{A}^p, \bar{x}')$. ⊣

LEMMA 3.6. *Suppose* T *satisfies all the properties, and let* $(\sigma, \bar{x}) \leq_\alpha (\tau, \bar{y})$, *Then* $(\sigma, \bar{x}) \leq_{\alpha+1} (\tau, \bar{y})$.

PROOF. Assume that $(\sigma, \bar{x}) \leq_\alpha (\tau, \bar{y})$. Take an extension $(\tau', \bar{y}')$ of $(\tau, \bar{y})$. We aim to show that $(\sigma, \bar{x})$ has an extension $(\sigma', \bar{x}')$ such that $(\sigma', \bar{x}') \geq_\alpha (\tau', \bar{y}')$. By the definition of α-support, there exist an extension $(\tau'', \bar{y}', \bar{u})$ of $(\tau', \bar{y}')$ and $\beta < \alpha$ such that for all $(\nu, \bar{z}, \bar{w})$, $(\tau'', \bar{y}', \bar{u}) \leq_\beta (\nu, \bar{z}, \bar{w})$ implies $(\tau', \bar{y}') \leq_\alpha (\nu, \bar{z})$. Since $(\sigma, \bar{x}) \leq_\alpha (\tau, \bar{y})$, $(\sigma, \bar{x})$ has an extension $(\sigma'', \bar{x}', \bar{u}')$ with $(\sigma'', \bar{x}', \bar{u}') \geq_\beta (\tau'', \bar{y}', \bar{u})$. By α-support, $(\sigma'', \bar{x}') \geq_\alpha (\tau', \bar{y}')$. Thus, $(\sigma, \bar{x}) \leq_{\alpha+1} (\tau, \bar{y})$ as desired. ⊣

LEMMA 3.7. *If* T *satisfies all the properties, and* p *and* q *are* α*-generic paths through* T. *Then* $\mathcal{A}^p \cong \mathcal{A}^q$.

PROOF. Let $\mathcal{F}$ be the set of finite partial isomorphisms that take $\bar{x} \in p$ to $\bar{y} \in q$, where for some $\sigma \prec p$, with variables $\bar{x}$ and $\tau \prec q$, with variables $\bar{y}$ we have $(\sigma, \bar{x}) \leq_\alpha (\tau, \bar{y})$. We show that $\mathcal{F}$ has the back-and-forth property.

Clearly, $\mathcal{F}$ is non-empty.

Forth: Suppose $\bar{x} \to \bar{y}$ is in $\mathcal{F}$. Say $\sigma \prec p$ have variables $\bar{x}$ and $\tau \prec q$ have variables $\bar{y}$ with $(\sigma, \bar{x}) \leq_\alpha (\tau, \bar{y})$. By Lemma 3.6, $(\sigma, \bar{x}) \leq_{\alpha+1} (\tau, \bar{y})$. Another application of Lemma 3.6 gives that $(\sigma, \bar{x}) \geq_{\alpha+1} (\tau, \bar{y})$. Let z be a variable not in $\bar{x}$ and σ' be an extension of σ with $\sigma' \prec p$ including z. By the definition of the back-and-forth relations, there is an extension $(\tau', \bar{y}, w)$ of $(\tau, \bar{y})$ (not necessarily along q), such that $(\tau', \bar{y}, w) \geq_\alpha (\sigma', \bar{x}, z)$. In the definition of α-support, there is an extension $(\tau'', \bar{y}, w, \bar{u})$ and ordinal $\beta < \alpha$ associated with $(\tau', \bar{y}, w)$ such that for all $(\nu, \bar{s}, t, \bar{r})$,

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

$(\tau'', \bar{y}, w, \bar{u}) \leq_\beta (v, \bar{s}, t, \bar{r})$ implies that $(\tau', \bar{y}, w) \leq_\alpha (v, \bar{s}, t)$. By α-genericity of q, there is an extension $(\mu, \bar{y}, w^*, \bar{u}*)$ of $(\tau, \bar{y})$ along q such that $(\mu, \bar{y}, w^*, \bar{u}*) \geq_\beta (\tau'', \bar{y}, w, \bar{u})$. This means that $(\mu, \bar{y}, w^*) \leq_\alpha (\tau', \bar{y}, w)$. By Lemma 3.6, $(\mu, \bar{y}, w^*) \geq_\alpha (\tau', \bar{y}, w)$. This yields that $(\mu, \bar{y}, w^*) \geq_\alpha (\sigma', \bar{x}, z)$, and so $\bar{x}, z \to \bar{y}, w^*$ is in $\mathcal{F}$.

Back: Suppose $\bar{x} \to \bar{a}$ is in $\mathcal{F}$. Say $\sigma \prec p$ have variables $\bar{x}$ and $\tau \prec q$ have variables $\bar{y}$ with $(\sigma, \bar{x}) \leq_\alpha (\tau, \bar{y})$. By Lemma 3.6, $(\sigma, \bar{x}) \geq_\alpha (\tau, \bar{y})$. The argument in the forth section yields for any $(\tau', \bar{y}, z)$ extending $(\tau, \bar{y})$ along q, there is a $(\mu, \bar{x}, w^*)$ extending $(\sigma, \bar{x})$ along p. So, $\bar{x}, w^* \to \bar{y}, z$ as desired. ⊣

To complete the proof of Theorem 3.2, it is enough to show the following.

LEMMA 3.8. *Let* T *satisfy all of the properties, and let* p *be an α-generic path. If $\mathcal{B} = \mathcal{A}^p$, then $T \cong T_{\mathcal{B}}$.*

PROOF. For an α-generic path q through T, we write $\mathcal{A}^q$ for the resulting structure. For an α-generic path r through $T_{\mathcal{B}}$, we write $\mathcal{B}^r$ for the resulting structure. By Lemma 3.7, these structures are all isomorphic. Moreover, in either tree, the structures obtained from α-generic paths that extend a specific node assign the same Π_α-type to the variables of that node. With this in mind, we define an isomorphism F from T onto $T_{\mathcal{B}}$, preserving the tree structure, and with the special feature that for $F(\sigma) = \tau$, where both nodes are at level n and for the tuple $\bar{x}$ consisting of the first n variables, the Π_α-type of $\bar{x}$ in structures $\mathcal{A}^q$ for α-generic paths q through T that extend σ, matches the Π_α-type of $\bar{x}$ in $\mathcal{B}^r$ for α-generic paths r through $T_{\mathcal{B}}$ that extend τ. Let $\mathcal{F}$ be the set of partial 1–1 functions f mapping a finite subtree S of T onto a finite subtree S' of $T_{\mathcal{B}}$, such that f preserves the tree structure and has the further special feature. We show that $\mathcal{F}$ has the back-and-forth property.

$F \neq \emptyset$: The function taking the root node of T to the root node of $T_{\mathcal{B}}$ has the special feature, since for α-generic paths q through T and r through $T_{\mathcal{B}}$, $\mathcal{A}^q \cong \mathcal{B}^r$.

Forth: Suppose f is in $\mathcal{F}$, mapping S onto S'. Let σ' be a further node in T. We may suppose that σ' is a successor to some node σ of S. Say that $F(\sigma) = \tau$. Let $\bar{x}$ be the variables of σ. For all α-generic paths q extending σ and all α-generic paths r extending τ, $(\mathcal{A}^q, \bar{x}) \cong (\mathcal{B}^r, \bar{x})$ under an isomorphism $F_{q,r}$. Let x' be the next variable after $\bar{x}$. Let r be an α-generic path extending τ. Then $F_{q,r}$ takes $\bar{x}$ to $\bar{x}$ and takes x' to some y. Let τ' be the successor of τ with $\tau' \prec q$. By α-permutation, τ has another successor $\tau*$ such that $(\tau', \bar{x}, y) \leq_\alpha (\tau^*, \bar{x}, x')$. If r^* is an α-generic extension of τ^*, then $(\mathcal{A}^q, \bar{x}, x') \cong (\mathcal{B}^r, \bar{x}, x')$. We let f' be the extension of f that takes σ' to $\tau*$. This is in $\mathcal{F}$.

Back: Suppose f is in $\mathcal{F}$, mapping the subtree S of T onto the subtree S' of $T_{\mathcal{B}}$, as above, and let τ' be a further node of $T_{\mathcal{B}}$, where τ' is a successor of some τ in S'. Say that $f(\sigma) = \tau$. Proceeding just as above, we find a successor σ^* of σ such that the extension of f taking σ^* to τ' is in $\mathcal{F}$. ⊣

§4. Sharpness of the bound. In the previous section, we showed that if $\alpha \geq 1$ is an X-computable ordinal, then $\mathbb{T}^\alpha$ is X-effective $\Pi_{2\alpha+2}$. In this section, we show that the bound is sharp.

THEOREM 4.1. *For an* X*-computable ordinal α, the class $\mathbb{T}^\alpha$ is complete* X*-effective $\Pi_{2\alpha+2}$.*

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

What drives the complexity of $\mathbb{T}^\alpha$ is the Scott rank. Let L be the language of graphs, and let $Mod(L)^\alpha$ consist of the structures in $Mod(L)$ of Scott rank at most α. It is well-known (see [23, Lemma II.67]) that for $\alpha \geq 1$ an X-computable ordinal, $Mod(L)^\alpha$ is X-effective $\Pi_{2\alpha+2}$. To prove Theorem 4.1, we prove the following.

THEOREM 4.2. *For all* X, *and all* X*-computable ordinals* $\alpha \geq 1$, $Mod(L)^\alpha$ *is* $\Pi_{2\alpha+2}$*-hard as a subset of the Polish space* $Mod(L)$.

Before proving Theorem 4.2, we show how it yields Theorem 4.1.

PROOF OF THEOREM 4.1, ASSUMING THEOREM 4.2. Recall that the Friedman–Stanley embedding is Turing computable and preserves Scott rank by [12]. Take a class $D \subseteq 2^\omega$ that is X-effective $\Pi_{2\alpha+2}$. By Theorem 4.2, there is an X-computable operator $\Phi : 2^\omega \to Mod(L)$ such that for all $f \in 2^\omega$, $f \in D$ iff $\Phi(f) \in Mod(L)^\alpha$. Composing Φ with the Friedman–Stanley embedding, we get an X-computable operator $\Psi : 2^\omega \to \mathbb{T}$ such that $f \in D$ iff $\Psi(f) \in \mathbb{T}^\alpha$. ⊣

The remainder of this section is devoted to the proof of Theorem 4.2. We use ideas from [13].

DEFINITION 14. Let γ be an X-computable ordinal.

1. $h^X_{\Sigma_\gamma}(f)$ is the set of indices for X-effective Σ_γ sets that contain f.
2. $h^X_{\Pi_\gamma}$ is the set of indices for X-effective Π_γ sets that contain f.

The following observation is stated in [13]. It is analogous to the result of Ash saying that for α an X-computable ordinal, the set of (indices for) X-computable Σ_γ (Π_γ) sentences true in a structure $\mathcal{A}$ is Σ^0_γ (Π^0_γ) relative to X and $\mathcal{A}$, with all imaginable uniformity. The proof, like that of Ash, is a straightforward induction on γ.

PROPOSITION 4.3. *For* $X \subseteq \omega$, $\gamma \geq 1$ *an* X*-computable ordinal, and* $f \in 2^\omega$, $h^X_{\Sigma_\gamma}(f)$ *is* Σ^0_γ *relative to* (X, f), *and* $h^X_{\Pi_\gamma}(f)$ *is* Π^0_γ *relative to* (X, f), *with all imaginable uniformity.* (*In fact, these sets are even complete for their respective point classes.*)

Below, for each X, we fix a path through $\mathcal{O}^X$ and identify each X-computable ordinal α with its unique notation on this path. We use this notation to more explicitly state what we are trying to prove.

THEOREM 4.4. *Given a set* X *and an* X*-computable ordinal* α, *we can compute a function* Φ *that takes each pair* (f, b), *where* $f \in 2^\omega$ *and* b *is an index for an* X*-effective* $\Pi_{2\alpha+2}$ *set* B, *to a graph* $\mathcal{A}^{(X,f)}_b$ *such that* $SR(\mathcal{A}^{(X,f)}_b) \leq \alpha$ *iff* $f \in B$.

Given an index b for an X-effective $\Pi_{2\alpha+2}$ set D, we get, effectively, a sequence of indices $(d_n)_{n\in\omega}$ for X-effective $\Sigma_{2\alpha+1}$ sets D_n such that $B = \cap_n D_n$. We will prove the following.

Technical Theorem. Given a set X and an X-computable ordinal α, we have an X-computable operator Ψ that takes each pair (f, d), where $f \in 2^\omega$ and d is an index for an X-effective $\Sigma_{2\alpha+1}$ set D, to a graph $\mathcal{B}^{(X,f)}_d$ such that $SR(\mathcal{B}^{(X,f)}_d) \leq \alpha$ iff $f \in D$.

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

Before proving the Technical Theorem, let us see how it gives Theorem 4.4.

PROOF OF THEOREM 4.4 FROM TECHNICAL THEOREM. Given X, f, and $(d_n)_{n\in\omega}$, we obtain the structure $\mathcal{A}$ from a sequence of structures $\mathcal{B}_n$, computable uniformly in X, f, and n, such that $\mathcal{B}_n$ has rank at most α iff $f \in D_n$. We pass effectively from the sequence $(\mathcal{B}_n)_{n\in\omega}$ to a single structure $\mathcal{B}$ for the language obtained by adding to L unary relation symbols U_n, for $n \in \omega$. In $\mathcal{B}$, the relations U_n partition the universe of $\mathcal{B}$ into infinite sets, and the binary relation E puts a copy of $\mathcal{B}_n$ on the set U_n. The rank of $\mathcal{B}$ is at most α iff all $\mathcal{B}_n$ have rank at most α. By Proposition 1.10, we can pass effectively from $\mathcal{B}$ to a graph $\mathcal{A}$, preserving isomorphism and Scott rank. ⊣

We split the proof of the Technical Theorem into four sections: Section 4.1 deals with $\alpha = 1$, Section 4.2 deals with α finite and greater than 1, Section 4.3 deals with limit α, and Section 4.4 deals with $\alpha = \lambda + n$, for λ a limit ordinal and $n > 0$ finite.

4.1. The case $\alpha = 1$. In this section, we prove the following.

Technical Theorem for $\alpha = 1$. Given a set X, we have an X-computable operator Ψ that takes each pair (f, d), where $f \in 2^\omega$ and d is an index for an X-effective Σ_3 set D, to a graph $\mathcal{B}_d^{(X,f)}$ such that $SR(\mathcal{B}_d^{(X,f)}) \leq 1$ iff $f \in D$ $(e \in h^X_{\Sigma_3}(f))$.

DEFINITION 15. For $X \subseteq \omega$ and $f \in 2^\omega$, $Cof^{(X,f)}$ is the set of n such that $W_n^{(X,f)}$ is co-finite.

The standard proof that Cof is Σ^0_3-hard has a great deal of uniformity. Given an index d for a Σ^0_3 set $S_d \subseteq \omega$ and $k \in \omega$, we can effectively find a number n such that W_n is co-finite iff $k \in S_d$. This is also true when we relativize. The indices d for sets S that are Σ^0_3 relative to (X, f) form a computable set I_{Σ_3}, which is the same for all (X, f). Given $d \in I$ and $k \in \omega$, we can compute n such that for all $X \subseteq \omega$ and all $f \in 2^\omega$, $W_n^{(X,f)}$ is co-finite iff $k \in S_d$, where this is the set with index d as a set Σ^0_3 relative to (X, f). Thus, to prove the Technical Theorem for $\alpha = 1$, it is enough to prove the following.

LEMMA 4.5. *There is a Turing operator taking each triple (X, f, n), where $X \subseteq \omega$, $f \in 2^\omega$, and $n \in \omega$, to a graph $\mathcal{A}_n^{(X,f)}$, a bunch of daisies, such that $n \in Cof^{(X,f)}$ iff $\mathcal{A}_n^{(X,f)}$ has rank at most* 1.

Recall that a *daisy* consists of a *center* that is the single common point of cycles of various sizes. The cycles are called *petals*. A *bunch* of daisies is a graph whose connected components are daisies.

PROOF. We build $\mathcal{A}_n^{(X,f)}$, following a uniform procedure based on $W_n^{(X,f)}$. We enumerate the centers of the daisies as c_i for $i \in \omega$. If $i \in W_n^{(X,f)}$, then c_i will have one petal of length $k + 3$ for each $k \in W_n^{(X,f)}$. If $i \notin W_n^{(X,f)}$, then c_i will have one petal of length $i + 3$, and one of length $k + 3$ for each $k < i$ such that $k \in W_n^{(X,f)}$. This is all—the daisy with center c_i has no further petals, and each graph element belongs to one of these daisies. At stage s, we have built a finite graph, with centers c_i for $i < s$ and petals based on elements enumerated into $W_n^{(X,f)}$ by stage s. In the limit, we get $\mathcal{A}^{(X,f)}$ with a completed daisy around c_i, just as described.

For simplicity, we suppose that $0 \in W_n^{(X,f)}$. This means that for $i \notin W_n^{(X,f)}$, c_i has at least two petals, one of length 3 and one of length $i + 3$. Then we can define

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

the set of centers by a simple existential formula saying that x is connected by the edge relation to at least three points. Without the simplifying assumption, we may have a center c_i with just one petal, all elements of which are automorphic, a case we wish to avoid. We must show that $n \in Cof^{(X,f)}$ iff $\mathcal{A}^{(X,f)}$ has rank at most 1, where this means that the orbits of all tuples are defined by existential formulas.

First, suppose $n \in Cof^{(X,f)}$. There are finitely many i such that $i \notin W_n^{(X,f)}$. For each such i, c_i is the only center with a petal of size $i+3$. We have an existential definition of c_i. All c_j for $j \in W_n^{(X,f)}$ are automorphic. We want an existential definition for the common orbit of these c_j. Take b greater than all $j \notin W_n^{(X,f)}$. Then to define the common orbit of c_i for $i \in W_n^{(X,f)}$, we say that it is a center with a petal of size $b+3$. For a tuple $\bar{a}$ on the daisy with center c_i, we have an existential formula $\psi(\bar{x}, c_i)$, describing a finite set of petals that contains $\bar{a}$ and saying how the elements a_i sit on these petals, in relation to c_i. The orbit of an arbitrary tuple $\bar{a}_1, \ldots, \bar{a}_m$, with $\bar{a}_j$ in the daisy with center c_i is defined by an existential formula $\varphi(\bar{x}_f, \ldots, \bar{x}_m)$ saying that there exist y_j in the orbit of c_j such that $\bigwedge_{1 \le j \le m} \psi(\bar{x}_j, y_j)$.

Now, suppose $n \notin Cof^{(X,f)}$. Consider any existential formula $\varphi(x)$ true of c_i for $i \in W_n^{(X,f)}$. Fix one such c_i. The truth of $\varphi(x)$ is witnessed by a finite subgraph G consisting of c_i, with finitely many of its petals, and G' consisting of finite parts of the daisies around finitely many other centers c_j. Take $b \notin W_n^{(X,f)}$ such that $b+3$ is greater than the size of any petal in G. Let G^* be the modification of G consisting of G' and the daisy with center c_b and all of its petals. Since $i < b$, all petals around c_i in G are matched by petals around c_b. There is an embedding of G in $\mathcal{A}_n^{(X,f)}$ that takes c_i to c_b, fixing G' pointwise. Thus, in $\mathcal{A}_n^{(X,f)}$, c_b satisfies $\varphi(x)$. Since $b \notin W_n^{(X,f)}$, c_b has a petal of size $b+3$, while c_i does not, so $\varphi(x)$ does not define the orbit. ⊣

4.2. The case $n > 1$ finite. Our goal in this section is to prove the following.

Technical Theorem for finite $n > 1$. Given a set X, we have an X-computable operator Ψ that takes each pair (f, d), where $f \in 2^\omega$ and d is an index for an X-effective Σ_{2n+1} set D, to a graph $\mathcal{B}_d^{(X,f)}$ such that $SR(\mathcal{B}_d^{(X,f)}) \le n$ iff $f \in D$.

Outline of proof *The structure $\mathcal{B}_d^{(X,f)}$ will be last in a sequence $\mathcal{B}_1, \mathcal{B}_2, \ldots, \mathcal{B}_n$ such that*

(a) *$\mathcal{B}_1$ is computable in $(X, f)^{2(n-1)}$ and* f, *with rank* 1 *if $f \in D$ and* 2 *otherwise*;
(b) *for $1 \le i < n$, $\mathcal{B}_{i+1}$ is computable in $(X, f)^{2(n-i)}$, with rank equal to $1 + SR(\mathcal{B}_i)$.*

Given X, f, and the index d for an X-effective Σ_{2n+1} set D, we compute an index d_1 for $\mathcal{B}_1$ that is $\Delta^0_{2(n-1)+1}$ relative to (X, f). For $1 \le i < n$, from d_i an index for $\mathcal{B}_i$ that is $\Delta^0_{2(n-i)+1}$ relative to (X, f), we compute an index d_{i+1} for $\mathcal{B}_{i+1}$ that is $\Delta^0_{2(n-(i+1))+1}$ relative to X and f. We arrive at an index for $\mathcal{B}_d^{(X,f)} = \mathcal{B}_n$ computable in (X, f). The rank of $\mathcal{B}_n$ is n if $f \in D$ and $n+1$ otherwise. To complete the proof of the Technical Theorem for finite $n > 1$, we flesh out items (a), computing the index for $\mathcal{B}_1$, and (b), passing from the index for $\mathcal{B}_i$ to the index for $\mathcal{B}_{i+1}$. Note that being $\Delta^0_{2(n-i)+1}$ relative to (X, f) is the same as being computable in $(X, f)^{2(n-i)}$.

For (a), we recall Section 4.1, where we described a uniform effective procedure that takes each triple (X, f, d), where $X \subseteq \omega$, d is an index for an X-effective Δ_3 set

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

D, and $f \in 2^\omega$, to a graph $\mathcal{A}_d^{(X,f)}$ that has Scott rank 1 if $f \in D$ and 2 otherwise. Recall that $h^X_{\Sigma_3}(f)$ is the set of indices for X-effective Σ_3 sets that contain f. This set is Σ^0_3 relative to (X, f), with a known index b. In Section 4.1, we construct $\mathcal{A}_d^{(X,f)}$ so that it has rank 1 if d is in the set with index b as a set Σ^0_3 relative to (X, f). The construction shows the following.

PROPOSITION 4.6 (Technical Theorem for $\alpha = 1$, re-stated). *Given* (Y, b, n), *where* b *is an index for a set* S *that is* Σ^0_3 *relative to* Y, *we can effectively find an index for a graph* $\mathcal{A}$ *computable relative to* Y, *such that*

$$SR(\mathcal{A}) = \begin{cases} 1 & \text{if } n \in S \\ 2 & \text{otherwise} \end{cases}.$$

Note that the operator from the Technical Theorem for $\alpha = 1$ that outputs the diagram of a graph is Y-computable. That said, the above operator does not use Y as it is only outputting an index for the computation which is determined computably.

Let d be an index for an X-effective Σ_{2n+1} set D, and let $f \in 2^\omega$. The set $h_{\Sigma_{2n+1}}(f)$ is Σ^0_{2n+1} relative to (X, f), with known index, and it is Σ^0_3 relative to $(X, f)^{2(n-1)}$, also with known index. Apply the proposition above, letting $Y = (X, f)^{(2(n-1))}$, letting $S = h_{\Sigma^0_{2n+1}}(f)$, letting b be the known index for S as a set Σ^0_3 relative to Y, and letting n be d. By the proposition, we can find an index b_1 for a graph $\mathcal{B}_1$, computable relative to $(X, f)^{2(n-1)}$, such that $\mathcal{B}_1$ has rank 1 if $d \in h^X_{\Sigma_{2n+1}}(f)$ (which means $f \in D$), and 2 otherwise.

(b) For $1 \leq i < n$, to pass from an index b_i for $\mathcal{B}_i$ computable in $(X, f)^{2(n-i)}$ to an index b_{i+1} for $\mathcal{B}_{i+1}$ computable in $(X, f)^{2(n-(i+1))}$, we use Harrison-Trainor's "unfriendly jump inversion," adding some further information. In [14], Harrison-Trainor proved the following.

THEOREM 4.7. *Let* $S \subseteq \omega$ *be a* Σ^0_3 *set. There is a uniformly computable sequence of graphs* $\mathcal{G}, \mathcal{H}, \mathcal{C}_i$ *such that*

1. $i \in S \iff \mathcal{C}_i \cong \mathcal{G}$;
2. $i \notin S \iff \mathcal{C}_i \cong \mathcal{H}$;
3. $\mathcal{H} \not\leq_2 \mathcal{G}$;
4. $SR(\mathcal{G}) = 1$, $SR(\mathcal{H}) = 2$;
5. $\mathcal{G} \equiv_1 \mathcal{H}$.

REMARK. Item (5) above is not explicitly stated in [14], but it is immediate from the construction given there.

Using Theorem 4.7, relativized to Y, Harrison-Trainor showed that given an index a for a graph $\mathcal{A}$ computable in Y'' and of finite Scott rank, we can find an index b for a structure $\mathcal{B}$ computable in Y such that $SR(\mathcal{B}) \leq 1 + SR(\mathcal{A})$. There is a copy of $\mathcal{A}$ defined in $\mathcal{B}$—the formula defining the universe is quantifier-free, and there are Σ_2 formulas defining the edge relation and its negation. However, looking more closely at the construction, we will see that $SR(\mathcal{B}) = 1 + SR(\mathcal{A})$.

THEOREM 4.8. *Given* Y *and an index* a *for a graph* $\mathcal{A}$ *computable in* Y'', *and of finite rank, we can find an index* b *for a structure* $\mathcal{B}$ *computable in* Y *such that* $SR(\mathcal{B}) = 1 + SR(\mathcal{A})$.

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

By Proposition 1.10, we can effectively replace the original $\mathcal{B}$ by a graph of the same rank. The structure $\mathcal{B}$ (in its initial version) has unary predicates U, V that partition the universe into disjoint infinite sets. The set U will correspond to the universe of a copy of $\mathcal{A}$, but the edge relation E is coded. For simplicity, we identify U with the universe of $\mathcal{A}$. The structure $\mathcal{B}$ has a unary function c from V to the set of triples (x, y, i), where x, y are distinct elements of U and $i \in 2$.[2] For each such triple, (x, y, i), the c-inverse image is an infinite set $V_{(x,y,i)}$. Finally, the structure $\mathcal{B}$ has a binary relation R that, on $V_{(x,y,i)}$ puts a copy of $\mathcal{G}$ if xEy and $i = 1$ or if $\neg xEy$ and $i = 0$, and $\mathcal{H}$ otherwise. The relation R is the union of its restrictions to the sets $V_{(x,y,i)}$.

The fact that from the index for $\mathcal{A}$ computable in Y'', we can compute the index for $\mathcal{B}$ computable in Y, is clear. The relations xEy and $\neg xEy$ are both Σ^0_3 in Y, with known indices, so we can put copies of $\mathcal{G}$, $\mathcal{H}$ on the sets $V_{(x,y,i)}$ by Theorem 4.7, Properties 1 and 2 as required. Since $\mathcal{G}$ has Scott rank 1, it has a Π_2 Scott sentence φ. This sentence is false in $\mathcal{H}$. For the copy of $\mathcal{A}$ defined in $\mathcal{B}$, the universe is defined by the quantifier-free formula $U(x)$. The relation $\mathcal{A} \models xEy$ has a Π_2 definition saying that φ holds in $V_{(x,y,0)}$, and a Σ_2 definition saying that $\neg\varphi$ holds in $V_{(x,y,1)}$.

We know that if $\mathcal{A}$ has rank n, $\mathcal{B}$ has rank at most $1 + n$. We must show that the rank of $\mathcal{B}$ is not less than $1 + n$. It is enough to show that if $\mathcal{A}$ is an infinite graph of rank n, not less, then the resulting $\mathcal{B}$ has rank $1 + n$, not less. We start with an analysis of the back-and-forth relations within unfriendly jump inversions which will help us with these calculations.

LEMMA 4.9. *Let $\bar{b} = \bar{a}, \bar{v}$, and $\bar{b}' = \bar{a}', \bar{v}'$ be tuples in $\mathcal{B}$, where $\bar{a}, \bar{a}'$ are in $\mathcal{A}$, $\bar{v}, \bar{v}'$ are in* V. *Let $\bar{v}_{(x,y,i)}$ be the part of $\bar{v}$ in $V_{(x,y,i)}$, and let $\bar{v}'_{(x',y',i)}$ be the corresponding part of $\bar{v}'$. We suppose that $x, y \in \bar{a}$ and $x', y' \in \bar{a}'$.*

(a) *$(\mathcal{B}, \bar{a}, \bar{v}) \leq_1 (\mathcal{B}, \bar{a}', \bar{v}')$ iff the equality relation on $\bar{a}$ matches that on $\bar{a}'$, and for all (x, y, i) such that $\bar{v}_{(x,y,i)}$ is non-empty, $(V_{(x,y,i)}, \bar{v}_{x,y,i}) \leq_1 (V_{(x',y',i)}, \bar{v}_{(x,y,i)})$.*

(b) *For $m \geq 2$, $(\mathcal{B}, \bar{a}, \bar{v}) \leq_m (\mathcal{B}, \bar{a}, \bar{v})$ iff $(\mathcal{A}, \bar{a}) \leq_{m-1} (\mathcal{A}, \bar{a}')$ and for each (x, y, i) such that $\bar{v}_{(x,y,i)}$ is non-empty, $(V_{(x,y,i)}, \bar{v}_{(x,y,i)}) \cong (V_{(x',y',i)}, \bar{v}_{(x,y,i)})$.*

PROOF. We proceed by induction on m, starting with the base case $m = 1$. In $\mathcal{B}$, existential formulas say nothing about a tuple $\bar{a}$ in $\mathcal{A}$ beyond the equality relation, and existential statements about a tuple $\bar{v}$ in V splits into separate statements about the tuples $\bar{v}_{(x,y,i)}$ in $V_{(x,y,i)}$. Thus, $(\mathcal{B}, \bar{a}, \bar{v}) \leq_1 (\mathcal{B}, \bar{a}', \bar{v}')$ iff the equalities on $\bar{a}$ and $\bar{a}'$ match and for each (x, y, i) such that $\bar{v}_{(x,y,i)}$ is non-empty, $(V_{(x,y,i)}, \bar{v}_{(x,y,i)}) \leq_1 (V_{(x',y',i)}, \bar{v}_{(x',y',i)})$.

Assuming that the statement holds for m, where $m \geq 1$, we prove it for $m + 1$. Suppose $(\mathcal{B}, \bar{a}, \bar{v}) \leq_{m+1} (\mathcal{B}, \bar{a}', \bar{v}')$. For all $\bar{c}'$ in $\mathcal{A}$, there exists $\bar{c}$ such that $(\mathcal{B}, \bar{a}', \bar{v}', \bar{c}') \leq_m (\mathcal{B}, \bar{a}, \bar{v}, \bar{c})$. By inductive hypothesis, $(\mathcal{A}, \bar{a}', \bar{c}') \leq_{m-1} (\mathcal{A}, \bar{a}, \bar{c})$. Thus, $(\mathcal{A}, \bar{a}) \leq_m (\mathcal{A}, \bar{a}')$. If $m = 1$, by induction hypothesis, we similarly have that $(V_{(x,y,i)}, \bar{v}_{(x,y,i)}) \leq_1 (V_{(x',y',i)}, \bar{v}_{(x',y',i)})$ for each x, y. For one of the $i \in 2$, $V_{(x,y,i)} \cong \mathcal{G}$, which means $(V_{(x,y,i)}, \bar{v}_{(x,y,i)}) \cong (V_{(x',y',i)}, \bar{v}_{(x',y',i)})$ as $\mathcal{G}$ is Scott rank 1. By construction, this also means that $(V_{(x,y,1-i)}, \bar{v}_{(x,y,1-i)}) \cong (V_{(x',y',1-i)}, \bar{v}_{(x',y',1-i)})$.

[2] To be precise, we may think of relations $c_0(v, x, y)$, which holds if $c(v) = (x, y, 0)$, and $c_1(v, x, y)$, which holds if $c(v) = (x, y, 1)$.

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

If $m > 1$, then $(V_{(x,y,i)}, \bar{v}_{(x,y,i)}) \cong (V_{(x',y',i)}, \bar{v}_{(x',y',i)})$ follows immediately from the induction hypothesis.

Now, suppose $(\mathcal{A}, \bar{a}) \leq_m (\mathcal{A}, \bar{a}')$ and for the relevant triples (x, y, i) and corresponding (x', y', i), $(V_{(x,y,i)}, \bar{v}_{(x,y,i)}) \cong (V_{(x',y',i)}, \bar{v}_{(x',y',i)})$. We show that $(\mathcal{B}, \bar{a}, \bar{v}) \leq_{m+1} (\mathcal{B}, \bar{a}', \bar{v}')$. For $\bar{c}'$ in $\mathcal{A}$ and $\bar{w}'$ in V, we need $\bar{c}$ and $\bar{w}$ such that $(\mathcal{B}, \bar{a}', \bar{c}', \bar{v}', \bar{w}') \leq_m (\mathcal{B}, \bar{a}, \bar{c}, \bar{v}, \bar{w})$. Extending, if necessary, we suppose that each element of $\bar{w}'$ is in $V_{(x',y',i)}$ for some $x', y' \in \bar{a}', \bar{c}'$. Since $(\mathcal{A}, \bar{a}) \leq_m (\mathcal{A}, \bar{a}')$, we have $\bar{c}$ such that $(\mathcal{A}, \bar{a}', \bar{c}') \leq_{m-1} (\mathcal{A}, \bar{a}, \bar{c})$. There is no problem choosing $\bar{w}$; we select $\bar{w}$ according to the isomorphism between $(V_{(x,y,i)}, \bar{v}_{(x,y,i)}) \cong (V_{(x',y',i)}, \bar{v}_{(x',y',i)})$ in each of the relevant $V_{(x,y,i)}$. By inductive hypothesis, this guarantees that $(\mathcal{B}, \bar{a}', \bar{c}', \bar{v}', \bar{w}') \leq_m (\mathcal{B}, \bar{a}, \bar{c}, \bar{v}, \bar{w})$, as desired. ⊣

We are now ready to complete the proof of Theorem 4.7. We begin with the case where the Scott rank of $\mathcal{A}$ is equal to one.

LEMMA 4.10. *Let $\mathcal{A}$ be a graph of rank* 1, *with pairs of distinct elements $(x, y), (x', y')$ such that xEy and $\neg x'Ey'$. Then the structure $\mathcal{B}$ obtained from $\mathcal{A}$ must have rank greater than* 1.

PROOF. Since $\mathcal{G}$ and $\mathcal{H}$ satisfy the same existential sentences, the pairs (x, y) and (x', y') satisfy the same existential formulas. By assumption, these pairs are not automorphic. Therefore, $\mathcal{B}$ must have rank greater than 1. ⊣

For $n \geq 2$, we now show that for $\mathcal{A}$ of rank at least n, the resulting $\mathcal{B}$ has rank at least $1 + n$.

LEMMA 4.11. *Let $n \geq 2$. If $\mathcal{A}$ is a graph of rank at least* n, *then $\mathcal{B}$ has rank at least $1 + n$.*

PROOF. The fact that $\mathcal{A}$ does not have rank at most $n - 1$ means that there is some $\bar{a}$ such that for all $\bar{b}$, there exist $\bar{a}', \bar{b}'$ such that

1. $(\mathcal{A}, \bar{a}, \bar{b}) \leq_{n-2} (\mathcal{A}, \bar{a}', \bar{b}')$;
2. $(\mathcal{A}, \bar{a}) \not\leq_{n-1} (\mathcal{A}, \bar{a}')$.

We claim that this same $\bar{a}$ witnesses that $\mathcal{B}$ does not have rank at most n. We must show that for each $\bar{b}$ in $\mathcal{B}$, there exist $\bar{a}', \bar{b}'$ such that

1. $(\mathcal{B}, \bar{a}, \bar{b}) \leq_{n-1} (\mathcal{B}, \bar{a}', \bar{b}')$;
2. $(\mathcal{B}, \bar{a}) \not\leq_n (\mathcal{B}, \bar{a}')$.

Take $\bar{b}$ in $\mathcal{B}$. Let $\bar{c}$ be the part of $\bar{b}$ in $\mathcal{A}$, and let $\bar{v}$ be the part in V. Extending, if necessary, we suppose that $\bar{v}$ breaks up into tuples $\bar{v}_{(x,y,i)}$ for $x, y \in \bar{a}, \bar{c}$. Since $\bar{a}$ witnesses that $\mathcal{A}$ does not have Scott rank at most $n - 1$, we have $\bar{a}', \bar{c}'$ such that $(\mathcal{A}, \bar{a}, \bar{c}) \leq_{n-1} (\mathcal{A}, \bar{a}', \bar{c}')$ and $(\mathcal{A}, \bar{a}) \not\leq_n (\mathcal{A}, \bar{a}')$. For each (x, y, i) such that $v_{(x,y,i)}$ is a non-empty tuple in $\bar{v}$, choose $\bar{v}'_{(x',y',i)}$ a corresponding tuple such that $(V_{(x,y,i)}, \bar{v}_{(x,y,i)}) \cong (V_{(x',y',i)}, \bar{v}'_{(x',y',i)})$. We can do this, since xEy iff $x'Ey'$ by $(\mathcal{A}, \bar{a}, \bar{b}) \leq_{n-2} (\mathcal{A}, \bar{a}', \bar{b}')$. By Lemma 4.8, $(\mathcal{B}, \bar{a}, \bar{c}, \bar{v}) \leq_{n-1} (\mathcal{B}, \bar{a}', \bar{c}', \bar{v}')$ and $(\mathcal{B}, \bar{a}) \not\leq_n (\mathcal{B}, \bar{a}')$. ⊣

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

4.3. The case α a limit ordinal. In this section, we prove the following.

THEOREM 4.12. *For $X \subset \omega$ and α an* X*-computable limit ordinal, there is an* X*-computable operator that takes pairs (f, d), where $f \in 2^\omega$ and d is an index for an* X*-effective $\Sigma_{\alpha+1}$ set $D \subseteq 2^\omega$, to a graph $\mathcal{A}_d^{(X,f)}$ that has Scott rank at most α iff $f \in D$.*

PROOF. Given d, we can find a sequence of indices (d_n) for X-effective Π_α sets D_n with union D. We may assume that D_n is an increasing union by taking unions of initial sequences if needed. Let $S = h^X_{\Pi_\alpha}(f)$. For $Y = (X, f)$, S is Π_α relative to Y. From α, we compute an increasing sequence of ordinals $(\alpha_n)_{n\in\omega}$ with limit α. By a result from [3], for each n, we can effectively associate with the index d_n an ordering $\mathcal{L}_n$, computable in Y, such that $\mathcal{L}_n$ has type ω^α if $d_n \in S$ and some $\beta_n < \omega^\alpha$, otherwise.

The initial version of $\mathcal{A}_d^{(X,f)}$ is a structure $\mathcal{B}$ with an equivalence relation $\sim$ that partitions the universe into disjoint sets V and U_n (not named). There is a relation that puts an ordering of type ω^α on V and an ordering of type $\omega^{\alpha_n} \cdot L_n$ on U_n. We must show that $SR(\mathcal{A}_d^{(X,f)}) \leq \alpha$ iff $f \in D$. Note that if $d_n \notin S$, then U_n has type less than ω^α. On the other hand, if $d_n \in S$, then U_n has type equal to ω^α.

Consider the case that $f \in D$. In particular, for some j, for all $i \geq j$ we have that $d_i \in S$. This means that the structure $\mathcal{A}_d^{(X,f)}$ will contain infinitely many copies of linear orderings of the form ω^α and finitely many equivalence classes with ordinals strictly less than ω^α. Let these finitely many ordinals be denoted $A_0 \leq \cdots \leq A_N$. Note that, by the ordinal analysis in [2], there is a sentence φ_N in $\Pi_{<\alpha}$ that states that a given ordinal is longer than A_N. Furthermore, each A_i has $SR(A_i) < \alpha$, so it has a $\Pi_{<\alpha}$ Scott sentence ψ_i and ω^α has a $\Pi_{\alpha+1}$ Scott sentence χ. We obtain a $\Pi_{\alpha+1}$ Scott sentence for $\mathcal{A}_d^{(X,f)}$ by expressing that for each i, there is one equivalence class satisfying ψ_i, the rest of the classes satisfy φ_N and every equivalence class has $\varphi_N \to \chi$. This gives, $SR(\mathcal{A}_d^{(X,f)}) \leq \alpha$.

Consider the case $x \notin T$. This means that for all j, $d_i \notin S$. The structure $\mathcal{A}_d^{(X,f)}$ will contain one copy of ω^α and the rest of the equivalence classes will contain a linear ordering strictly less than ω^α. Furthermore, by construction, for each i, among these smaller linear orderings, there will be an ordinal A_i that is left divisible by ω^{α_i}. Let b be the first element of the ω^α and a_i be the first element of A_i. It follows from the ordinal analysis in [2] that $(\omega^\alpha, b) \equiv_{\alpha_i} (A_i, a_i)$. In particular, for each i, there is no Σ_{α_i} description of the automorphism orbit of b. Taken together, we conclude that there is no Σ_α description of the automorphism orbit of b. Therefore, $SR(\mathcal{A}_d^{(X,f)}) > \alpha$.

By Proposition 1.10, we can pass effectively from $\mathcal{B}$ to a graph $\mathcal{A}_d^{(X,f)}$, preserving rank. ⊣

4.4. The case $\alpha = \lambda + n$, for limit λ and finite $n \geq 1$. In this section, we prove the following.

THEOREM 4.13. *For $X \subseteq \omega$ and $\alpha = \lambda + n$, where λ is an* X*-computable limit ordinal and $n \geq 1$ is finite, there is an* X*-computable operator that takes each pair*

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

(f, d), *where* $f \in 2^\omega$ *and* d *is an index for an* X*-effective* $\Sigma_{2\alpha+1} = \Sigma_{\lambda+2n+1}$ *set* D *to a graph* $\mathcal{A}_d^{(X,f)}$ *such that* $\mathcal{A}_d^{(X,f)}$ *has rank at most* $\lambda + n$ *iff* $f \in D$.

Here is an outline of the proof. Recall that $f \in D$ iff $d \in h^X_{\Sigma_{\lambda+2n+1}}(f)$, where $h^X_{\Sigma_{\lambda+2n+1}}(f)$ is a subset of ω that is $\Sigma^0_{\lambda+2n+1}$ relative to (X, f). Let $Y = (X, f)$.

1. By the results from Sections 4.1 and 4.2, we can pass effectively from the index d to an index a for a graph $\mathcal{A}$, computable in $Y^{\lambda+1}$, such that $SR(\mathcal{A}) \leq n$ iff $f \in D$.
2. We consider a pair of X-computable structures $\mathcal{G}, \mathcal{H} \in Mod(L)$, from [6, 10], with some special properties different from those in Section 4.2. Using the pair $\mathcal{G}, \mathcal{H}$, we pass effectively from the index a for $\mathcal{A}$, as a structure computable in $Y^{\lambda+1}$, to an index b for a structure $\mathcal{B}$ computable in Y. The construction is quite similar to that in Section 4.2.
3. Using results from [6], with some of the ideas going back to [20], we show that this $\mathcal{B}$ has the desired Scott rank; that is, $SR(\mathcal{B}) \leq \lambda + n$ iff $SR(\mathcal{A}) \leq n$.
4. Finally, we apply Proposition 1.10 to pass from $\mathcal{B}$ to the desired graph $\mathcal{A}_d^{(X,f)}$.

Below, we say more about Steps (2) and (3).

4.4.1. Properties of $\mathcal{G}, \mathcal{H}$*, construction of* $\mathcal{B}$. Our goal here is to say exactly what properties are needed for $\mathcal{G}$ and $\mathcal{H}$, and then to describe precisely the construction that takes $\mathcal{A}$ to $\mathcal{B}$. We fix X and an X-computable limit ordinal λ. In [10], it is shown that there are $\mathcal{G}, \mathcal{H}$ in $Mod(L)$ (linear orderings) with the following properties:

1. $\mathcal{G}, \mathcal{H}$ are both rigid;
2. the pair $\mathcal{G}, \mathcal{H}$ is "$\lambda + 1$-friendly"; that is, they are X-computable and the relations $\leq_\beta$ for $1 \leq \beta < \lambda + 1$ are uniformly X-c.e.;
3. $\mathcal{G} \equiv_\lambda \mathcal{H}$;
4. $\mathcal{G} \not\leq_{\lambda+1} \mathcal{H}$ and $\mathcal{H} \not\leq_{\lambda+1} \mathcal{G}$;
5. both $\mathcal{G}, \mathcal{H}$ have Scott rank at most λ.

Given $\mathcal{G}$ and $\mathcal{H}$, we define an X-effective operator Φ_λ that takes each graph $\mathcal{A}$ to a structure $\mathcal{B}$, with unary relations U, V, a ternary relation C, and a binary relation R. As in Section 4.2, U, V partition the universe into infinite sets, and we identify U with $\mathcal{A}$. The relation C serves to partition V into infinite sets $V_{(x,y)}$, where (x, y) is a pair of distinct elements of U. $C(v, x, y)$ means that $v \in V_{(x,y)}$. The relation R puts on each $V_{(x,y)}$ a copy of $\mathcal{G}$ if xEy and $\mathcal{H}$ otherwise. Because of item (4), we have $\Sigma_{\lambda+1}$ formulas saying of a pair (x, y) in U that $V_{(x,y)}$ is isomorphic to $\mathcal{G}$, or to $\mathcal{H}$.

The next lemma follows from the discussion in Chapter X.3 of [23] regarding the operator Φ_λ. However, we can give a simple proof.

LEMMA 4.14. *Let* $\mathcal{G}$ *and* $\mathcal{H}$ *be* L*-structures, with* $\Pi_{\lambda+2}$ *Scott sentences* φ_G, φ_H, *and let* Φ_λ *be the operator described above, taking each* $\mathcal{A} \in Mod(L)$ *to a structure* $\mathcal{B}$ *in which* $\mathcal{G}$ *and* $\mathcal{H}$ *are used to code the binary relation. The class of structures* $\mathcal{B}$ *isomorphic to* $\Phi_\lambda(\mathcal{A})$, *for* $\mathcal{A} \in Mod(L)$, *is* $\Pi_{\lambda+2}$.

PROOF. We axiomatize the class by a $\Pi_{\lambda+2}$-sentence ρ saying the following:

1. U is infinite;
2. for all $x \neq y$ in U, the corresponding $V_{(x,y)}$, with the restriction of the relation R, satisfies one of φ_G and φ_H.

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

It is computable Π_2 to say that U is infinite. We have a $\Pi_{\lambda+2}$ formula saying of x, y, that if both are in U and $x \neq y$, then $V_{(x,y)}$ satisfies one of the sentences φ_G or φ_H. ⊣

We also describe how to pass from a $Y^{\lambda+1}$-computable code for $\mathcal{A}$ to a Y-computable code for $\mathcal{B}$. This resembles Lemma 5.3 in [6]

LEMMA 4.15. *Given* Y *and an index* a *for a graph* $\mathcal{A}$ *computable in* $Y^{\lambda+1}$ *we can find an index* b *for a graph* $\mathcal{B} \cong \Phi_\lambda(\mathcal{A})$ *computable in* Y.

PROOF. Note that $\mathcal{G}$ and $\mathcal{H}$ are $\lambda+1$-friendly and have $\mathcal{G} \equiv_\lambda \mathcal{H}$. By the pair of structures theorem [3], this means that distinguishing $\mathcal{G}$ and $\mathcal{H}$ is $\Delta_{\lambda+1}(Y)$ hard. In other terms, given an effective $\Delta_{\lambda+1}$ set S there is a Y effective method of constructing $\mathcal{C}_x$ with $\mathcal{C}_x \cong \mathcal{G}$ if $x \in S$ and $\mathcal{C}_x \cong \mathcal{H}$ if $x \notin S$.

Using a, uniformly given $x, y \in \mathcal{A}$, we can find whether xEy in a $Y^{\lambda+1}$-computable manner. This means that there is a $\Delta_{\lambda+1}(Y)$ set S of pairs x, y with xEy. We find the index b for $\mathcal{B}$ by describing a Y computable method of building $\mathcal{B}$. For each element in $\mathcal{A}$, we put a corresponding element with predicate U in $\mathcal{B}$. We ensure that each $x, y \in \mathcal{A}$ has $V_{x,y} \cong \mathcal{C}_{\langle x,y\rangle}$. The constructed $\mathcal{B}$ is isomorphic to $\Phi_\lambda(\mathcal{A})$, as it places a copy of $\mathcal{G}$ on $V_{x,y}$ when xEy and a copy of $\mathcal{H}$ on $V_{x,y}$ otherwise. ⊣

4.4.2. Scott rank of $\mathcal{B}$. Our goal here is to show that $\mathcal{A}$ has Scott rank at most n iff $\mathcal{B}$ has Scott rank at most $\lambda + n$. We use some known results, given in [6, Section 5].

LEMMA 4.16. *For each* Π_k*-sentence* φ*, there is a* $\Pi_{\lambda+k}$ *sentence* φ_* *such that* $\mathcal{A} \models \varphi$ *iff* $\mathcal{B} \models \varphi_*$.

This is clear from the fact that we have $\Sigma_{\lambda+1}$ formulas that define a copy of $\mathcal{A}$ in $\mathcal{B}$.

LEMMA 4.17. *For each* $\Pi_{\lambda+k}$ *sentence* ψ*, there is a* Π_k*-sentence* ψ^* *such that* $\mathcal{B} \models \psi$ *iff* $\mathcal{A} \models \psi^*$.

The argument given in [6, Section 5] works by noting that, in some sense, Φ_λ is better than X-computable. Without loss of generality, Φ_λ may produce the λ jump of $\mathcal{B}$ (defined precisely in [23, Chapter X]). Any $\Pi_{\lambda+k}$ sentence ψ in $\mathcal{B}$ corresponds to a Π_k sentence $\hat{\psi}$ in $\mathcal{B}^{(\lambda)}$. From there, the Pullback Theorem [18] gives that $\hat{\psi}$ corresponds to a Π_k sentence ψ^* in $\mathcal{A}$.

The result below now follows from a known pattern of reasoning [11, Section 6.2]. We can give a simple proof.

LEMMA 4.18. *For* $\mathcal{G}$, $\mathcal{H}$, *and* Φ *as described, and* $\Phi(\mathcal{A}) = \mathcal{B}$*, we have* $SR(\mathcal{A}) \leq n$ *iff* $SR(\mathcal{B}) \leq \lambda + n$.

PROOF. First, suppose $SR(\mathcal{A}) \leq n$. Then $\mathcal{A}$ has a Π_{n+1} Scott sentence φ. By Lemma 4.14, there is a $\Pi_{\lambda+n+1}$ sentence φ_* such that $\mathcal{A} \models \varphi$ iff $\mathcal{B} \models \varphi_*$. By Lemma 4.13, there is a $\Pi_{\lambda+2}$ sentence ρ characterizing the class of structures isomorphic to those in the range of the X-computable operator Φ defined on $Mod(L)$. The sentence $\varphi_* \,\&\, \rho$ is $\Pi_{\lambda+n+1}$, and it is a Scott sentence for $\mathcal{B}$. Therefore $SR(\mathcal{B}) \leq \lambda + n$. Now, suppose $SR(\mathcal{B}) \leq \lambda + n$. Then $\mathcal{B}$ has a $\Pi_{\lambda+n+1}$ Scott sentence ψ. By Lemma 4.15,

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

there is a Π_{n+1} sentence ψ^* such that $\mathcal{A} \models \psi^*$ iff $\mathcal{B} \models \psi$. Then ψ^* is a Scott sentence for $\mathcal{A}$. Therefore, $SR(\mathcal{A}) \leq n$. ⊣

This was the last piece we needed for the proof of Theorem 4.12, and Theorem 4.12 was all we needed to complete the proof of Theorem 4.1.

REFERENCES

[1] R. Alvir, J. F. Knight, and C. McCoy, *Complexity of Scott sentences*. ***Fundamenta Mathematicae***, vol. 251 (2020), pp. 109–129.

[2] C. J. Ash, *Recursive labeling systems and stability of recursive structures in hyperarithmetical degrees*. ***Transactions of the American Mathematical Society***, vol. 298 (1986), pp. 497–514.

[3] C. J. Ash and J. F. Knight, *Pairs of recursive structures*. ***Annals of Pure and Applied Logic***, vol. 46 (1990), pp. 211–234.

[4] ———, ***Computable Structures and the Hyperarithmetical Hierarchy***, Elsevier, Amsterdam, 2000.

[5] W. Calvert, D. Cummins, J. F. Knight, and S. Miller, *Comparing classes of finite structures*. ***Algebra and Logic***, vol. 43 (2004), pp. 374–392.

[6] R. Chen, D. Gonzalez, and M. Harrison-Trainor, *Optimal syntactic definitions of Back-and-forth types*. ***Transactions of the American Mathematical Society***. To appear. https://doi.org/10.1090/tran/9695.

[7] R. Cohen, ***Set Theory and the Continuum Hypothesis***, Dover, Garden City, 1966.

[8] H. Friedman and L. Stanley, *A Borel reducibility theory for classes of countable structures*. ***Journal of Symbolic Logic***, vol. 54 (1989), pp. 894–914.

[9] S. Gao, *Some dichotomy theorems for isomorphism relations of countable models*. ***Journal of Symbolic Logic***, vol. 66 (2001), pp. 902–922.

[10] S. S. Goncharov, V. Harizanov, J. F. Knight, C. McCoy, R. Miller, and R. Solomon, *Enumerations in computable structure theory*. ***Annals of Pure and Applied Logic***, vol. 136 (2005), pp. 219–246.

[11] D. Gonzalez and M. Harrison-Trainor, *Scott spectral gaps are bounded for linear orderings*, preprint. https://arxiv.org/pdf/2411.12084.

[12] D. Gonzalez and D. Rossegger, *Scott sentence complexities of linear orderings*. ***Journal of Symbolic Logic***, vol. 90 (2025), no. 4, pp. 1563–1592.

[13] C. Hall, J. F. Knight, and K. Lange, *Complexity of well-ordered sets in an abelian group*. ***Monatshefte für Mathematik***, vol. 208 (2025), pp. 665–686.

[14] M. Harrison-Trainor, *Relative to any non-arithmetic set*, preprint. https://arxiv.org/abs/2505.23613.

[15] M. Harrison-Trainor and A. Montalbán, *The tree of tuples of a structure*. ***Journal of Symbolic Logic***, vol. 87 (2022), pp. 21–46.

[16] G. Hjorth, *Around non-classifiability for countable torsion-free abelian groups*, ***Abelian Groups and Modules (Dublin, 1998)*** (P. C. Eklof and R. Göbel, editors), Trends in Mathematics, Birkhäuser, Basel, 1999, pp. 269–292.

[17] C. Karp, *Finite-quantifier equivalence*, ***Theory of Models (Proceedings of the 1963 International Symposium at Berkeley)*** (H. Addison and A. Tarski, editors), Studies in Logic and the Foundations of Mathematics, North-Holland, Amsterdam, 1965, pp. 407–412.

[18] J. F. Knight, S. Miller, and M. Vanden Boom, *Turing computable embeddings*. ***Journal of Symbolic Logic***, vol. 72 (2007), pp. 901–918.

[19] J. F. Knight, A. Soskova, and S. Vatev, *Coding in graphs and linear orderings*. ***Journal of Symbolic Logic***, vol. 85 (2020), pp. 673–690.

[20] C. Maher, ***On embeddings of computable structures, classes of structures, and computable isomorphism***, Ph.D. Dissertation, University of Notre Dame, 2009.

[21] A. Montalbán, *A robuster Scott rank*, ***Proceedings of the American Mathematical Society***, vol. 143 (2015), no. 12, pp. 5427–5436.

[22] ———, ***Computable Structure Theory: Within the Arithmetic***, Perspectives in Logic, Cambridge University Press, Cambridge, 2021.

[23] ———, ***Computable Structure Theory: Beyond the Arithmetic***, Perspectives in Logic, Cambridge University Press, Cambridge, 2026.

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press

[24] G. Paolini and S. Shelah, *Torsion-free abelian groups are Borel complete*. ***Annals of Mathematics***, vol. 199 (2024), pp. 1177–1224.

[25] D. Scott, *Logic with denumerably long formulas and finite strings of quantifiers*, ***Theory of Models (Proceedings of the 1963 International Symposium at Berkeley)*** (J. W. Addison, L. Henkin and A. Tarski, editors), North-Holland, Amsterdam, 1965, pp. 320–331.

[26] S. Thomas, *The classification problem for torsion-free abelian groups of finite rank*. ***Journal of the American Mathematical Society***, vol. 16 (2002), pp. 233–258.

DEPARTMENT OF MATHEMATICS
UNIVERSITY OF NOTRE DAME
HURLEY HALL, 255 HURLEY
NOTRE DAME, IN 46556
USA
E-mail: dgonza42@nd.edu
E-mail: j1knight@nd.edu

https://doi.org/10.1017/jsl.2026.10247 Published online by Cambridge University Press