STRENGTH AND WEAKNESS IN COMPUTABLE STRUCTURE THEORY

JOHANNA N.Y. FRANKLIN

Abstract. We survey the current results about degrees of categoricity and the degrees that are low for isomorphism as well as the proof techniques used in the constructions of elements of each of these classes. We conclude with an analysis of these classes, what we may deduce about them given the sorts of proof techniques used in each case, and a discussion of future lines of inquiry.

1. Introduction

The question of whether a computable isomorphism between two computable structures exists was first discussed in computable model theory sixty years ago [17]. Later, this question was generalized to the question of whether an isomorphism of a given Turing degree exists between two computable structures. There has been a great deal of recent work on Turing degrees that have been shown to be very strong with respect to computing isomorphisms between structures and those that have been shown to be very weak. The first such class of degrees is called the *degrees of categoricity*; degrees in the second such class are called *low for isomorphism*. Both of these classes of degrees have proven to be difficult to characterize completely; in fact, no full characterization exists for either class. In this paper, we will synthesize the work on these topics and the proof techniques involved, present some open questions, and discuss possible approaches to the subject.

1.1. Terminology.

We begin with a discussion of the most relevant definitions; other terms will be defined as necessary throughout the paper. We assume the reader is familiar with computability theory in general and computable structure theory in particular; [29, 30, 34] and [18] are useful references for these subjects, respectively. We will use the notation from Ash and Knight [2] when we discuss the hyperarithmetic hierarchy (as do the authors of all the papers concerning this hierarchy that we survey), and we suggest [31] as a general reference.

The most fundamental concept in this paper is that of an isomorphism relative to a particular Turing degree $\mathbf{d}$.

Definition 1.1. Given a Turing degree $\mathbf{d}$ and computable structures $\mathcal{A}$ and $\mathcal{B}$, we say that $\mathcal{A}$ is $\mathbf{d}$-*computably isomorphic to* $\mathcal{B}$ (which we will write $\mathcal{A} \cong_{\mathbf{d}} \mathcal{B}$) if there is an isomorphism between $\mathcal{A}$ and $\mathcal{B}$ that is computable from $\mathbf{d}$. If $\mathbf{d} = \mathbf{0}$, we say that $\mathcal{A}$ is *computably isomorphic to* $\mathcal{B}$ and write $\mathcal{A} \cong_{\Delta^0_1} \mathcal{B}$.

This idea is then used to define the concept of computable categoricity relative to a given Turing degree $\mathbf{d}$.

Definition 1.2. A computable structure $\mathcal{A}$ is $\mathbf{d}$-*computably categorical* if, for every computable structure $\mathcal{B}$ that is classically isomorphic to $\mathcal{A}$, we have $\mathcal{A} \cong_{\mathbf{d}} \mathcal{B}$.

Date: July 7, 2016.
The author would like to thank the editors of this Festschrift for the invitation to contribute.

Now we can define the central concepts in this paper: degrees of categoricity and lowness for isomorphism.

Definition 1.3. [11] A Turing degree $\mathbf{d}$ is a *degree of categoricity* if there is a computable structure $\mathcal{A}$ such that $\mathcal{A}$ is $\mathbf{c}$-computably categorical if and only if $\mathbf{c} \geq_T \mathbf{d}$. This degree $\mathbf{d}$ is furthermore a *strong degree of categoricity* if there is a computable structure $\mathcal{A}$ with computable copies $\mathcal{A}_1$ and $\mathcal{A}_2$ such that $\mathcal{A}$ has degree of categoricity $\mathbf{d}$ and every isomorphism from $\mathcal{A}_1$ to $\mathcal{A}_2$ computes $\mathbf{d}$.

In short, a degree is a degree of categoricity if it is the least degree that, for some computable structure, can compute an isomorphism from that structure to any classically isomorphic computable copy of itself. This means that we can think of it as calibrating the complexity of that computable structure in some way. Furthermore, a degree is a strong degree of categoricity if it not only computes such isomorphisms but can be computed by any isomorphism from one copy of a particular computable structure to another. We can thus say that a (strong) degree of categoricity is, in some way, a very strong degree: it is guaranteed to have a certain level of computational power for some computable structure.

On the other hand, a degree that is low for isomorphism is a degree that is very weak indeed for any pair of computable structures:

Definition 1.4. [14] A Turing degree $\mathbf{d}$ is *low for isomorphism* if for every pair of computable structures $\mathcal{A}$ and $\mathcal{B}$, $\mathcal{A} \cong_{\mathbf{d}} \mathcal{B}$ if and only if $\mathcal{A} \cong_{\Delta^0_1} \mathcal{B}$.

The word *low* is used in this definition as it has been used in computability theory since the 1970s: a degree $\mathbf{d}$ is generally called low for a relativizable class $\mathcal{C}$ if, when it is used as an oracle, the new, relativized class is no different than the original, unrelativized one (that is, when $\mathcal{C}^D = \mathcal{C}$ for $D \in \mathbf{d}$). This notion, first used in computability theory by Soare in [33], has appeared in almost every context in computability theory: degree theory [33], learning theory [32], and, more recently, algorithmic randomness [7, 13, 27]. Franklin and Solomon's paper introduced this concept into computable structure theory for the first time [14].

These notions appear to be entirely incompatible. Nontrivial degrees of categoricity possess some additional information required to compute an isomorphism for some structure, while degrees that are low for isomorphism have none. Clearly, the only degree that satisfies both of these conditions is $\mathbf{0}$.

At this point, there are several natural questions to ask. What sorts of closure do these classes possess? It is clear from the definition that the degrees that are low for isomorphism are closed downwards, but do they form an ideal? Are these degrees compatible or incompatible with natural classes of degrees, such as the hyperimmune-free degrees, minimal degrees, or low degrees?

Examples of degrees of categoricity and degrees that are low for isomorphism have been found, but a full characterization has been elusive for each. In this paper, we hope to present some of the constructions of these degrees and to analyze these constructions as well as to present some more general metainformation about both kinds and consider reasons that each type of degree is so difficult to characterize. We discuss degrees of categoricity in Section 2 and degrees that are low for isomorphism in Section 3, and we include an analysis of these classes in Section 4.

2. Degrees of categoricity

As mentioned, the concept of a degree of categoricity was first defined by Fokina, Kalimullin, and R. Miller in [11]. In this paper, they demonstrated that certain degrees were degrees of categoricity, showed that there were only countably many strong degrees of categoricity, and considered the question of degrees of categoricity for particular classes of structures. Csima, Franklin, and Shore then extended their results through the hyperarithmetic hierarchy and proved that there were only countably many degrees of categoricity [4] and, more recently, Csima and Harrison-Trainor showed that the degrees of categoricity of "natural" structures are very limited indeed [5].

2.1. **Examples of degrees of categoricity.** All of the results in this section are centered around the Ershov hierarchy [8, 9, 10]. Fokina, Kalimullin, and R. Miller's primary results can be stated as the following theorem:

Theorem 2.1. [11] *If* $\mathbf{d}$ *is a Turing degree that is c.e. or d.c.e. in* $\mathbf{0}^{(m)}$ *and* $\mathbf{0}^{(m)} \leq_T \mathbf{d}$ *for some* $m \in \omega$*, then* $\mathbf{d}$ *is a strong degree of categoricity. Furthermore,* $\mathbf{0}^{(\omega)}$ *is a strong degree of categoricity.*

We outline their constructions in increasing order of complexity. They begin by simply showing that a c.e. degree $\mathbf{d}$ is a degree of categoricity. To do so, they fix a c.e. Turing degree $\mathbf{d}$, a c.e. set W_e inside it, and a computable injective function h with range W_e. From this, they construct a structure $\mathcal{B}$ witnessing that $\mathbf{d}$ is a degree of categoricity.

$\mathcal{B}$ is a directed graph with two constant elements, c and d, and is constructed as follows. Four elements, α, β, γ, and δ, are dedicated to be the "origin" nodes, and the sequences $x_0, x_1, \ldots, x_i, \ldots$ and $y_0, y_1, \ldots, y_i, \ldots$ are the "target" nodes. (There is also a set of "witness" nodes $\{u_i\}_{i\in\omega}$, but those do not appear as elements of the graph and we will ignore them in this sketch.) We declare that $c^{\mathcal{B}} = \gamma$ and $d^{\mathcal{B}} = \delta$. At stage 0, $\mathcal{B}$ only has edges from β to every y_i and from δ to every x_i.

When a number i enters W_e, we add the following additional edges to our graph:

- an edge from α to x_i,
- an edge from β to x_i,
- an edge from γ to y_i, and
- an edge from δ to y_i.

At the end of our construction, we see that if $i \in W_e$, then there are edges from α, β, and δ to x_i and edges from γ, δ, and β to y_i. Therefore, any automorphism of $\mathcal{B}$ that swaps x_i and y_i must swap α and γ, and it may either fix β and δ or swap them. Furthermore, if $i \notin W_e$, then the only edge to x_i comes from δ and the only edge to y_i comes from β. This means that if an automorphism of $\mathcal{B}$ swaps x_i and y_i, then it must swap β and δ as well but its behavior on α and γ does not matter.

We first argue that $\mathbf{d}$ is a degree of categoricity for $\mathcal{B}$. Suppose we have another computable structure $\mathcal{A}$ that is classically isomorphic to $\mathcal{B}$ and we wish to build an isomorphism g from $\mathcal{B}$ to $\mathcal{A}$ computable in $\mathbf{d}$. We first note that we can identify $g(\gamma)$ as $c^{\mathcal{A}}$ and $g(\delta)$ as $d^{\mathcal{A}}$. Since we defined the x_is and y_is as sequences, we can identify the pair $g(x_i)$ and $g(y_i)$ for each i. Now we use W_e as our oracle to determine which is which: if $i \in W_e$, then the element that is connected to $c^{\mathcal{A}}$ is $g(y_i)$; if $i \notin W_e$, then the element that is connected to $d^{\mathcal{A}}$ is $g(x_i)$.

Furthermore, we can prove that $\mathbf{d}$ is a strong degree of categoricity for $\mathcal{B}$ by exhibiting a structure $\mathcal{A}$ such that any isomorphism between $\mathcal{B}$ and $\mathcal{A}$ can compute W_e. We define $\mathcal{A}$ in this case to be identical to $\mathcal{B}$ save for the choice of constants: we set $c^{\mathcal{A}} = \widehat{\alpha}$ and $d^{\mathcal{A}} = \widehat{\delta}$ (where $\widehat{\alpha}$ and $\widehat{\delta}$ are the $\mathcal{A}$-equivalents of α and δ). There is exactly one isomorphism from $\mathcal{B}$ to $\mathcal{A}$: the isomorphism that swaps the x_is that are connected to α with the corresponding y_is that are connected to γ and preserves the rest. Knowledge of this isomorphism clearly allows one to determine W_e.

The proof that every d.c.e. degree is a degree of categoricity is slightly more complicated. Again, we fix a d.c.e. degree $\mathbf{d}$ and a d.c.e. set $A-B$ in $\mathbf{d}$, where A and B are c.e. sets such that $B \subseteq A$. We construct a directed graph once more. However, this time we will have a single sequence $x_0, x_1, \ldots, x_i, \ldots$ where the i^{th} element is connected to the four points a_i, b_i, c_i, and d_i and these four points form a square at stage 0: there are arrows from a_i to b_i, b_i to c_i, c_i to d_i, and d_i to a_i.

We now choose our witnessing computable structure $\mathcal{B}$ to be the substructure of the directed graph above with all the x_is, c_is, and d_is, but only the a_is for $i \in A$ and only the b_is for $i \in B$. Since $\mathcal{B}$'s universe is c.e., we can proceed as though $\mathcal{B}$ is computable.

Now suppose that $\mathcal{A}$ is a computable structure that is isomorphic to $\mathcal{B}$. To compute an isomorphism g from $\mathcal{B}$ to $\mathcal{A}$, we first identify $g(x_i)$ for each i. If $i \in D$, then there is no b_i, and we can uniquely define $g(a_i)$, $g(b_i)$, and $g(c_i)$. If $i \notin D$, then either we have to identify $g(a_i)$, $g(b_i)$, $g(c_i)$, and $g(d_i)$ (if i never entered D) or only $g(c_i)$, and $g(d_i)$ (if i entered and then exited D). In any case, we define $g_(c_i)$, and $g(d_i)$ as soon as we find two elements that are candidates for them; if we later determine that $i \in A$ and therefore $i \in B$ as well, we can extend the isomorphism appropriately.

To prove that $\mathbf{d}$ is actually a strong degree of categoricity, we show that an isomorphism exists between the structure previously described and the structure $\mathcal{A}$, where $\mathcal{A}$ is the substructure of the original directed graph with all the x_is, c_is, and d_i,s but only the a_is for $i \in B$ and only the b_is for $i \in A$. Suppose we have such an isomorphism. Then, for each $i \in \omega$, we can see that $i \in D$ if and only if $i \notin B$ and $f(c_i) \neq c_i$.

Both of the results above can be seen to relativize to degrees c.e. and d.c.e. in and above $\mathbf{0}^{(m)}$ for any $m \in \omega$ using Marker's construction from [24], so to fully prove Theorem 2.1, we only need show that there is a computable structure whose degree of categoricity is $\mathbf{0}^{(\omega)}$. As is logical for a limit case, this structure is simply the cardinal sum of the computable structures constructed to show that the degrees $\mathbf{0}^{(n)}$ are degrees of categoricity for all $n \in \omega$.

Fokina, Kalimullin, and R. Miller's paper did not treat the 3-c.e. case, which remains unsolved to this writing. Let us discuss briefly why it is far more difficult than the d.c.e. case. In the d.c.e. case, there are three possible scenarios for each $i \in \omega$: i is in D, which means that all of a_i, c_i, and d_i are in the first structure; i never entered D, which means that a_i, b_i, c_i, and d_i are in the structure, or i entered and then exited D, which means that only c_i and d_i are in the structure. Suppose that, in addition to these scenarios, we also had to deal with the case in which i had entered, exited, and then entered D again. This would mean that there would have to be two subcases for each of $i \in D$ and $i \notin D$. So far, no way has been found to code information into a structure in such a way that the first and third "versions" can be made isomorphic (the cases where $i \notin D$), the second and fourth "versions" can also be made isomorphic (the cases where $i \in D$), and we can transition from each version to the next in a computable way.

One of the questions asked in [11] was whether or not their construction could be extended to higher hyperarithmetic degrees. This was answered by Csima, Franklin, and Shore in [4], where they proved the following result.

Theorem 2.2. *If α is a computable ordinal, then $\mathbf{0}^{(\alpha)}$ is a strong degree of categoricity. If, in addition, α is a successor ordinal, then every degree that is c.e. or d.c.e. in and above $\mathbf{0}^{(\alpha)}$ is a strong degree of categoricity.*

Once again, these constructions use directed graphs. The authors use Hirschfeldt and White's "back-and-forth trees" in their construction [22], which are computable subtrees of $\omega^{<\omega}$ with no infinite paths. We outline their construction below.

They fix a system of notation for ordinals as follows: 1 denotes the ordinal 0, 2^a denotes the ordinal $\alpha+1$ when a denotes α, and $3{\cdot}5^e$ denotes a limit ordinal λ under certain technical conditions, including the totality of φ_e. This makes it possible to define two structures, $\mathcal{A}_a$ and $\mathcal{E}_a$, for each notation a using transfinite recursion. $\mathcal{A}_1$ is a single node, and $\mathcal{E}_1$ is a single root node that has infinitely many children, all of which are childless. If a represents the successor of a successor ordinal represented by b (so $a = 2^b$), then $\mathcal{A}_a$ consists of a single root node with infinitely many copies of $\mathcal{E}_b$ attached, and $\mathcal{E}_a$ consists of a single root node with infinitely many copies of both $\mathcal{A}_b$ and $\mathcal{E}_b$ attached. Finally, if a is the successor of a limit ordinal coded by e (so $a = 2^{3\cdot5^e}$), we must first define auxiliary trees $\mathcal{L}_{e,k}$ for every $k \in \omega$ as well as a structure $\mathcal{L}_{e,\infty}$ as follows:

- $\mathcal{L}_{e,k}$ consists of exactly one copy of $\mathcal{A}_{\varphi_e(n)}$ for all $n \leq k$ and exactly one copy of $\mathcal{E}_{\varphi_e(n)}$ for all $n > k$, and
- $\mathcal{L}_{e,\infty}$ consists of exactly one copy of $\mathcal{A}_{\varphi_e(n)}$ for every $n \in \omega$.

Now we can define $\mathcal{A}_a$ to consist of a root node with infinitely many copies of $\mathcal{L}_{e,k}$ for every $k \in \omega$ and $\mathcal{E}_a$ to consist of a root node with infinitely many copies of $\mathcal{L}_{e,k}$ for each $k \in \omega$ and infinitely many copies of $\mathcal{L}_{e,\infty}$. These procedures will always give us a computable tree.

We note that $\mathcal{A}_a$ can always be converted to $\mathcal{E}_a$ just by adding infinitely copies of either the appropriate $\mathcal{E}_b$ or the appropriate $\mathcal{L}_{e,\infty}$. This will be essential for our construction.

Now, in preparation for building the structures that witness the existence of the degrees of categoricity previously mentioned, we make note of several technical facts about these structures. A lemma from [22] allows us to see that, given an ordinal α and a Σ_α predicate P, for every notation a for α, there is a sequence of trees $\mathcal{T}_n$ that is uniformly computable from a and a Σ_α index for P such that for all n, $\mathcal{T}_n$ is isomorphic to one of $\mathcal{E}_a$, $\mathcal{A}_a$, $\mathcal{L}_{e,k}$, or $\mathcal{L}_{e,\infty}$ depending on whether $P(n)$ holds and whether α is a successor or limit ordinal. We can also define the rank of a back-and-forth limb of a tree ($\mathcal{S}$ is a limb of $\mathcal{T}$ if $\mathcal{S} \subseteq \mathcal{T}$ and is closed under the "child" relation within $\mathcal{T}$, and $\mathcal{S}$ is a back-and-forth limb if it is isomorphic to one of our back-and-forth trees) and then use this rank to associate a natural complexity with a back-and-forth tree based on its isomorphism type. This will let us prove that $\mathbf{0}^{(\alpha)}$ can compute an isomorphism between two back-and-forth limbs of different computable trees as long as both limbs have rank less than a (the notation for α) and are classically isomorphic, and this computation is uniform in the roots for the limbs.

Now we can prove that for every α, there is a computable structure $\mathcal{S}_a$ with strong degree of categoricity $\mathbf{0}^{(\alpha)}$. For each notation, we construct a "standard" copy of $\mathcal{S}_a$ and a "hard" copy $\widehat{\mathcal{S}}_a$. All of these structures consist of infinitely many disjoint copies of these back-and-forth trees. We present the case for $a = 2$ (the notation for $\mathbf{0}'$) here and then describe the other cases briefly.

The "standard" copy, $\mathcal{S}_2$, will consist of infinitely many disjoint copies of $\mathcal{A}_1$ and $\mathcal{E}_1$. The set of edges in this copy is $\{(\langle 2n, 0\rangle, \langle 2n, k\rangle) \mid k > 0\}$. The elements in the odd columns are not connected to any other elements and are thus each isomorphic to $\mathcal{A}_1$; each even column is isomorphic to $\mathcal{E}_1$ with $\langle 2n, 0\rangle$ as the root node.

Now we use an approximation $\{K_s\}_{s\in\omega}$ to $0'$ to build the "hard" copy $\widehat{\mathcal{S}}_2$. The set of edges of this copy is defined to be $\{(\langle 2n, 0\rangle, \langle 2n, t\rangle) \mid n \in K_t\}$. In this case, if $n \in 0'$, a subset of the n^{th} even column will be isomorphic to $\mathcal{E}_1$, and all of the other elements will form substructures isomorphic to $\mathcal{A}_1$. Clearly, if one is given an isomorphism between the "standard" and "hard" copies, $0'$ can be computed by determining which of the odd columns contain copies of $\mathcal{A}_1$, and if one is given any two computable copies of $\mathcal{S}_2$, then the only questions that need to be answered to compute an isomorphism between them are Σ^0_1 and Π^0_1, which $\mathbf{0}'$ can answer.

For an ordinal β that is the successor of a successor ordinal α with notation a, our structure will consist of infinitely many disjoint copies of $\mathcal{A}_a$ and $\mathcal{E}_a$. The "standard" copy will code the $\mathcal{E}_a$s in the even columns and the $\mathcal{A}_a$s in the odd columns; the "hard" copy will code the $\mathcal{E}_a$s in the columns corresponding to those n in the jump of $\mathbf{0}^{(\alpha)}$ and the $\mathcal{A}_a$s in the other columns. The basic argument is the one given above, though with more bookkeeping: we must show that the root nodes of all the connected components of each structure and the back-and-forth indices of their limbs can be computed in this jump. This allows us to define a bijection between the root nodes in each structure that preserves back-and-forth indices, which is all we need to compute an isomorphism between the standard copy and an arbitrary computable copy.

For an ordinal α that is a limit ordinal coded by e, we construct the "standard" copy by coding a copy of $\mathcal{E}_{\varphi_e(n)}$ in the $\langle k, n\rangle^{th}$ column if k is even and a copy of $\mathcal{A}_{\varphi_e(n)}$ in the $\langle k, n\rangle^{th}$ column if k is odd. In the "hard" copy, we determine where to code copies of $\mathcal{E}_{\varphi_e(n)}$ and $\mathcal{A}_{\varphi_e(n)}$ depending on whether n is in $\mathbf{0}^{(\alpha)}$.

Finally, for an ordinal that is the successor of a limit ordinal coded by e, our structure will consist of infinitely many disjoint copies of $\mathcal{L}_{e,\infty}$ and $\mathcal{L}_{e,k}$ for all $k \in \omega$. We construct the "standard" copy by coding a copy of $\mathcal{L}_{e,k}$ in the $\langle n, k, 0\rangle^{th}$ column if n is even and a copy of $\mathcal{L}_{e,\infty}$ in it otherwise. The fact that we can compute a sequence of trees of the form $\mathcal{E}_a$, $\mathcal{A}_a$, $\mathcal{L}_{e,k}$, and $\mathcal{L}_{e,\infty}$ as previously described lets us construct a "hard" copy that codes information about our ordinal. Since all the connected components are back-and-forth trees with rank below the ordinal we are considering, the corresponding Turing degree is enough to compute an isomorphism between these copies, and we can argue as before that it is enough to compute an isomorphism between any two computable copies of this structure.

We now move from the c.e. case to the d.c.e. case and argue that any degree $\mathbf{d}$ d.c.e. in and above $\mathbf{0}^{(\alpha)}$ for a computable successor ordinal α must be a strong degree of categoricity. Once again, two different structures, $\mathcal{G}$ and $\widehat{\mathcal{G}}$ are constructed, the former the "standard" copy and the latter the "hard" copy. Let $D \in \mathbf{d}$ witness that $\mathbf{d}$ is d.c.e. in and above $\mathbf{0}^{(\alpha)}$. We will use the same general technique as in [11]: information is coded into 4-cycles containing the nodes a, b, c, and d based on whether n enters D and then leaves it, enters and never leaves, or never enters it at all. This information is coded by attaching either $\mathcal{A}_\alpha$s or $\mathcal{E}_\alpha$s to each of the nodes in the 4-cycle. The nodes a and c are treated identically in $\mathcal{G}$ and $\widehat{\mathcal{G}}$, but the roles of b and d are swapped, and the choices of $\mathcal{A}_\alpha$ and $\mathcal{E}_\alpha$ are made in such a way to ensure that if n is in our set, we can create an isomorphism regardless of the way in which it entered. Furthermore, to ensure that any isomorphism between these structures can compute $\mathbf{d}$, we add a 3-cycle to each of $\mathcal{G}$ and $\widehat{\mathcal{G}}$. In $\mathcal{G}$, each node in this 3-cycle will have a copy of the "standard" structure we built previously attached to it, and $\widehat{\mathcal{G}}$ will have a copy of the "hard" structure we built previously attached to it.

The other primary example of degrees of categoricity to date comes from Csima and Harrison-Trainor [5]. In this paper, they consider computable structures on cones in the Turing degrees. They begin by defining a relativized version of degrees of categoricity:

Definition 2.3. [5] A structure $\mathcal{A}$ has *degree of categoricity* $\mathbf{d}$ *relative to* $\mathbf{c}$ if $\mathbf{d}$ is the least degree that can compute an isomorphism between any two $\mathbf{c}$-computable copies of $\mathcal{A}$. If there are also two

$\mathbf{c}$-computable copies of $\mathcal{A}$ such that for every isomorphism f between them, $f \oplus \mathbf{c} \geq_T \mathbf{d}$, then $\mathcal{A}$ has *strong degree of categoricity* $\mathbf{d}$ *relative to* $\mathbf{c}$.

Definition 2.4. A structure $\mathcal{A}$ has a *(strong) degree of categoricity on a cone* if there is some $\mathbf{d}$ such that for every $\mathbf{c} \geq_T \mathbf{d}$, $\mathcal{A}$ has a (strong) degree of categoricity relative to $\mathbf{c}$. Furthermore, we say that a structure $\mathcal{A}$ has a *(strong) degree of categoricity* $\mathbf{0}^{(\alpha)}$ *on a cone* if there is some $\mathbf{d}$ such that for every $\mathbf{c} \geq_T \mathbf{d}$, $\mathcal{A}$ has a (strong) degree of categoricity $\mathbf{c}^{(\alpha)}$ relative to $\mathbf{c}$.

Their main theorem is as follows:

Theorem 2.5. [5] *Suppose that $\mathcal{A}$ is a computable structure. Then on a cone, $\mathcal{A}$ has a strong degree of categoricity, and this degree is $\mathbf{0}^{(\alpha)}$, where α is the least computable ordinal such that $\mathcal{A}$ is $\mathbf{0}^{(\alpha)}$-computably categorical on a cone.*

The general proof of this theorem involves a version of Ash's metatheorem [2]: Montalbán recently developed a variant on it for successor ordinals [25], and Csima and Harrison-Trainor expanded his variant to include limit ordinals. Here we will only sketch their proof for structures that are $\mathbf{0}'$-computably categorical on a cone due to the complexity of the general proof.

We begin by supposing that $\mathcal{A}$ is not computably categorical on any cone and choose a degree $\mathbf{e}$ that can compute $\mathcal{A}$ and a Scott family for $\mathcal{A}$ with certain properties and that satisfies some technical conditions. We let $\mathbf{d} \geq_T \mathbf{e}$, and then we choose $\mathbf{c}$ to be c.e. in and above $\mathbf{d}$ and choose a $C \in \mathbf{c}$ and take a $\mathbf{d}$-computable approximation to it. This allows us to build our $\mathcal{B}$ and a sequence $\langle f_s \rangle$ of partial isomorphisms computably in $\mathbf{d}$. Thus, the limit of the partial isomorphisms, f, will be a C-computable isomorphism between $\mathcal{B}$ and $\mathcal{A}$. This means that $\mathbf{c}$ will compute an isomorphism between $\mathcal{B}$ and $\mathcal{A}$, and we can further use $g \oplus d$ to compute $\mathbf{c}$ for every isomorphism f between $\mathcal{A}$ and $\mathcal{B}$.

We then use Knight's theorem on the upwards closure of degree spectra from [23] to show that every isomorphism between $\mathcal{B}$ and $\mathcal{A}$ computes $\mathbf{c}$ instead of simply that $g \oplus d$ computes $\mathbf{c}$ for every isomorphism f between $\mathcal{A}$ and $\mathcal{B}$. This lets us see that a structure cannot have a degree of categoricity properly between $\mathbf{0}$ and $\mathbf{0}'$ on a cone.

Csima and Harrison-Trainor also prove the following:

Theorem 2.6. [5] *Suppose $\mathcal{A}$ is a countable structure. Then, on a cone, if $\mathcal{A}$ is Δ^0_α-categorical, then for every copy $\mathcal{B}$ of $\mathcal{A}$, there is a degree $\mathbf{d}$ that is $\Sigma^0_{\alpha-1}$ in $\mathcal{B}$ if α is a successor ordinal and Δ^0_α in $\mathcal{B}$ if α is a limit ordinal such that $\mathbf{d}$ computes an isomorphism between $\mathcal{A}$ and $\mathcal{B}$ and all isomorphisms between $\mathcal{A}$ and $\mathcal{B}$ compute $\mathbf{d}$.*

This theorem is proved using a more technical result. We begin by considering a structure $\mathcal{A}$ and a degree $\mathbf{c}$ such that $\mathcal{A}$ is $\mathbf{c}$-computable and Δ^0_α-categorical on the cone above $\mathbf{c}$. We can assume that $\mathcal{A}$ has a c.e. Scott family S of computable Σ_α formulas relative to $\mathbf{c}$ with a certain collection of properties. Then, given a copy $\mathcal{B}$ of $\mathcal{A}$, we can consider the set $S(\mathcal{B})$ of pairs $(\bar{b}, \varphi)$ such that $\varphi(\bar{b})$ is true in $\mathcal{B}$ and $\varphi \in S$. Our degree $\mathbf{d}$ will be the degree of $S(\mathcal{B}) \oplus \mathcal{B} \oplus \mathbf{c}$. We then show that there is an isomorphism $f : \mathcal{A} \cong \mathcal{B}$ such that $f \oplus \mathbf{c} \equiv_T \mathbf{d}$ and then, using a set closely related to $S(\mathcal{B})$, use the properties associated with this particular Scott family to show that $\mathbf{d}$ is the desired degree.

Csima and Harrison-Trainor then proceed to argue that this means that the only natural degrees of categoricity are these degrees: arguments concerning structures found naturally in mathematics tend to relativize, and therefore any natural structure has a given property exactly if it has that property on a cone.

We can see that the key to all of these constructions is the ability to approximate a set in the degree in question well enough to construct a computable structure that encodes it.

2.2. **Bounding this class from above.** In [1], Anderson and Csima turned their attention to classes of degrees that are incompatible with the degrees of categoricity. Their first result may be summarized as follows.

Theorem 2.7. [1] *There is a degree below* $\mathbf{0}''$ *that is not a degree of categoricity; in fact, there is a* Σ^0_2 *degree that is not a degree of categoricity.*

The proof that $\mathbf{0}''$ computes a degree that is not a degree of categoricity actually shows that $\mathbf{0}''$ computes a degree that is low for isomorphism. It is quite straightforward: we simply build a set X by finite extensions using a $0''$ oracle. At stage $\langle \ell, m, k\rangle + 1$, we first extend our finite approximation to ensure that our set is not computable by $\varphi_{\langle \ell,m,k\rangle}$ using $0'$. We then use $0'$ to determine whether our approximation can be extended to a string σ such that Φ^σ_ℓ is not a partial isomorphism from $\mathcal{A}_m$ to $\mathcal{A}_k$; if so, that extension is our new approximation. If not, we use $0''$ to check to see if we can extend our approximation to a string σ such that Φ^σ_ℓ is either not total or not surjective; if so, that extension is our new approximation. Otherwise, we know that any extension of our approximation can be extended to an isomorphism from $\mathcal{A}_m$ to $\mathcal{A}_k$, so we can find a computable isomorphism from $\mathcal{A}_m$ to $\mathcal{A}_k$ and take our new approximation to be the current one.

To compute such a degree that is Σ^0_2, we simply build our set D to be left-c.e. in $0'$. For each tuple $\langle e, i, j\rangle$, we satisfy the requirement that if Φ^D_e is an isomorphism from $\mathcal{A}_i$ to $\mathcal{A}_j$, then there is a computable isomorphism between these structures as well (so, once again, we compute a degree that is low for isomorphism). At each stage, we consider the highest priority requirement (suppose it is the requirement for the tuple $\langle e, i, j\rangle$) and ask if there is a string $\sigma \succeq 1$ such that Φ^σ_e is not a partial isomorphism from $\mathcal{A}_i$ to $\mathcal{A}_j$; if so, we do it and satisfy our requirement. If not, we ask at successive stages whether we can find a string $\sigma \succeq 1$ that can always be extended to a longer partial map from ω to ω. If the answer is always yes, then we can use that functional Φ_e and get a computable isomorphism from $\mathcal{A}_i$ to $\mathcal{A}_j$; if the answer is ever no, we choose a new approximation witnessing this. This is left-c.e. in $0'$ and may injure lower-priority requirements.

Anderson and Csima also demonstrated that the degrees of categoricity are disjoint from the hyperimmune-free degrees:

Theorem 2.8. [1] *No noncomputable hyperimmune-free degree is a degree of categoricity.*

This proof proceeds by contradiction. We assume that a structure $\mathcal{A}$ witnesses that $\mathbf{d}$ is a hyperimmune-free degree of categoricity and that $\mathbf{d}$ computes an f witnessing that $\mathcal{A} \cong \mathcal{B}$. Since $\mathbf{d}$ is hyperimmune free, there must be a computable function h that dominates both f and f^{-1}. This function h is then used to build an infinite computably bounded tree $T \subseteq \omega^{<\omega}$ whose infinite paths code isomorphisms between $\mathcal{A}$ and $\mathcal{B}$. One of these paths is guaranteed to be computable from $\mathbf{0}'$, so there is $g \leq_T 0'$ witnessing that $\mathcal{A} \cong \mathcal{B}$. This means that $\mathcal{A}$ is $\mathbf{0}'$-computably categorical and thus that $\mathbf{d} \leq_T \mathbf{0}'$, which is impossible since $\mathbf{d}$ is hyperimmune free.

Anderson and Csima also proved that if A is a set and G is Cohen 2-generic in A or if G is Cohen 2-generic relative to a perfect tree, then the degree of $G \oplus A$ is not a degree of categoricity. We note that in fact they proved here that all such degrees are low for isomorphism and that this proof is very similar to the proof of Theorem 2.7, so we reserve a comparable proof until Section 3.

We can further restrict the Turing degrees that may be degrees of categoricity as follows. Fokina, Kalimullin, and R. Miller proved in [11] that every strong degree of categoricity is hyperarithmetic using the Effective Perfect Set Theorem [26]. Csima, Franklin, and Shore proved later in [4] that every degree of categoricity, strong or not, is hyperarithmetic. Their proof requires Kreisel's Basis Theorem [31]. To prove this, we begin by taking an arbitrary degree $\mathbf{d}$ that is not hyperarithmetic and an arbitrary computable structure $\mathcal{A}$ and listing all the computable copies of $\mathcal{A}$: $\mathcal{A}_0, \mathcal{A}_1, \ldots$. The class of isomorphisms between $\mathcal{A}_0$ and $\mathcal{A}_1$ is Π^0_2 and thus Σ^1_1, and by Kreisel's Basis Theorem, there is an isomorphism f_1 such that $\mathbf{d} \not\leq_h f_1$. In fact, we can relativize Kreisel's Basis Theorem to find a sequence of isomorphisms $f_0, f_1, \ldots$ such that f_i is an isomorphism between $\mathcal{A}_0$ and $\mathcal{A}_i$ and $\mathbf{d} \not\leq_h f_1 \oplus \ldots \oplus f_i$ for each i. We take an exact pair $\mathbf{a}$ and $\mathbf{b}$ for this sequence and note that both of these degrees can compute an isomorphism between any two copies of $\mathcal{A}$. This means that any degree of categoricity for $\mathcal{A}$ must be below both $\mathbf{a}$ and $\mathbf{b}$. If $\mathbf{d}$ is such a degree, then it must therefore be computable from $f_1 \oplus \ldots \oplus f_n$ for some n, which would lead to a contradiction.

2.3. **Open questions.** We can see that there can only be countably many degrees of categoricity since they are all hyperarithmetic. However, the examples produced are all of the same type and come nowhere near the upper bounds we have established for this class: all known examples are d.c.e. in and above some degree of the form $\mathbf{0}^{(\alpha)}$. All efforts to extend these constructions to even 3-c.e. degrees have failed to date, and indeed Csima and Harrison-Trainor's work shows that no natural structure can even have properly d.c.e. degree. This leads to a first obvious question:

Question 2.9. Is there a degree that is n-c.e. in and above $\mathbf{0}^{(\alpha)}$ for some computable ordinal α and some $n > 2$ that is not a degree of categoricity?

We may also ask a weaker version of this question inspired by the observation that the known degrees of categoricity all have very simple approximations in the intervals $[\mathbf{0}^{(\alpha)}, \mathbf{0}^{(\alpha+1)}]$. Must this always be true?

Question 2.10. Is there a degree of categoricity that is not contained in an interval of the form $[\mathbf{0}^{(\alpha)}, \mathbf{0}^{(\alpha+1)}]$ for some computable ordinal α?

On the more technical side, we note that there is a case that Csima, Franklin, and Shore did not consider in [4]:

Question 2.11. If α is a computable limit ordinal, is every degree that is c.e. or d.c.e. in and above $\mathbf{0}^{(\alpha)}$ a (strong) degree of categoricity?

We also note that the degrees of categoricity for one particular class of structures have been studied: in [11], it is shown that any c.e. degree is the degree of categoricity of some computable algebraic field. It may be illuminating to consider the degrees of categoricity for other nonuniversal structures:

Question 2.12. Which Turing degrees may be degrees of categoricity for a particular class $\mathcal{C}$ of structures?

We now go on to the two most fundamental questions in the area. First of all, all the known degrees of categoricity are strong degrees of categoricity, which leads to the following question:

Question 2.13. Is every degree of categoricity a strong degree of categoricity?

Secondly, we can ask for a full characterization.

Question 2.14. Characterize the Turing degrees that are degrees of categoricity.

3. Degrees that are low for isomorphism

We now turn our attention to Turing degrees that are very far from being degrees of categoricity: those that are low for isomorphism, introduced by Franklin and Solomon in [14]. They use directed graphs to study this concept as the authors considering degrees of categoricity have done, but here these graphs are used because of the need to quantify over all structures in all computable languages. This decision is based on work by Hirschfeldt, Khoussainov, Shore, and Slinko, who proved in [19] that directed graphs are universal in the following sense: arbitrary countable structures $\mathcal{A}$ and $\mathcal{B}$ in a computable language can be coded into countable directed graphs $G(\mathcal{A})$ and $G(\mathcal{B})$ such that

- $\mathcal{A} \cong \mathcal{B}$ if and only if $G(\mathcal{A}) \cong G(\mathcal{B})$,
- $\mathcal{A}$ is computable exactly when $G(\mathcal{A})$ is computable, and
- if $\mathcal{A}$ and $\mathcal{B}$ are computable, then for any Turing degree $\mathbf{d}$, $\mathcal{A} \cong_{\mathbf{d}} \mathcal{B}$ if and only if $G(\mathcal{A}) \cong_{\mathbf{d}} G(\mathcal{B})$.

3.1. Examples of degrees that are low for isomorphism.

The most common theme in these proofs is that of forcing. In fact, any reasonable sort of computability-theoretic forcing at the right level will allow us to produce a degree that is low for isomorphism.

The first type of forcing considered in [14] is forcing with generic reals; specifically, with Cohen and Matthias generic reals. The following theorem is obtained:

Theorem 3.1. [14] *Every Cohen 2-generic degree and every Matthias 3-generic degree is low for isomorphism.*

The proofs in this paper rely heavily on machinery from reverse mathematics [20, 21]; here we present a direct proof for the Cohen generic case.

Let G be a Cohen 2-generic real. (In the future, when we write "generic" without further qualification, we will mean "Cohen generic.") We must show that for any $\mathcal{A}$ and $\mathcal{B}$ such that $\mathcal{A} \cong_G \mathcal{B}$, $\mathcal{A} \cong_{\Delta^0_1} \mathcal{B}$. We begin by considering the following statements:

- Φ_e^X maps an element of $\mathcal{A}$ to an element of $\mathcal{B}$ that witnesses that Φ_e^X is not an isomorphism from $\mathcal{A}$ to $\mathcal{B}$.
- Φ_e^X is total.
- Φ_e^X is surjective.

We note that the first of these statements is $\Sigma_1^{0,X}$, since it states that at some stage, Φ_e^X maps an element of $\mathcal{A}$ to an element of $\mathcal{B}$ that do not satisfy the same formulas in the atomic diagram. It is clear that the latter two statements are $\Pi_2^{0,X}$. Now we fix an $\mathcal{A}$ and $\mathcal{B}$ such that $\mathcal{A} \cong_G \mathcal{B}$. Since G is 2-generic, it must force the truth or falsity of each of the above statements, and since G does compute an isomorphism between $\mathcal{A}$ and $\mathcal{B}$, we know that G must force the first statement to be false and the others to be true. Let ρ be the initial segment of G that forces all these things. We will construct a computable sequence $\rho = \sigma_0 \preceq \sigma_1 \preceq \sigma_2 \preceq \dots$ in stages, defining σ_i at stage i, so that each new term σ_i lets us define a longer partial isomorphism between $\mathcal{A}$ and $\mathcal{B}$.

To define σ_{i+1} for an even i, we consider the partial isomorphism found through σ_i. There is a least element n_{i+1} of $\mathcal{A}$ whose image is undefined by this partial isomorphism, so we search above σ_i for an extension σ_{i+1} that, when used as an oracle on Φ_e, will place n_{i+1} in our domain. Such an extension must exist because ρ has already forced totality, and the mapping it finds must be extendible to an isomorphism between $\mathcal{A}$ and $\mathcal{B}$ because ρ has forced the first statement to be false. Now we have extended the initial segment of the domain of our partial isomorphism.

To define σ_{i+1} for an odd i, we do the same thing, but in reverse: there is a least element m_{i+1} of $\mathcal{B}$ that is not yet mapped to by the partial isomorphism defined at the end of the previous step using σ_i as an oracle. Now, we search above σ_i for an extension σ_{i+1} that, when used as an oracle with Φ_e, will place m_{i+1} in our range. In this case, such an extension must exist because ρ has forced surjectivity, and we have preserved our ability to extend to an isomorphism between $\mathcal{A}$ and $\mathcal{B}$ as before.

Franklin and Solomon also use Sacks forcing with computable perfect trees to produce a degree that is low for isomorphism that are minimal and hyperimmune free as well (see Chapter V.5 in [29] for a discussion of this sort of forcing). Using a noneffective enumeration of all pairs $(\mathcal{A}_i, \mathcal{B}_i)$ of all infinite computable directed graphs, we build a sequence of computable perfect trees $T_0 \supseteq T_1 \supseteq \ldots$ such that T_0 is the identity tree and $T_i(\lambda) \subseteq T_{i+1}(\lambda)$ for each i. The resulting set D is the set such that $T_i(\lambda) \preceq D$ for every i. Four kinds of requirements must be satisfied in this proof:

- Noncomputability: For every e, $D \neq \Phi^e$.
- Hyperimmune-freeness: For every e, either Φ_e^D is not total or Φ_e^D is majorized by a computable function.
- Minimality: For every e, if Φ_e^D is total, then either Φ_e^D is computable or $D \leq_T \Phi_e^D$.
- Lowness for isomorphism: For every e and i, if Φ_e^D is an isomorphism from $\mathcal{A}_i$ to $\mathcal{B}_i$, then $\mathcal{A}_i \cong_{\Delta_1^0} \mathcal{B}_i$.

The construction once again proceeds by stages, and one of these requirements is satisfied at each stage. The first three requirements are satisfied in the usual way (see [29] for details). We will discuss the lowness for isomorphism requirements here.

Suppose we want to ensure that the lowness for isomorphism requirement is satisfied for Φ_e and the pair $(\mathcal{A}_i, \mathcal{B}_i)$. Without loss of generality, we can assume we have already satisfied the hyperimmune-freeness requirement for e and that we are working at stage $s+1$. We now proceed by cases.

If we satisfied the hyperimmune-freeness requirement by guaranteeing that Φ_e^D will not be total, then our lowness for isomorphism requirement is satisfied trivially and we simply choose the root of our new tree T_{s+1} to be any nonroot element of T_s.

If we satisfied the hyperimmune-freeness requirement by guaranteeing that Φ_e^D will be total and majorized by a computable function, we know that Φ_e^A is total for every branch A of T_s. We now check to see whether there is a string σ and a number n such that $\Phi_e^{T_s(\sigma)}\restriction n$ halts and $\Phi_e^{T_s(\sigma)}\restriction n$ is not a partial isomorphism from $\mathcal{A}_i$ to $\mathcal{B}_i$. If there is such a string σ, we take T_{s+1} to be the full subtree of T_s above σ. Otherwise, we know that any branch in T_s will give us an isomorphism between $\mathcal{A}_i$ and $\mathcal{B}_i$, and we can define a new subtree inside T_s computably so a computable isomorphism can actually be found.

Franklin and Solomon also asked in [14] if one could "cap" the level of Cohen genericity associated with lowness for isomorphism at 2-genericity: in other words, if it is possible for a 1-generic that is not computed by a 2-generic to be low for isomorphism. In [16], Franklin and Turetsky answered this question in the negative by constructing a 1-generic G that satisfies the following requirements:

(One$_e$): G either meets or avoids the Σ_1^0 set W_e.
(Two$_i$): There is a Σ_2^0 set X_i such that if $\Phi_i^Y = G$, then Y neither meets nor avoids X_i.
(IM$_{\langle i,j_1,j_2\rangle}$): if Φ_i^G is an isomorphism between $\mathcal{A}_{j_1}$ and $\mathcal{A}_{j_2}$, then $\mathcal{A}_{j_1} \cong_{\Delta_1^0} \mathcal{A}_{j_2}$.

The first requirement can be satisfied through a standard finite injury approach: if we find at some stage that we can extend our finite approximation to G to meet W_e, we do so, and it is satisfied automatically otherwise.

Now, to satisfy (Two$_i$), we use infinitely many subrequirements:

(Two$_{\langle i,\tau\rangle}$): If there is a $Y \succ \tau$ such that $\Phi_i^Y = G$, then Y does not meet X_i and there is some string $\rho \succ \tau$ such that $\rho \in X_i$.

In meeting each of these subrequirements, we construct our X_i. Suppose we have a finite approximation g to G and we are trying to satisfy (Two$_{\langle i,\tau\rangle}$). We reserve the next bit b at position $|g|$ for our use and initially require that $G(b) = 0$. Now we try to find a string $\rho \succ \tau$ such that there is no Y extending ρ with $\Phi_i^Y = G$. If at some point we see a ρ extending τ where $\Phi_i^\rho \succeq g^\frown 0$, we put this ρ in X_i and change $G(b)$ to 1. Since the construction is $\mathbf{0}''$, the set of all these ρs over all τ will be Σ_2^0.

We argue briefly that this X_i serves its intended purpose: that for any i and Y such that $\Phi_i^Y = G$, Y cannot meet or avoid X_i. If Y avoids X_i, then we fix an initial segment τ of Y where this happens and consider the appropriate node on the true path. By our definition of X_i, there is no $\rho \succeq \tau$ with $\Phi_i^\rho \preceq g^\frown 0$, so $\Phi_i^Y(|g|) \neq 0$. However, if there is no such ρ, $g^\frown 0$ will be an initial segment of G, so Φ_i^Y and G must differ at position $|g|$.

If Y does meet X_i, then we fix an initial segment ρ where this happens and consider the (Two$_{\langle i,\tau\rangle}$)-strategy that caused us to add this ρ to X_i and the node on the priority tree that witnesses this. By definition, we know that we have a potential initial segment g of G associated with this node and that $\Phi_i^\rho \succeq g^\frown 0$. There are two possible scenarios. In the first, the node in question is to the left of the true path, and the string g is not actually an initial segment of G. Therefore, we cannot have $\Phi_i^Y = G$. In the second, the node in question is actually on the true path. In this case, $g^\frown 1$ will be an initial segment of G, and Φ_i^Y and G must disagree at position $|g|$.

To satisfy (IM$_{\langle i,j_1,j_2\rangle}$), we use a standard infinitary construction. We establish a length of agreement function for the appropriate node on the priority tree. If at some stage we can find a string extending our current approximation that defines a longer isomorphism, we choose it as our new approximation and take the infinite outcome at our node on the priority tree; otherwise we choose a finite outcome.

3.2. **Bounding this class from above.** Franklin and Solomon also identify significant classes of degrees that cannot be low for isomorphism. The first such class is the nontrivial Δ_2^0 degrees:

Theorem 3.2. *No nontrivial Δ_2^0 degree is low for isomorphism and thus no degree that computes a nontrivial Δ_2^0 degree is low for isomorphism.*

The proof is quite straightforward. We take a representative D of a noncomputable Δ_2^0 degree $\mathbf{d}$ and fix a Δ_2^0 approximation $\langle D_s \rangle$ to it. We then use this approximation to construct two computable directed graphs, G and H, so that the unique isomorphism between them is Turing equivalent to D.

We begin by placing a $(n+2)$-cycle in each of G and H for every $n \in \omega$. The $(n+2)$-cycle component will code n's membership in D. Then, for each $(n+2)$-cycle in G, we add an arrow from some element x_n to a new element a_n, and for each $(n+2)$-cycle in H, we add an arrow from some element y_n to a new element b_n. At this point, $n \notin D$, G and H are isomorphic, and this isomorphism must map a_n to b_n.

If, at stage s, n enters D, we add a new element a' to G and a new element b' to H so that there are edges from a_n to a' and from x_n to a' and edges from b' to b_n and y_n to b'. We still have an isomorphism between G and H, but now the isomorphism must map a_n to b' and a' to b_n.

If n exits D at a later stage, we add new elements a'' and b'' to G and H respectively. This time, we add edges from a'' to a_n and from x_n to a'' and edges from b_n to b'' and from y_n to b''. We can see that G and H are still isomorphic, but the isomorphism maps a_n to b_n once more.

We can repeat this pattern and see that since after some point our approximation to D will be constant on n, the $(n+2)$-cycles in G and H will stabilize, and the isomorphism between G and H will map a_n to b_n if and only if $n \notin D$. This is enough to see that $G \cong_{\mathbf{c}} H$ if and only if $\mathbf{d} \leq_T \mathbf{c}$.

This lets us see that no degree above $\mathbf{0}'$ is low for isomorphism either, since the degrees that are low for isomorphism are closed downward.

They also show using a similar proof that if a degree can compute a separating set for a pair of computably inseparable c.e. sets, that degree cannot be low for isomorphism.

Franklin and Solomon then turn their attention to measure and prove the following theorem:

Theorem 3.3. *No Martin-Löf random degree is low for isomorphism.*

Here, we sketch a proof that a set of degrees of measure one is not low for isomorphism and then discuss briefly how it can be modified to prove the theorem above.

We begin by observing that we can produce a class of degrees that are not low for isomorphism with some positive measure and conclude using Kolmogorov's 0-1 law that it must actually have measure 1. We construct two isomorphic computable directed graphs G and H and a Π^0_1 class $\mathcal{C}$ so that

- **(P1):** $G \ncong_{\Delta^0_1} H$,
- **(P2):** $\mu(\mathcal{C}) \geq \frac{1}{2}$, and
- **(P3):** if $X \in \mathcal{C}$, then X can compute an isomorphism from G to H.

(P1) and (P3) clearly combine to guarantee that no element of $\mathcal{C}$ can be low for isomorphism and will not need to be modified when we require Martin-Löf randomness instead of simply positive measure; the only adaptation we will need to make to (P2) is to construct a sequence of trees whose measure increases in a very controlled way and forms the complement of a Martin-Löf test.

To satisfy (P1), we meet the following requirement:

$$R_e\text{: } \Phi_e \text{ is not an isomorphism from } G \text{ to } H.$$

As we satisfy this, we ensure that our diagonalization strategy for R_e does not remove too much measure from $\mathcal{C}$, thus satisfying (P2) at the same time. Finally, to satisfy (P3), we construct a Turing functional Γ so that for any X in $\mathcal{C}$, Γ^X is an isomorphism from G to H.

Our graphs G and H initially begin as infinitely many e-components for each $e \in \omega$, where an e-component is an $(e+3)$-cycle with a coding node u distinguished by a loop. In the course of our construction, we will add "tails" to coinfinitely infinitely many of the e-components when we actively diagonalize to satisfy R_e: a "tail" consists of two nodes x_0 and x_1 with arrows from u to x_0 to x_1 to x_0. This guarantees that a set X can compute an isomorphism between G and H if and only if it can compute a bijection between the coding nodes in G and H and, furthermore, successfully match up the tailed and untailed coding nodes.

First we discuss how we will meet a single requirement R_e. To do this, we fix an e-component in G and diagonalize against its coding node a_e. If there is no stage s where a_e is mapped to a coding node b of an e-component in H, the requirement is satisfied trivially. Otherwise, we actively diagonalize by adding tails to an infinite coinfinite set of coding nodes of e-components in H, including b. We also add tails to an infinite coinfinite set of coding nodes of e-components in G but ensure that a_e is not among them to make sure that no isomorphism between G and H can map a_e to b as Φ_e does.

These infinite coinfinite sets are also used to define the Turing functional Γ and to ensure that enough reals can compute an isomorphism between G_0 and H_0. At stage 0, we define Γ so that for each $e \in \omega$ and each string σ of length $e+2$, we define Γ^σ so it maps the coding nodes for e-components in G_0 to e-components in H_0. Furthermore, we make sure that different strings of the same length do not produce the same mapping. Now observe that these mappings will continue to extend to isomorphisms at later stages as long as they map untailed components to untailed components and tailed components to tailed components (when tails are added to our structures at later stages). If, however, we satisfy R_e by adding tails to some components, we must make sure that we remove any branch X from our tree such that Γ^X maps an untailed component to a newly tailed component or vice versa.

Now we describe an abbreviated version of R_0's strategy to give an idea of how to balance these conflicting requirements. To ensure that (P2) holds, we ensure that we do not remove more than $\frac{1}{4}$ of the measure from our tree. To do this, we choose the infinite coinfinite sets that we will use to diagonalize against a_0 carefully and define Γ in such a way that, no matter what coding node Φ_0 may map a_0 to, we can diagonalize in such a way that we can remove no more than one string of measure $\frac{1}{4}$ and still have the oracles remaining in the class correctly map untailed components to untailed components and tailed components to tailed components.

The class $\mathcal{C}$ will be defined as the set of branches in the intersection of our computable sequence of trees $2^{<\omega} = T_0 \supseteq T_1 \supseteq \ldots$, and this class is obtained by removing the strings that are no longer appropriate oracles for Γ given the changes in G and H that have taken place. Since we code information about where the e-components map at a string of length $e+2$ and we have arranged the coding nodes so no more than one of the strings of that length will fail to code an isomorphism, we remove at most $\frac{1}{4} + \frac{1}{8} + \ldots = \frac{1}{2}$ from our tree overall, and we have a tree of positive measure.

Now we explain how this proof can be modified to show that no Martin-Löf random real is low for isomorphism. We begin by recalling the definition of Martin-Löf randomness; for a more thorough discussion of algorithmic randomness, see [7].

Definition 3.4. A *Martin-Löf* test is an effectively c.e. sequence $\langle V_i \rangle$ of subsets of $2^{<\omega}$ such that $\mu([V_i]) \leq 2^{-i}$ for all i, and a real X is *Martin-Löf random* if $X \notin \cap_i [V_i]$ for every Martin-Löf test $\langle V_i \rangle$.

Note that the class $\mathcal{C}$ we built has measure at least $\frac{1}{2}$; its complement is therefore a Σ^0_1 class of measure no more than $\frac{1}{2}$. Its complement could therefore be the first component of a Martin-Löf test. We construct an entire Martin-Löf test by constructing not just one Π^0_1 class $\mathcal{C}$ but an effective sequence of nested Π^0_1 classes $\mathcal{C}_0 \subseteq \mathcal{C}_1 \subseteq \ldots$ where the i^{th} class has measure at least $1 - \frac{1}{2^{i+1}}$. Their complements will therefore form a Martin-Löf test, and any Martin-Löf random real X will be in $\mathcal{C}_i$ for some i and will thus not be low for isomorphism.

To construct this sequence of classes, we repeat the construction we just described as follows. $\mathcal{C}_0$ will be generated as above. We arrange for each class $\mathcal{C}_{i+1}$ to be larger than the previous one as follows. When we remove a string σ from $\mathcal{C}_i$ (or, indeed, any previous class), we do not remove that string from $\mathcal{C}_{i+1}$. Instead, we start a new version of the construction inside this string. If a new diagonalization process within these constructions requires that we remove a string from $\mathcal{C}_{i+1}$, it will be longer and thus we will remove less measure from $\mathcal{C}_{i+1}$ than we did from $\mathcal{C}_i$ to satisfy any given diagonalization requirement; with some planning, we can require that $\mu(\mathcal{C}_i) \geq 1 - \frac{1}{2^i}$ and thus that the complement of $\mathcal{C}_i$ can be the i^{th} component of our Martin-Löf test.

We also observe that this is the strongest result that can be obtained concerning lowness for isomorphism and randomness: the computably random degrees and those that are low for isomorphism are not disjoint, since every high degree contains a computably random real [28], and there is a high 2-generic.

3.3. **Open questions.** We first observe that Franklin and Turetsky's result still leaves a gap in the genericity hierarchy:

Question 3.5. Is there a properly 1-generic degree that is low for isomorphism and not computable from a weakly 2-generic?

While there seems to be no easy way to adapt their construction to answer this question, it may be possible to construct such a degree in some other way.

Csima has also defined a similar notion, *lowness for categoricity.* She has defined a degree $\mathbf{d}$ to be low for categoricity if every computable structure that is $\mathbf{d}$-computably categorical is already computably categorical [3]. Lowness for isomorphism clearly implies lowness for categoricity, but whether the converse holds is uncertain.

Question 3.6. Is every degree that is low for categoricity also low for isomorphism?

As with degrees of categoricity, we may also consider the degrees that are low for isomorphism for a particular class of structures. Suggs has studied several cases, including linear orders [35]; some of this work appears in [14].

Question 3.7. Describe the degrees that are low for isomorphism for a particular class of structures $\mathcal{C}$.

We end with two questions that are rather hard and closely related:

Question 3.8. Are there other natural classes of degrees that are either subsets of or disjoint to the degrees that are low for isomorphism?

Some candidates for such classes include the computably traceable degrees (a subset of the hyperimmune-free degrees) and the c.e. traceable degrees.

We end with, once more, the obvious question.

Question 3.9. Characterize the Turing degrees that are low for isomorphism.

4. Discussion and musings

While the degrees of categoricity and those that are low for isomorphism both lack a full characterization, they lack this characterization in very different ways. It is easy to describe all known degrees of categoricity: they all belong to intervals of the form $[\mathbf{0}^{(\alpha)}, \mathbf{0}^{(\alpha+1)}]$ for some computable ordinal α; in fact, they are all even d.c.e. in and above a degree of the form $\mathbf{0}^{(\alpha)}$ for some such α. All of the known constructions are very similar—one codes information about an appropriate set in the degree in question into two copies of the same structure, using an approximation of this set—and none of them extend to the 3-c.e. case. It is known that the degrees of categoricity are all hyperarithmetic and thus that this class is countable, so it must be null. It is also small with

TABLE 1. Lowness for isomorphism: Categories of degrees

	Low for isomorphism	Not low for isomorphism
Nontrivial Δ^0_2	no	yes: Δ^0_2
Nontrival Δ^0_3	yes: Cohen forcing	yes: Martin-Löf random
Minimal	yes: perfect trees	yes: Δ^0_2
Not minimal	yes: Cohen forcing	yes: Δ^0_2
Hyperimmune	yes: Cohen forcing	yes: Δ^0_2
Hyperimmune-free	yes: perfect trees	yes: Martin-Löf random

respect to category, since no such degree can be 2-generic. Furthermore, no degree of categoricity can be hyperimmune free.

On the other hand, there is no convenient way to describe the class of degrees that are low for isomorphism. While the Cohen 2-generics and Matthias 3-generics are known to be subsets of this class and no Martin-Löf degree can belong to it, most of the results in this area consist of showing that some degree of a certain kind is low for isomorphism and another degree of the same kind is not; a summary appears in Table 1. For any category that does contain degrees that are low for isomorphism, the proof method is indicated; for any category that does not, one type of counterexample is indicated. It is clear that lowness for isomorphism is not closely related to any natural class except the Δ^0_2 degrees.

Most of the results on lowness for isomorphism were obtained by forcing. In general, any type of forcing that will allow us to force a functional to be total and surjective and never to fail to be a partial isomorphism will permit us to construct a set that is low for isomorphism.[1] However, lowness for isomorphism is not a property strictly determined by the ability to force: Anderson and Csima constructed a Σ^0_2 example of a degree that is low for isomorphism using a standard injury argument.

We can also argue that there are very few degrees that are low for isomorphism: they have measure 0 since no Martin-Löf degree is low for isomorphism. However, unlike the degrees of categoricity, they are large with respect to category since every 2-generic degree is low for isomorphism.

In short, the degrees of categoricity and the degrees that are low for isomorphism appear to be diametrically opposed, bros. All known degrees of categoricity have a simple approximation: one that is no more than d.c.e. in some jump of $\mathbf{0}$. In some way, they form the "backbone" of the Turing degrees. The degrees that are low for isomorphism, on the other hand, are in general, those that cannot be effectively approximated (Anderson and Csima's Σ^0_2 example is, once again, a delightfully puzzling exception). Unsurprisingly, they are bounded away from each other: no degree that is comparable to $\mathbf{0}'$ is low for isomorphism, which is where all the all the known degrees of categoricity reside.

Since both of these classes resist characterization and they are bounded away from each other, it may be of interest to consider the class of degrees that fall between them. What kinds of degrees are neither low for isomorphism nor degrees of categoricity? They must resist approximation, but

[1]We note with some amusement that the isomorphism condition is actually lower in the arithmetic hierarchy than the others and is therefore not the condition that determines the degree of genericity necessary to force lowness for isomorphism.

not too much. Furthermore, this class of degrees is large with respect to measure and small with respect to category. There are certainly natural classes of degrees with this property, and if one of them proved to be disjoint from both of the classes we have considered in this paper, it might illuminate the features inherent in each of them.

Question 4.1. Is there a natural class of degrees that is disjoint from both the degrees of categoricity and the degrees that are low for isomorphism?

We also notice that the degrees that are low for isomorphism do not form an ideal in the Turing degrees: while they are downward closed, they are not closed under join because there is a pair of 2-generics whose join computes $\mathbf{0}'$. However, we may ask a follow-up question: are we, in fact, considering the most appropriate degree structure? In algorithmic randomness, the Schnorr trivial reals have unexpected properties in the Turing degrees [6, 12], but they behave as one expects trivial reals to behave in the truth-table degrees [15]. It may be that these notions are better understood in another degree structure:

Question 4.2. Describe the behavior of the degrees of categoricity and the degrees that are low for isomorphism in an alternate degree structure such as the weak truth-table or truth-table degrees.

References

[1] Bernard A. Anderson and Barbara F. Csima. Degrees that are not degrees of categoricity. *Notre Dame J. Formal Logic.* To appear.
[2] C.J. Ash and J. Knight. *Computable Structures and the Hyperarithmetical Hierarchy.* Number 144 in Studies in Logic and the Foundations of Mathematics. North-Holland, 2000.
[3] Barbara Csima. Degrees of categoricity and related notions, November 2013. BIRS Computable Model Theory Workshop.
[4] Barbara F. Csima, Johanna N.Y. Franklin, and Richard A. Shore. Degrees of categoricity and the hyperarithmetic hierarchy. *Notre Dame J. Formal Logic*, 54(2):215–231, 2013.
[5] Barbara F. Csima and Matthew Harrison-Trainor. Degrees of categoricity on a cone. *J. Symbolic Logic.* To appear.
[6] Rod Downey, Evan Griffiths, and Geoffrey LaForte. On Schnorr and computable randomness, martingales, and machines. *Math. Log. Q.*, 50(6):613–627, 2004.
[7] Rodney G. Downey and Denis R. Hirschfeldt. *Algorithmic Randomness and Complexity.* Springer, 2010.
[8] Yu. L. Ershov. A hierarchy of sets, I. *Algebra and Logic*, 7(1):212–232, 1968. English translation. Originally appearing in Algebra i Logika, 7(1):47–74, 1968 (Russian).
[9] Yu. L. Ershov. On a hierarchy of sets, II. *Algebra and Logic*, 7(4):25–43, 1968. English translation. Originally appearing in Algebra i Logika, 7(4):15–47, 1968 (Russian).
[10] Yu. L. Ershov. On a hierarchy of sets, III. *Algebra and Logic*, 9(1):20–31, 1970. English translation. Originally appearing in Algebra i Logika, 9(1):34–51, 1970 (Russian).
[11] Ekaterina B. Fokina, Iskander Kalimullin, and Russell Miller. Degrees of categoricity of computable structures. *Arch. Math. Logic*, 49(1):51–67, 2010.
[12] Johanna N.Y. Franklin. Schnorr trivial reals: A construction. *Arch. Math. Logic*, 46(7–8):665–678, 2008.
[13] Johanna N.Y. Franklin. Lowness and highness properties for randomness notions. In T. Arai et al., editor, *Proceedings of the 10th Asian Logic Conference*, pages 124–151. World Scientific, 2010.
[14] Johanna N.Y. Franklin and Reed Solomon. Degrees that are low for isomorphism. *Computability*, 3(2):73–89, 2014.
[15] Johanna N.Y. Franklin and Frank Stephan. Schnorr trivial sets and truth-table reducibility. *J. Symbolic Logic*, 75(2):501–521, 2010.
[16] Johanna N.Y. Franklin and Dan Turetsky. Genericity and lowness for isomorphism. Submitted.
[17] A. Fröhlich and J. C. Shepherdson. Effective procedures in field theory. *Philos. Trans. Roy. Soc. London. Ser. A.*, 248:407–432, 1956.

[18] Valentina S. Harizanov. Pure computable model theory. In *Handbook of recursive mathematics, Vol. 1*, volume 138 of *Stud. Logic Found. Math.*, pages 3–114. North-Holland, Amsterdam, 1998.
[19] D. Hirschfeldt, B. Khoussainov, R. Shore, and A. Slinko. Degree spectra and computable dimensions in algebraic structures. *Ann. Pure Appl. Logic*, 115(1-3):71–113, 2002.
[20] Denis R. Hirschfeldt and Richard A. Shore. Combinatorial principles weaker than Ramsey's theorem for pairs. *J. Symbolic Logic*, 72(1):171–206, 2007.
[21] Denis R. Hirschfeldt, Richard A. Shore, and Theodore A. Slaman. The atomic model theorem and type omitting. *Trans. Amer. Math. Soc.*, 361(11):5805–5837, 2009.
[22] Denis R. Hirschfeldt and Walker M. White. Realizing levels of the hyperarithmetic hierarchy as degree spectra of relations on computable structures. *Notre Dame J. Formal Logic*, 43(1):51–64 (2003), 2002.
[23] Julia F. Knight. Degrees coded in jumps of orderings. *J. Symbolic Logic*, 51(4):1034–1042, 1986.
[24] David Marker. Non Σ_n axiomatizable almost strongly minimal theories. *J. Symbolic Logic*, 54(3):921–927, 1989.
[25] Antonio Montalbán. Priority arguments via true stages. *J. Symb. Log.*, 79(4):1315–1335, 2014.
[26] Y.N. Moschovakis. *Descriptive Set Theory.* Number 100 in Studies in Logic and the Foundations of Mathematics. North-Holland, 1980.
[27] André Nies. *Computability and Randomness.* Clarendon Press, Oxford, 2009.
[28] André Nies, Frank Stephan, and Sebastiaan A. Terwijn. Randomness, relativization and Turing degrees. *J. Symbolic Logic*, 70(2):515–535, 2005.
[29] Piergiorgio Odifreddi. *Classical Recursion Theory.* Number 125 in Studies in Logic and the Foundations of Mathematics. North-Holland, 1989.
[30] Piergiorgio Odifreddi. *Classical Recursion Theory, Volume II.* Number 143 in Studies in Logic and the Foundations of Mathematics. North-Holland, 1999.
[31] Gerald E. Sacks. *Higher Recursion Theory.* Springer-Verlag, 1990.
[32] Theodore A. Slaman and Robert M. Solovay. When oracles do not help. In *Fourth Annual Workshop on Computational Learning Theory*, pages 379–383. Morgan Kaufman., Los Altos, CA, 1991.
[33] Robert Soare. The Friedberg-Muchnik theorem re-examined. *Canadian J. of Math.*, 24:1070–1078, 1972.
[34] Robert I. Soare. *Recursively Enumerable Sets and Degrees.* Perspectives in Mathematical Logic. Springer-Verlag, 1987.
[35] Jacob Suggs. *Degrees that are low for $\mathcal{C}$ isomorphism.* PhD thesis, University of Connecticut, 2015.

Department of Mathematics, Room 306, Roosevelt Hall, Hofstra University, Hempstead, NY 11549-0114, USA
E-mail address: `johanna.n.franklin@hofstra.edu`