Notre Dame Journal of Formal Logic
Volume 57, Number 1, 2016

Categoricity Spectra for Rigid Structures

Ekaterina Fokina, Andrey Frolov, and Iskander Kalimullin

Abstract For a computable structure $\mathcal{M}$, the categoricity spectrum is the set of all Turing degrees capable of computing isomorphisms among arbitrary computable copies of $\mathcal{M}$. If the spectrum has a least degree, this degree is called the degree of categoricity of $\mathcal{M}$. In this paper we investigate spectra of categoricity for computable rigid structures. In particular, we give examples of rigid structures without degrees of categoricity.

1 Introduction

We study algorithmic properties of isomorphisms between a computable structure $\mathcal{M}$ and its countable copies. A structure $\mathcal{A}$ is *computable* if $|\mathcal{A}|$ is a computable subset of ω and all basic predicates and functions are uniformly computable, or equivalently, the atomic diagram $D(\mathcal{A})$, thought of as a subset of ω, is computable.

We will make use of the following fact. Let σ be a computable signature, and let $\sigma = \bigcup_i \sigma_i$ for a computable sequence of finite signatures $\sigma_0 \subseteq \sigma_1 \subseteq \cdots$. Let $\mathcal{M}$ be a countable structure in the signature σ. Then $\mathcal{M}$ is computable if and only if there exists a computable sequence $(\mathcal{M}_i)_{i\in\omega}$ of finite structures such that

1. $\mathcal{M} = \bigcup_i \mathcal{M}_i$,
2. $\mathcal{M}_i \subseteq \mathcal{M}_{i+1}$, for all i, and
3. each $\mathcal{M}_i$ is a σ_i-structure with domain $\{0, \dots, t_i\}$, where the function sending i to t_i is computable.

In other words, a computable structure is a structure that may be effectively constructed step by step, where at each step we define an ever larger finite piece of the structure.

In this paper we are interested in complexity of isomorphisms between computable presentations of a countable structure. The main notion in this area of in-

Received February 14, 2013; accepted August 26, 2013
First published online October 30, 2015
2010 Mathematics Subject Classification: Primary 03C57, 03D45
Keywords: computable structure, rigid structure, computably categorical, categoricity spectrum, degree of categoricity
 10.1215/00294527-3322017

vestigation is that of *computable categoricity*. This notion has been part of computable model theory since Fröhlich and Shepherdson first produced an example of two computable fields that were isomorphic but not computably isomorphic (see [4]). Mal'cev in [7] studied the question of uniqueness of a constructive enumeration for a model and introduced the notion of a recursively stable model. Later in [8] he built isomorphic computable infinite-dimensional vector spaces that were not computably isomorphic. In the same paper he introduced the notion of an *autostable* model, which is equivalent to that of a computably categorical model. Since then, the definition of computable categoricity has been standardized and relativized to arbitrary Turing degrees $\mathbf{d}$, and has been the subject of much study (see, e.g., Goncharov [5] and Hirschfeldt, Khoussainov, Shore, and Slinko [6]).

Definition 1.1 A computable structure $\mathcal{M}$ is $\mathbf{d}$-*computably categorical* if, for every computable structure $\mathcal{A}$ isomorphic to $\mathcal{M}$, there exists a $\mathbf{d}$-computable isomorphism from $\mathcal{M}$ onto $\mathcal{A}$. In case $\mathbf{d} = \mathbf{0}$, we simply say that $\mathcal{M}$ is *computably categorical*.

As the above-mentioned early examples show, a computable structure does not need to be computably categorical. However, some noncomputable degree may compute an isomorphism between any two computable copies of the structure. The following definition intends to describe the set of degrees with this property.

Definition 1.2 Let $\mathcal{M}$ be any computable structure. The *categoricity spectrum* of $\mathcal{M}$ is the set

$$\mathrm{CatSpec}(\mathcal{M}) = \{\mathbf{d} : \mathcal{M} \text{ is } \mathbf{d}\text{-computably categorical}\},$$

the set of all Turing degrees capable of computing isomorphisms among arbitrary computable copies of $\mathcal{M}$.

Definition 1.3 A Turing degree $\mathbf{d}$ is the *degree of categoricity of* $\mathcal{M}$ if $\mathbf{d}$ is the least degree in $\mathrm{CatSpec}(\mathcal{M})$, if it exists. Finally, $\mathbf{d}$ is *categorically definable* if it is the degree of categoricity of some computable structure.

This terminology is intended to recall the notions of the spectrum of a structure $\mathcal{A}$, and the degree of the isomorphism class of $\mathcal{A}$, which was defined by Richter in [10] to be the least degree in the spectrum of $\mathcal{A}$, if such a degree exists.

The notion of the categoricity spectrum and of degrees of categoricity was introduced in Fokina, Kalimullin, and Miller [3]. The question under investigation was which Turing degrees are categorically definable. Since there are only countably many computable structures, most Turing degrees are not categorically definable. The main result of that paper gave a partial answer for the case of arithmetical degrees. It was later extended to hyperarithmetical degrees by Csima, Franklin, and Shore in [2].

Theorem 1.4 (Csima, Franklin, and Shore [2, Theorems 3.1, 3.2])

1. *For every computable ordinal* α, $\mathbf{0}^{(\alpha)}$ *is the degree of categoricity of a computable structure.*
2. *For* α *a computable successor ordinal, every degree d-c.e. in and above* $\mathbf{0}^{(\alpha)}$ *is a degree of categoricity.*

The structures produced in [2] and [3] have the following additional property: they show that the above-mentioned degrees are *strong* degrees of categoricity, where a degree $\mathbf{d}$ is a *strong degree of categoricity* if there exists a computable structure $\mathcal{A}$

with computable copies $\mathcal{B}$, $\mathcal{M}$ such that $\mathcal{A}$ is $\mathbf{d}$-computably categorical and for every isomorphism $f : \mathcal{B} \to \mathcal{M}$, $\mathbf{d} \leq_T f$. Obviously, strong degrees of categoricity are degrees of categoricity.

Furthermore, for all suitable α, for all degrees that are c.e. in and above $\mathbf{0}^{(\alpha)}$, the constructed structures are rigid. Recall that a structure is *rigid* if it has no nontrivial automorphisms. If a rigid structure $\mathcal{M}$ is $\mathbf{d}$-categorical, then it is also $\mathbf{d}$-*stable*; that is, every isomorphism from $\mathcal{M}$ onto a computable copy is $\mathbf{d}$-computable. When we pass to d-c.e. structures, we lose the property of rigidity.

Negative results were provided in both [2] and [3]. Namely, if $\mathbf{d}$ is a nonhyperarithmetical degree, then $\mathbf{d}$ cannot be a degree of categoricity. That is, nonhyperarithmetical degrees are not categorically definable. Moreover, Anderson and Csima [1] showed that not all hyperarithmetical degrees are degrees of categoricity.

Theorem 1.5 (Anderson and Csima [1, Proposition 2.1])

1. *There exists a Σ^0_2 degree that is not categorically definable.*
2. *Every degree of a set which is 2-generic relative to some perfect tree is not a degree of categoricity.*
3. *Every noncomputable hyperimmune-free degree is not a degree of categoricity.*

Not every computable structure has a degree of categoricity. The first negative example was built by Miller in [9].

Theorem 1.6 (Miller [9, Corollary 5.8]) *There exists a computable field F with splitting algorithm which is not computably categorical and such that for some $\mathbf{d}_1, \mathbf{d}_2 \in \mathrm{CatSpec}(F)$, $\mathbf{d}_1 \wedge \mathbf{d}_2 = \mathbf{0}$.*

However, these examples are not rigid structures. We present new series of computable structures with no degree of categoricity that are rigid. The main theorems we prove in the paper are the following.

Theorem 2.13 *There exists a computable rigid structure with no degree of categoricity.*

Theorem 2.14 *For every c.e. nonzero degree $\mathbf{x}$, there exists an $\mathbf{x}$-computably categorical computable rigid structure with no degree of categoricity.*

2 Rigid Structures with No Degree of Categoricity

The main goal of this section is to prove Theorems 2.13 and 2.14. Before we prove the results, we study general properties of categoricity spectra for rigid structures. We recall the following classical result.

Theorem 2.1 (Kleene–Post–Spector [11, Chapter 13.4, Theorem XVI]) *Let $\mathbf{a}_0, \mathbf{a}_1, \ldots, \mathbf{a}_n, \mathbf{a}_{n+1}, \ldots$ be an increasing sequence of degrees. Then there exist degrees $\mathbf{b}, \mathbf{c}$ that are upper bounds of this sequence, and no upper bound of $\{\mathbf{a}_n\}$ is a lower bound for $\mathbf{b}, \mathbf{c}$. Equivalently, no nonprincipal countable ideal of degrees has a least upper bound.*

Proposition 2.2 *A rigid computable structure $\mathcal{M}$ has a degree of categoricity if and only if the degrees of isomorphisms between computable copies of $\mathcal{M}$ generate a principal ideal.*

Proof Let I be an ideal generated by degrees of isomorphisms between various computable copies of $\mathcal{M}$. As $\mathcal{M}$ is rigid, the ideal I is countable. A degree $\mathbf{a}$ is a degree of categoricity if and only if $\mathbf{a}$ is the least upper bound of I. By the Kleene–Post–Spector theorem it is possible only if I is a principal ideal generated by $\mathbf{a}$. $\square$

We now define an auxiliary computable structure $\mathcal{N}$ with universe partitioned computably into four pieces:

$$\{x_i : i \in \omega\} \cup \{a_i : i \in \omega\} \cup \{b_i : i \in \omega\} \cup \{c_i : i \in \omega\}.$$

We view $\{x_i : i \in \omega\}$ as an ω-chain, while $\{a_i : i \in \omega\}$, $\{b_i : i \in \omega\}$, and $\{c_i : i \in \omega\}$ serve only as witness elements. The language has one binary predicate P. In the structure $\mathcal{N}$, P is true on all pairs of each of the following forms

$$(x_i, x_{i+1}) \qquad (x_i, a_i) \qquad (a_i, b_i) \qquad (x_i, c_i)$$

for every $i \in \omega$.

Now, given arbitrary subsets $A \subseteq B$ of ω, we are going to define a rigid structure $\mathcal{M}(A, B)$ as the substructure of $\mathcal{N}$ on the universe

$$\{x_i : i \in \omega\} \cup \{a_i : i \in \omega\} \cup \{b_i : i \in B\} \cup \{c_i : i \in A\}.$$

Figure 1 shows an example of such a structure for the case where $0 \in A$; $1 \notin B$ (hence, also $1 \notin A$); $2 \in B \setminus A$; $3 \notin B$; $4 \in A$; $5 \in B \setminus A$:

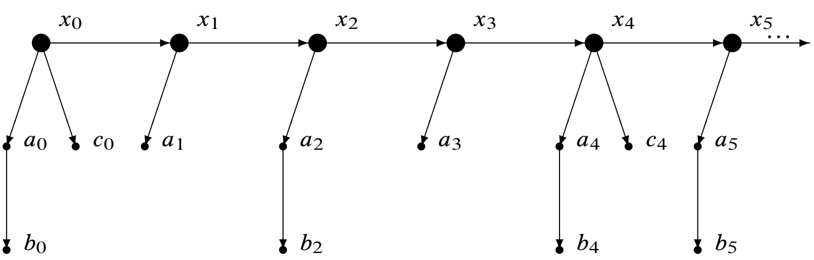

Figure 1 Example of $\mathcal{M}(A, B)$, where $0 \in A$, $1 \notin B$, $2 \in B \setminus A$, $3 \notin B$, $4 \in A$, $5 \in B \setminus A$.

Proposition 2.3 *For given sets $A \subseteq B$, the structure $\mathcal{M}(A, B)$ is computable if and only if A, B are computably enumerable.*

Proof We make use of the equivalent definition of a computable structure mentioned in the Introduction. Assume that $\mathcal{M}(A, B)$ is computable. Then $\mathcal{M}(A, B) = \bigcup_s \mathcal{M}_s$, where $\{\mathcal{M}_s\}_{s\in\omega}$ is a computable sequence of finite substructures of $\mathcal{M}(A, B)$. Then enumerate i into A_s whenever two witness elements a_i, c_i connected to x_i appear in $\mathcal{M}_s$. Enumerate i into B_s whenever in $\mathcal{M}_s$ we see a witness element b_i connected to an element connected to x_i. Both sequences $\{A_s\}_{s\in\omega}, \{B_s\}_{s\in\omega}$ are computable and $\bigcup A_s = A, \bigcup B_s = B$; therefore, A and B are c.e.

To prove the opposite direction, given A and B, consider a pair of their computable enumerations $\{A_s\}_{s\in\omega}, \{B_s\}_{s\in\omega}$, respectively. Build $\mathcal{M}_s$ defining edges according to the information from A_s, B_s; that is, whenever i is enumerated into B_s, add an edge from x_i to a chain of two witness elements (a_i, b_i), and whenever i is enumerated into A_s, add the second witness element c_i connected to x_i. Then

$\{\mathcal{M}_s\}_{s\in\omega}$ is an increasing computable sequence of finite substructures of $\mathcal{M}$ and $\bigcup \mathcal{M}_s = \mathcal{M}$; thus, $\mathcal{M}$ is computable. $\square$

From the proof of the proposition, it is clear that each computable representation of $\mathcal{M}(A, B)$ corresponds to a computable enumeration $\mathcal{P} = \langle\{A_s\}_{s\in\omega}, \{B_s\}_{s\in\omega}\rangle$ of the sets A, B.

Let $A \subseteq B$ be c.e. sets, and let $\{A_s\}_{s\in\omega}$ and $\{B_s\}_{s\in\omega}$ be computable enumerations of A and B, respectively. We now explain how different enumerations of the sets A, B may affect the complexity of isomorphisms between the corresponding copies of $\mathcal{M}(A, B)$.

Let $\mathcal{P}' = \langle\{A'_s\}_{s\in\omega}, \{B'_s\}_{s\in\omega}\rangle$ and $\mathcal{P}'' = \langle\{A''_s\}_{s\in\omega}, \{B''_s\}_{s\in\omega}\rangle$ be two enumerations of the sets A, B generating two computable presentations $\mathcal{M}'$ and $\mathcal{M}''$ of $\mathcal{M}(A, B)$, respectively. Suppose we try to build an isomorphism between $\mathcal{M}'$ and $\mathcal{M}''$. There is no problem constructing the isomorphism between the ω-chains formed by $\{x_i\}_{i\in\omega}$ in both copies. Now assume that a witness element connected to x_i has appeared in both copies and we want to extend the partial isomorphism we have built so far. If $i \notin B$, we know that these elements are the only elements connected to x_i's in the corresponding copies, and we can extend the isomorphism. Otherwise we may run into trouble. Suppose that we extended the isomorphism as above, assuming that the appeared elements are the a_i's in their copies of $\mathcal{M}(A, B)$. If $i \in B \setminus A$, then we are fine, as the elements b_i will appear later connected to the a_i's and we extend the isomorphism in the unique way. However, if $i \in A$, that is, if, in fact, two elements, a_i and c_i are connected to x_i, it may be the case that in one copy, say, $\mathcal{M}'$, the element b_i will be connected to the element we believe is a_i, and in the second copy it will be connected to the other element connected to x_i which has not yet appeared. So, at a later stage we will not be able to extend our partial isomorphism.

Consider the same example as above with $0 \in A$; $1 \notin B$; .... Figure 2 shows the above-mentioned trouble for $i = 0$:

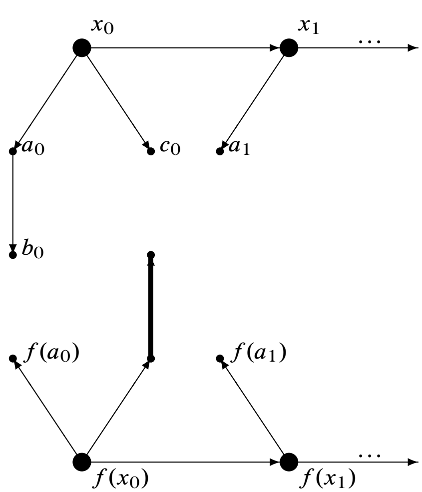

Figure 2 Example of a partial isomorphism that turns out to be incorrect.

In other words, to avoid the trouble, whenever i is enumerated into A, we first need to wait for the stage where i is enumerated into B and only then extend the isomorphism between the finite structures built up to this stage. With this idea in mind, we define a new function which will be useful for further reasoning.

For a pair of enumerations

$$\mathcal{P} = \big\langle \{A_s\}_{s\in\omega}, \{B_s\}_{s\in\omega} \big\rangle$$

of the sets $A \subseteq B$, define a function

$$g_{\mathcal{P}}(i) = (\mu s)\big[(\forall t > s)[i \in A_t \implies i \in B_t]\big].$$

It is not hard to see that

$$g_{\mathcal{P}} \equiv_T \big\{\langle i, s\rangle \;\big|\; (\forall t > s)[i \in A_t \implies i \in B_t]\big\},$$

which gives a $\forall$-definition of $g_{\mathcal{P}}$. Thus, $g_{\mathcal{P}}$ has a c.e. degree. Moreover, $g_{\mathcal{P}} \leq_T A, B$.

Proposition 2.4 *For arbitrary copies $\mathcal{M}'$ and $\mathcal{M}''$ of $\mathcal{M}(A, B)$, there exists a pair of computable enumerations*

$$\mathcal{P} = \big\langle \{A_s\}_{s\in\omega}, \{B_s\}_{s\in\omega} \big\rangle$$

of the sets A and B, such that the isomorphism $f : \mathcal{M}' \to \mathcal{M}''$ is computable relative to $g_{\mathcal{P}}$.

Proof Let $\mathcal{P}' = \langle \{A'_s\}_{s\in\omega}, \{B'_s\}_{s\in\omega}\rangle$ and $\mathcal{P}'' = \langle \{A''_s\}_{s\in\omega}, \{B''_s\}_{s\in\omega}\rangle$ be the pairs of computable enumerations of A and B that result from $\mathcal{M}'$ and $\mathcal{M}''$, respectively. Define $\mathcal{P} = \langle \{A_s\}_{s\in\omega}, \{B_s\}_{s\in\omega}\rangle$ as follows:

$$A_s = A'_s \cup A''_s,$$
$$B_s = B'_s \cap B''_s.$$

Now after the step $g_{\mathcal{P}}(i)$ we can be sure that if i has been enumerated into any of A'_s or A''_s, it already appeared in both B'_s, B''_s. Thus, after the step $g_{\mathcal{P}}(i)$ we know how to extend the isomorphism onto the elements connected to x_i in such a way that it will not be damaged at later stages. □

Proposition 2.5 *For an arbitrary pair of computable enumerations*

$$\mathcal{P} = \big\langle \{A_s\}_{s\in\omega}, \{B_s\}_{s\in\omega} \big\rangle$$

of c.e. sets A and B, where $A \subseteq B$, there exist computable copies $\mathcal{M}', \mathcal{M}''$ of the structure $\mathcal{M}(A, B)$ such that the function $g_{\mathcal{P}}$ is computable relative to $f : \mathcal{M}' \to \mathcal{M}''$ (in fact, $f \equiv_T g_{\mathcal{P}}$).

Proof Let $\mathcal{M}'$ be the copy of $\mathcal{M}(A, B)$ constructed as described after Proposition 2.2. We build an isomorphic copy $\mathcal{M}''$ in such a way that the isomorphism between the two presentations computes $g_{\mathcal{P}}$. The universe of $\mathcal{M}''$ also is

$$\{x_i : i \in \omega\} \cup \{a_i : i \in \omega\} \cup \{b_i : i \in B\} \cup \{c_i : i \in A\},$$

but the relation P is defined differently. As before, we declare P to be true on all the pairs

$$(x_i, x_{i+1}) \qquad (x_i, a_i) \qquad (x_i, c_i),$$

whenever the corresponding elements are in the domain of $\mathcal{M}''$. For $i \in B \setminus A$ we connect b_i to a_i, as before. But for $i \in A$ we connect b_i to a_i or to c_i depending on whether i was first enumerated into A or B: if i is first enumerated into A, then $P(c_i, b_i)$ holds in $\mathcal{M}''$, otherwise $P(a_i, b_i)$ holds in $\mathcal{M}''$ (note that this also includes the case $i \in B \setminus A$). For our example from above, assume that 0 is enumerated into A before it is enumerated into B, but 4 is first enumerated into B. Then the presentations $\mathcal{M}'$, $\mathcal{M}''$ as Figure 3 shows:

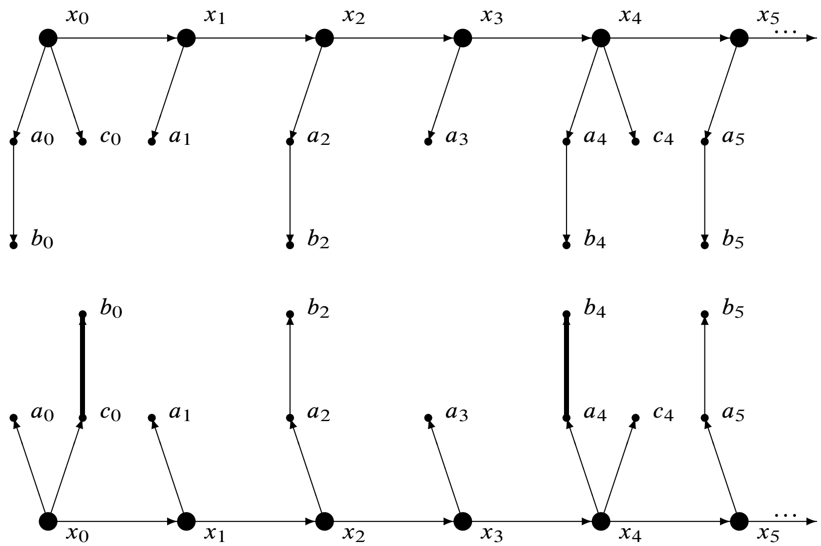

Figure 3 Structures $\mathcal{M}'$ and $\mathcal{M}''$ for $\mathcal{M}(A, B)$ from Figure 1.

Obviously, the structures $\mathcal{M}'$ and $\mathcal{M}''$ are isomorphic. Moreover, the isomorphism f determines whether an element was first enumerated into A or B, which is enough to compute the function $g_{\mathcal{P}}$. □

Proposition 2.6 *The ideal I generated by the degrees of isomorphisms between various computable copies of the structure $\mathcal{M}(A, B)$ may be generated by the degrees of the functions $g_{\mathcal{P}}$ for various pairs of computable enumerations $\mathcal{P} = \langle \{A_s\}_{s\in\omega}, \{B_s\}_{s\in\omega} \rangle$ of the sets A, B.*

Proof This follows directly from Propositions 2.4 and 2.5. □

Proposition 2.7 *The structure $\mathcal{M}(A, B)$ is $\mathbf{d}$-computably categorical if and only if for every pair of computable enumerations $\mathcal{P} = \langle \{A_s\}_{s\in\omega}, \{B_s\}_{s\in\omega} \rangle$ of the sets A, B, the function $g_{\mathcal{P}}$ is $\mathbf{d}$-computable.*

Proof This also follows directly from Propositions 2.4 and 2.5. □

Proposition 2.8 *Let*

$$\mathcal{P} = \left\langle \{A_s\}_{s\in\omega}, \{B_s\}_{s\in\omega} \right\rangle$$

and

$$\mathcal{P}' = \left\langle \{A'_s\}_{s\in\omega}, \{B'_s\}_{s\in\omega} \right\rangle$$

be pairs of computable enumerations of c.e. sets $A \subseteq B$. Then for the pair of computable enumerations

$$\mathcal{P}'' = \big\langle \{A_s \cup A'_s\}_{s\in\omega}, \{B_s \cap B'_s\}_{s\in\omega} \big\rangle,$$

the function $g_{\mathcal{P}''}$ computes both the functions $g_{\mathcal{P}}$ and $g_{\mathcal{P}'}$.

Proof Notice that the inclusion

$$(\forall t > s)[i \in A_t \cup A'_t \implies i \in B_t \cap B'_t]$$

directly implies both inclusions

$$(\forall t > s)[i \in A_t \implies i \in B_t]$$

and

$$(\forall t > s)[i \in A'_t \implies i \in B'_t].$$

Therefore, $g_{\mathcal{P}} \leq_T g_{\mathcal{P}''}$ and $g_{\mathcal{P}'} \leq_T g_{\mathcal{P}''}$. □

Proposition 2.9 *There exist c.e. sets $A \subseteq B$ such that the ideal I generated by the degrees of isomorphisms between various computable copies of the structure $\mathcal{M}(A, B)$ is not principal.*

Before we give a proof of this statement, we will prove a stronger fact.

Proposition 2.10 *There exist c.e. sets $A \subseteq B$ such that for every c.e. set W, where $W \leq_T A$, $W \leq_T B$, there exist a pair of computable enumerations*

$$\mathcal{P} = \big\langle \{A_s\}_{s\in\omega}, \{B_s\}_{s\in\omega} \big\rangle$$

of the sets A, B, such that $g_{\mathcal{P}} \not\leq_T W$.

Proof We will construct c.e. sets A and B satisfying the requirements below for all c.e. sets W and Turing operators Φ and Ψ:

$$R_{W,\Phi,\Psi,} : W = \Phi(A) = \Psi(B) \implies (\exists \mathcal{P})[g_{\mathcal{P}} \not\leq_T W],$$

where $\mathcal{P}$ is a pair of computable enumerations of the sets A and B.

For each triple $\mathcal{T} = (W, \Phi, \Psi)$, the requirement $R_{\mathcal{T}}$ will be met by constructing a computable enumeration

$$\mathcal{P} = \mathcal{P}^{\mathcal{T}} = \big\langle \{A^{\mathcal{T}}_s\}_{s\in\omega}, \{B^{\mathcal{T}}_s\}_{s\in\omega} \big\rangle$$

of the sets A and B satisfying the subrequirements

$$R_{\mathcal{T},\Theta} : W = \Phi(A) = \Psi(B) \implies g_{\mathcal{P}^{\mathcal{T}}} \neq \Theta(W)$$

for each Turing operator Θ.

In fact, for our purposes it is enough to define the enumeration $A^{\mathcal{T}}_s$ exactly as the enumeration A_s, but it will not be true for $B^{\mathcal{T}}_s$ and B_s. At each stage s, we will have

$$A_s = A^{\mathcal{T}}_s \subseteq B^{\mathcal{T}}_{s+1} \subseteq B_{s+1}.$$

Also, in the case of $W = \Phi(A) = \Psi(B)$ we should have the agreement

$$\bigcup_s B_s =_{\mathrm{dfn}} B = B^{\mathcal{T}} =_{\mathrm{dfn}} \bigcup_s B^{\mathcal{T}}_s$$

for $\mathcal{T} = (W, \Phi, \Psi)$. If $W \neq \Phi(A)$ or $W \neq \Psi(B)$, we do not care about a disagreement between B and $B^{\mathcal{T}}$.

In the construction below φ, ψ, and θ denote the *use*-functions for Turing operators Φ, Ψ, and Θ, respectively.

The strategy for a subrequirement $R_{\mathcal{T},\Theta}$, where $\mathcal{T} = (W, \Phi, \Psi)$:

1. Choose a sufficiently large witness x, not yet enumerated into A or B (and, therefore, not enumerated into $A_s^{\mathcal{T}}$ or $B_s^{\mathcal{T}}$).
2. Wait for a stage s such that $\Theta_s(W_s, x) = 0$ and
$$W_s \upharpoonright \theta_s(W_s; x) = \Phi_s(A_s) \upharpoonright \theta_s(W_s; x).$$
3. Set a priority restraint on enumeration into A of elements $a < \varphi_s(A_s; y)$ for all $y < \theta_s(W_s; x)$.
4. Enumerate x into B_{s+1}.
5. Temporarily stop the strategies for subrequirements $R_{\mathcal{T},\Theta'}$, $\Theta' \neq \Theta$.
6. Wait for a stage $t > s$ such that
$$W_s \upharpoonright \theta_s(W_s; x) = \Psi_t(B_t) \upharpoonright \theta_s(W_s; x).$$
7. Set a priority restraint on enumeration into B of elements $b < \psi_t(B_t; y)$ for all $y < \theta_s(W_s; x)$.
8. Enumerate x into $A_{t+1} = A_{t+1}^{\mathcal{T}}$ and into $B_{t+2}^{\mathcal{T}}$ (so that we have $g_{\mathcal{P}\mathcal{T}}(x) = t + 1 \neq 0$).
9. Resume the strategies for subrequirements $R_{\mathcal{T},\Theta'}$, $\Theta' \neq \Theta$.

End of strategy description.

Possible outcomes of the strategy for $R_{\mathcal{T},\Theta}$, $\mathcal{T} = (W, \Phi, \Psi)$**:**

A. The strategy gets stuck at 2. Then $W = \Phi(A)$ implies $\Theta(W; x) \neq 0 = g_{\mathcal{P}\mathcal{T}}(x)$.

B. The strategy gets stuck at 6. Then either $\Phi(A) \neq W$ or $\Psi(B) \neq W$. Strategies for $R_{\mathcal{T},\Theta'}$, $\Theta' \neq \Theta$ are not resumed at 9, but the restart is not needed since the whole requirement $R_{\mathcal{T}}$ is satisfied. Also, we have $x \in B - B^{\mathcal{T}}$, but an agreement between B and $B^{\mathcal{T}}$ is not needed for the same reason.

C. The strategy successfully finishes at 9. Then $W = \Psi(B)$ implies that $g_{\mathcal{P}\mathcal{T}}(x) \neq 0 = \Theta(W; x)$ since the W-use of the computation is preserved via B-restraints at 7.

Of course, the strategy above can be successful only if its restraints are not injured. Namely, if a restraint posed at 3 (A-restraint) or 7 (B-restraint) is injured, then we can have a simultaneous change in $W = \Phi(A) = \Psi(B)$ that causes $0 \neq g_{\mathcal{P}\mathcal{T}}(x) = \Theta(W; x)$ in the outcome 2

This conflict between different strategies can be solved by standard finite injury arguments. Whenever a restraint of a strategy becomes injured by a higher priority strategy, or whenever the execution of 4 (enumeration into B) or 8 (enumeration into A) is blocked due to a restraint of higher priority strategy, we should *initialize* the injured/blocked strategy.

The initialization of a strategy means

- enumeration of the old witness $x \in B$ into $B^{\mathcal{T}}$ (to avoid a disagreement between B and $B^{\mathcal{T}}$); and
- restart of the strategy with a new witness x.

To get the whole *construction* we need to

- fix a priority ω-ordering of all quadruples $\langle W, \Phi, \Psi, \Theta\rangle = \langle \mathcal{T}, \Theta\rangle$;
- assign to each $\langle W, \Phi, \Psi, \Theta\rangle = \langle \mathcal{T}, \Theta\rangle$ a strategy for the subrequirement $R_{\mathcal{T},\Theta}$;
- simultaneously run all the strategies.

The obtained sets A and B will satisfy the necessary conditions since each strategy acts at finitely many stages, and therefore makes only finitely many injuries to strategies of lower priority. □

Now we are ready to prove Proposition 2.9.

Proof of Proposition 2.9 Take the sets A, B constructed in Proposition 2.10 and build the corresponding $\mathcal{M}(A, B)$. By Proposition 2.4, for every isomorphism f between two copies of $\mathcal{M}(A, B)$, there exists a computable enumeration $\mathcal{Q} = \langle \{A_s\}_{s\in\omega}, \{B_s\}_{s\in\omega}\rangle$ of A, B such that $f \le_T g_{\mathcal{Q}}$. Recall that $g_{\mathcal{Q}} \le_T A, B$. Let W be the c.e. set

$$W = \big\{\langle x, s\rangle \ \big|\ (\exists t > s)[x \in A_t \& x \notin B_t]\big\} \equiv_T g_{\mathcal{Q}}.$$

For this W, there exists an enumeration $\mathcal{P}$ of A, B such that $g_{\mathcal{P}} \not\le_T W$ by Proposition 2.10. Then by Proposition 2.5, there exists an isomorphism $f' \ge_T g_{\mathcal{P}}$ between some computable copies of $\mathcal{M}(A, B)$. Then $f' \not\le_T W$; thus, $f' \not\le_T f$.

Similarly, for a finite set $f_1, \dots, f_n$ of isomorphisms between computable copies of $\mathcal{M}(A, B)$, find the corresponding pairs of enumerations $\mathcal{Q}_1, \dots, \mathcal{Q}_n$. Then by Proposition 2.8, there exists an enumeration $\mathcal{Q}$ such that $g_{\mathcal{Q}} \ge_T g_{\mathcal{Q}_1}, \dots, g_{\mathcal{Q}_n}$. For $\mathcal{Q}$, find $\mathcal{P}$ and f' as above by using Propositions 2.10 and 2.5. Then $f' \not\le_T f_1 \oplus \cdots \oplus f_n$; that is, the ideal is not principal. □

Note that the construction of the set B in the proof of Proposition 2.10 is compatible with the standard c.e. permitting method. Namely, let X be a c.e. noncomputable set, and let f be a computable function with its range equal to X. We can modify our strategy for the requirements $R_{\mathcal{T},\Theta}$ accepting several active witnesses in the following way.

Modified requirements $R_{\mathcal{T},\Theta}$:

1. If there is an active and *certified* witness $x > f(s)$, where s is a current stage, then immediately go to 4 below. Otherwise, choose a new active (sufficiently large) witness z, not yet enumerated into A and B (and, therefore, into the enumerations $A_s^{\mathcal{T}}$ and $B_s^{\mathcal{T}}$).
2. Wait for a stage s such that either $x > f(s)$ for some active certified witness x, or $\Theta_s(W_s, z) = 0$ and
$$W_s \upharpoonright \theta_s(W_s; z) = \Phi_s(A_s) \upharpoonright \theta_s(W_s; z).$$
In the former case, we immediately go to 4 below. In the latter case, the active witness becomes *certified*, and we immediately go to 3.
3. Set a priority restraint on enumeration into A of elements $a < \varphi_s(A_s; y)$ for all $y < \theta_s(W_s; z)$. Return to 1.
4. Enumerate x into B_{s+1} ....

(Steps 4–9 are absolutely the same as before.)

Note that if we had infinitely many certified witnesses for a single strategy, then

$$a \notin X \iff (\exists s)(\exists x > a)\big[a \notin \{f(0), \dots, f(s)\}\&x \text{ is certified at } s\big],$$

contradicting the noncomputability of X. Hence, each strategy certifies only finitely many witnesses, so that the total restraint posed on 3 is finite.

The modified strategy produces the set B with the property

$$x \notin B \iff (\exists s)\big[x \notin B_s \& X \upharpoonright x \subseteq \{f(0), \dots, f(s)\}\big],$$

and therefore we have $B \leq_T X$. Thus, we have proved the following.

Proposition 2.11 *For every c.e. noncomputable set X there exist c.e. sets $A \subseteq B$ such that $B \leq_T X$, and for every c.e. set W, where $W \leq_T A$, $W \leq_T B$, there exist a pair of computable enumerations*

$$\mathcal{P} = \big\langle \{A_s\}_{s\in\omega}, \{B_s\}_{s\in\omega} \big\rangle$$

of the sets A, B such that $g_{\mathcal{P}} \not\leq_T W$.

Using Proposition 2.11 instead of Proposition 2.10, we get a "permitting" version of Proposition 2.9.

Proposition 2.12 *For every c.e. noncomputable set X there exist c.e. sets $A \subseteq B$ such that $B \leq_T X$, and the ideal I generated by the degrees of isomorphisms between various computable copies of the structure $\mathcal{M}(A, B)$ is not principal.*

From Propositions 2.2 and 2.9, we immediately get the main result of the paper.

Theorem 2.13 *There exists a rigid computable structure with no degree of categoricity.*

Using Proposition 2.12 instead of Proposition 2.9, we get an even stronger result.

Theorem 2.14 *For every c.e. nonzero degree $\mathbf{x}$, there exists an $\mathbf{x}$-computably categorical rigid computable structure with no degree of categoricity.*

3 Open Problems

Even though the notion of computable categoricity appeared in the very beginning of computable model theory, the study of categoricity spectra and degrees of categoricity is a relatively new topic. A number of questions remain unsolved. Here we mention just a couple of them. More can be found in [2] and [3].

Question 3.1 *Can a union of two cones be the categoricity spectrum of a computable structure?*

Question 3.2 *Can a d-c.e. degree be a degree of categoricity of a rigid structure?*

Recall that the known examples are rigid for c.e. degrees of categoricity but not rigid for properly d-c.e. degrees of categoricity.

References

[1] Anderson, B. A., and B. F. Csima, "Degrees that are not degrees of categoricity," to appera in Notre Dame Journal of Formal Logic, preprint, arXiv:1210.4220v2 [math.LO]. 47

[2] Csima, B. F., J. N. Y. Franklin, and R. A. Shore, "Degrees of categoricity and the hyperarithmetic hierarchy," *Notre Dame Journal of Formal Logic*, vol. 54 (2013), pp. 215–31. Zbl pre06157722. MR 3028796. DOI 10.1215/00294527-1960479. 46, 47, 55

[3] Fokina, E. B., I. Kalimullin, and R. Miller, "Degrees of categoricity of computable structures," *Archive for Mathematical Logic*, vol. 49 (2010), pp. 51–67. Zbl 1184.03026. MR 2592045. DOI 10.1007/s00153-009-0160-4. 46, 47, 55

[4] Fröhlich, A., and J. C. Shepherdson, "Effective procedures in field theory," *Philosophical Transactions of the Royal Society of London Series A*, vol. 248 (1956), pp. 407–32. Zbl 0070.03502. MR 0074349. 46

[5] Goncharov, S. S., "Autostable models and algorithmic dimensions," pp. 261–87 in *Handbook of Recursive Mathematics, Vol. 1*, vol. 138 of *Studies in Logic and the Foundations of Mathematics*, North-Holland, Amsterdam, 1998. Zbl 0958.03030. MR 1673641. DOI 10.1016/S0049-237X(98)80007-4. 46

[6] Hirschfeldt, D. R., B. Khoussainov, R. A. Shore, and A. M. Slinko, "Degree spectra and computable dimensions in algebraic structures," *Annals of Pure and Applied Logic*, vol. 115 (2002), pp. 71–113. Zbl 1016.03034. MR 1897023. DOI 10.1016/S0168-0072(01)00087-2. 46

[7] Mal'cev, A. I., "Constructive algebras, I," *Uspekhi Mathematicheskikh Nauk*, vol. 16 (1961), pp. 3–60. Zbl 0129.25903. MR 0151377. 46

[8] Mal'cev, A. I., "On recursive Abelian groups," *Doklady Akademiya Nauk SSSR*, vol. 146 (1962), pp. 1009–12. Zbl 0156.01105. MR 0151378. 46

[9] Miller, R., "d-computable categoricity for algebraic fields," *Journal of Symbolic Logic*, vol. 74 (2009), pp. 1325–51. Zbl 1202.03044. MR 2583823. DOI 10.2178/jsl/1254748694. 47

[10] Richter, L. J., "Degrees of structures," *Journal of Symbolic Logic*, vol. 46 (1981), pp. 723–31. Zbl 0512.03024. MR 0641486. DOI 10.2307/2273222. 46

[11] Rodgers, H., *Theory of Recursive Functions and Effective Computability*, Massachusetts Institute of Technology Press, Cambridge, Mass., 1987. MR 0886890. 47

Acknowledgments

The first author was supported by Austrian Science Fund (FWF) project V206-N13. The second author was partially supported by Russian Federation for Basic Research grant RFBR-15-01-08252 and by a subsidy allocated to Kazan Federal University for the state assignment in the sphere of scientific activities project no. 1.2045.2014. The third author was partially supported by Russian Federation for Basic Research grants RFBR-15-31-20607 and RFBR-15-41-02507.

Fokina
Institute of Discrete Mathematics and Geometry
University of Vienna of Technology
Vienna 1040
Austria
ekaterina.fokina@tuwien.ac.at

Frolov
Department of Mathematics
Kazan (Volga Region) Federal University
Kazan 420008
Russia
Andrey.Frolov@kpfu.ru

Kalimullin
Institute of Mechanics and Mathematics
Kazan (Volga Region) Federal University
Kazan 420008
Russia
Iskander.Kalimullin@kpfu.ru