Computable Transformations of Structures

Russell Miller*

Queens College – C.U.N.Y., 65-30 Kissena Blvd.
Queens NY 11367 USA
Graduate Center of C.U.N.Y., 365 Fifth Avenue
New York, NY 10016 USA
qcpages.qc.cuny.edu/~rmiller

Abstract. The *isomorphism problem*, for a class of structures, is the set of pairs of structures within that class which are isomorphic to each other. Isomorphism problems have been well studied for many classes of computable structures. Here we consider isomorphism problems for broader classes of countable structures, using Turing functionals and applying the notions of finitary and countable computable reductions which have been developed for equivalence relations more generally.

1 Introduction

In much of mathematics, two first-order structures which are isomorphic to each other are treated as being exactly the same for all purposes: the objects of study are really the equivalence classes under isomorphism, rather than the structures themselves. Computable structure theory addresses this situation at a deeper level. It is well known that two isomorphic structures may have substantially different algorithmic properties, and therefore, when we consider questions of computability for first-order structures, isomorphism is far too coarse an equivalence relation to be ignored. The fundamental equivalence relation in this discipline is *computable isomorphism*: two structures (both countable, with domain ω) are computably isomorphic if some Turing-computable function on ω is in fact an isomorphism between them. In this case, essentially all known computability-theoretic properties transfer from either structure to the other. In theoretical computer science, complexity theory would go deeper yet, but we will not focus on those questions here.

Of course, the question of whether two structures are isomorphic remains extremely important in computable structure theory. Instead of being so low-level as to be ignored (as in much of model theory), it becomes an object of serious study. The statement that structures $\mathcal{A}$ and $\mathcal{B}$ are isomorphic is on its face a Σ_1^1 sentence about $\mathcal{A}$ and $\mathcal{B}$. (For the rest of this article, all structures are countable with domain ω.) In specific cases, however, it may be not be a Σ_1^1-complete question, but may lie at various levels in the hyperarithmetical hierarchy instead. For

* The author was supported by Grant # DMS – 1362206 from the N.S.F., and by grants from the PSC-CUNY Research Award Program and the Queens College Research Enhancement Fund.

example, for algebraically closed fields, isomorphism depends solely on the characteristic of the field and its transcendence degree over its prime subfield, both of which can be expressed with just a few first-order quantifiers. Vector spaces over $\mathbb{Q}$ are quite similar: for both, the question of whether two computable models of the given theory are isomorphic is a Π^0_3-complete question. The study of isomorphism problems often turns into a search for *invariants*, such as the characterstic and the transcendence degree, which determine isomorphism.

On the other hand, it is known that for computable graphs (which simply means computable symmetric irreflexive subsets of ω^2), the isomorphism problem is Σ^1_1-complete. In [5], Friedman and Stanley created a framework for showing other isomorphism problems to be equally difficult. They showed, for example, that given any two computable graphs $\mathcal{G}_0$ and $\mathcal{G}_1$, one can produce computable linear orders $\mathcal{L}_0$ and $\mathcal{L}_1$ such that

$$\mathcal{G}_0 \cong \mathcal{G}_1 \iff \mathcal{L}_0 \cong \mathcal{L}_1.$$

The "production" of these linear orders is a hyperarithmetic procedure – indeed, a computable procedure – and therefore the isomorphism problem for computable linear orders must aso be Σ^1_1-complete.

However, if one knows that the isomorphism problem for computable algebraically closed fields is Π^0_3-complete and wishes to show the same for computable rational vector spaces, a hyperarithmetic procedure in general is insufficient. For classification at these levels, effective procedures are required, and have been examined in [2–4, 8], by Calvert, Cummins, Knight, S. Miller, and Vanden Boom, in various combinations. The results in [2, Section 4] yield a computable function f which accepts as input the indices (e_0, e_1) of any two computable algebraically closed fields and computes the indices $(i_0, i_1) = f(e_0, e_1)$ of two computable rational vector spaces which are isomorphic if and only if the original two algebraically closed fields were. (By an *index* e for a computable structure $\mathcal{A}$, we mean a number such that the e-th partial computable function φ_e is the characteristic function of the atomic diagram of $\mathcal{A}$, under a fixed coding into ω of atomic sentences in the language of $\mathcal{A}$ with constants from ω.)

The method here works well for computable structures, but the results are sometimes surprising. For example, the isomorphism problem for computable algebraic fields of characteristic 0 (that is, subfields of the algebraic closure $\overline{\mathbb{Q}}$) turns out to be only Π^0_2-complete. It does reduce to the isomorphism problem for algebraically closed fields, but not vice versa – which is puzzling, since $\mathbf{ACF}_0$ has straightforward invariants determining the isomorphism type, while no such invariants are known for the class of algebraic fields.

2 Equivalence Relations on Cantor Space

Our purpose here is to bring to this situation methods from the study of Borel reductions on equivalence relations. We begin by introducing that topic, which has been well studied in descriptive set theory with a focus on equivalence relations on Cantor space 2^ω, and more recently has been extended by many authors to the context of equivalence relations on ω itself.

Let E and F be equivalence relations on domains S and T, respectively. A *reduction* of E to F is a function $g : S \to T$ such that:

$$(\forall x_0, x_1 \in S)\ [x_0\ E\ x_1 \iff g(x_0)\ F\ g(x_1)].$$

If this holds, then by computing g and deciding the relation F, one can decide E as well. Thus E is "no harder to decide" than F, at least modulo the difficulty of computing g.

The next definition was given in [10].

Definition 1. *Let E and F be equivalence relations on subsets $\mathfrak{C}$ and $\mathfrak{D}$ of 2^ω, respectively. A* computable reduction *of E to F is a reduction $g : \mathfrak{C} \to \mathfrak{D}$ given by a computable function Φ (that is, an oracle Turing functional) on the reals involved:*

$$(\forall A \in \mathfrak{C})(\forall x \in \omega)\ \chi_{g(A)}(x) = \Phi^A(x).$$

If such a reduction exists, then E is computably reducible to F*, denoted $E \leq_0 F$.*

Descriptive set theorists usually eschew this definition in favor of the more general concept of a *Borel reduction*, which is to say, a reduction that happens to be a Borel function. This is the context in which Friedman and Stanley developed their work. More recently, computability theorists have taken to considering computable functions (from ω to ω) as reductions, in the context of equivalence relations on ω. The term "computable reduction" therefore often refers to that context, but we will use it here as well for the reductions described in Definition 1, trusting the reader to distinguish the two concepts based on the equivalence relations in question.

Another refinement of reducibilities on equivalence relations was introduced by Ng and the author in [11]. Studying equivalence relations on ω, they defined *finitary reducibilities*. In the context of Cantor space, it is natural to extend their notion to all cardinals $\mu < 2^\omega$ (as indeed was suggested in their article), yielding the following definitions, which also appeared in [10].

Definition 2. *For equivalence relations E and F on domains S and T, and for any cardinal $\mu < |S|$, we say that a function $g : S^\mu \to T^\mu$ is a μ-*ary reduction *of E to F if, for every $\boldsymbol{x} = (x_\alpha)_{\alpha\in\mu} \in S^\mu$, we have*

$$(\forall \alpha < \beta < \mu)\ [x_\alpha\ E\ x_\beta \iff g_\alpha(\boldsymbol{x})\ F\ g_\beta(\boldsymbol{x})],$$

where $g_\alpha : S^\mu \to T$ are the component functions of $g = (g_\alpha)_{\alpha<\mu}$. For limit cardinals μ, a related notion applies with $<\mu$ in place of μ: a function $g : S^{<\mu} \to T^{<\mu}$ which restricts to a ν-ary reduction of E to F for every cardinal $\nu < \mu$ is called a $(<\mu)$-ary reduction. *(For $\mu = \omega$, an ω-ary reduction is a* countable reduction*, and a $(<\omega)$-ary reduction is a* finitary reduction.*)*

When $S \subseteq 2^\omega$ and $T \subseteq 2^\omega$ and the μ-ary reduction g is computable, we write $E \leq_0^\mu F$, with the natural adaptation $E \leq_\alpha^\mu F$ for α-jump μ-ary reductions. Likewise, when a $(<\mu)$-ary reduction g is α-jump computable, we write $E \leq_\alpha^{<\mu} F$, When $\alpha > 0$, it is important to note that $\Phi^{((\boldsymbol{x})^{(\alpha)})}$ is required to equal $g(\boldsymbol{x})$; this allows more information in the oracle than it would if we had required $\Phi^{((x_0^{(\alpha)} \oplus x_1^{(\alpha)} \oplus \cdots)} = g(\boldsymbol{x})$, with the jumps of the individual inputs taken separately.

In our context for applying these notions, the domains S and T will be subsets of Cantor space, defined by

$$S = \{A \subseteq \omega : A \text{ codes the atomic diagram of a structure in } \mathfrak{C}\},$$

for some class $\mathfrak{C}$ of countable structures with domain ω, with T likewise defined by $\mathfrak{D}$. For us the equivalence relation on each of these domains will be isomorphism on the structures coded. One could explore further, of course, using elementary equivalence of those structures, or bi-embeddability, or other equivalence relations on structures.

3 Early Examples

To begin with, we consider the situation described in the introduction. The models of $\mathbf{ACF}_0$ form a particularly simple class of structures, with isomorphism equivalent to having the same transcendence degree (since we have restricted here to characteristic 0; similar remarks apply to any other fixed characteristic). Isomorphism between algebraic fields of characteristic 0 – that is, the subfields of $\overline{\mathbb{Q}}$ – seems a more challenging problem. However, analysis of computable models in these classes yields the opposite conclusion: isomorphism of computable models of $\mathbf{ACF}_0$ is Π^0_3-complete, whereas isomorphism of computable algebraic fields is only Π^0_2 (and is complete at this level). The latter remark follows from a lemma which appears as [13, Corollary 3.9].

Lemma 1. *Two algebraic field extensions E and F of $\mathbb{Q}$ are isomorphic if and only if every finitely generated subfield of each one embeds into the other.* □

By the Primitive Element Theorem, the condition here can be expressed by saying that, for every irreducible polynomial $q \in \mathbb{Q}[X]$, E possesses a root of q if and only if F does. For computable fields E and F, this is clearly a Π^0_2 condition.

When we broaden our analysis to the classes $\mathfrak{C}$ of all models of $\mathbf{ACF}_0$ with domain ω and $\mathfrak{D}$ of all algebraic field extensions of $\mathbb{Q}$ with domain ω, we gain a richer view of the situation. Write ACF and Alg for the sets of atomic diagrams of elements of $\mathfrak{C}$ and $\mathfrak{D}$, respectively, and $\cong_{ACF}$ and $\cong_{Alg}$ for the isomorphism relations on these sets of reals. First of all, it is clear that $\cong_{Alg} \not\leq_0 \cong_{ACF}$, as a full computable reduction would require every one of the continuum-many isomorphism classes in Alg to map to a distinct isomorphism class in ACF, and ACF has only countably many isomorphism classes in all. (To see that Alg has uncountably many, write p_n for the n-th prime and notice that for every $A \neq B \subseteq \omega$, the fields $\mathbb{Q}[\sqrt{p_n} \ : \ n \in A]$ and $\mathbb{Q}[\sqrt{p_n} \ : \ n \in B]$ cannot be isomorphic, as no finite set of square roots of primes generates the square root of any other prime.)

On the other hand, it is not difficult to give a binary computable reduction of Alg to ACF. Such a reduction is simply a Turing functional which, given the atomic diagrams of two algebraic fields F_0 and F_1, computes the diagrams of algebraically closed fields K_0 and K_1 as follows. Fix an enumeration $q_0, q_1, \ldots$ of all irreducible polynomials in $\mathbb{Q}[X]$. (We use here the fact that $\mathbb{Q}$ has a splitting algorithm, which was proven by Kronecker in [9].) At stage 0 we start with $\mathbb{Q}$ as K_0 and $\mathbb{Q}(t)$ as K_1 (with t transcendental).

Now for each s, compute the greatest number $n_s \leq s$ such that

$$(\forall n \leq n_s)(\forall i < \deg(q_n))\ [(\exists \text{ roots } x_0 < \ldots < x_i \leq s \text{ of } q_n(X) \text{ in } F_0) \\ \iff (\exists \text{ roots } y_0 < \ldots < y_i \leq s \text{ of } q_n(X) \text{ in } F_1)].$$

(Here $x_0 < \ldots < x_i \leq s$ refers to the order of the x_j in ω, not in F_0, which is not an ordered field. Hence this statement is decidable from the atomic diagrams of F_0 and F_1.) At stage $s+1$, if $(\forall t \leq s)\ n_t < n_{s+1}$, we adjoin a new element to each of K_0 and K_1, independent over all previous elements. If not, we adjoin no new independent elements. In either case, we also take one more step towards making K_0 and K_1 into models of $\mathbf{ACF}_0$.

At the end of this process, K_0 and K_1 will be models of $\mathbf{ACF}_0$. If $F_0 \cong F_1$, then new transcendentals were adjoined to each at infinitely many stages, so both have infinite transcendence degree, yielding $K_0 \cong K_1$ as desired. Otherwise, Lemma 1 yields some (least) n for which q_n has more roots in F_0 than in F_1 (without loss of generality). In this case, once all the roots in F_0 have appeared, n_s will never exceed n, and so no further transcendentals will ever again be added to either field. But at every finite stage, K_1 has larger transcendence degree than K_0, since we started that way at stage 0, and so $K_0 \ncong K_1$ as desired.

The process above can be converted into a countable reduction, yielding the next result.

Proposition 1. *Alg* $\leq_0^\omega$ *ACF*.

We sketch the construction of a countable computable reduction. Let d_n be the degree of the polynomial q_n. For each F, define the path $p_F \in \omega^\omega$ by

$$p_F(n) = |\{x \in F : q_n(x) = 0\}|.$$

Each such path is confined to the *possible nodes* satisfying $p_F(n) \leq d_n$ for all n, which form a finite-branching subtree. We assign numbers to these nodes: those at level 1 are numbered $0, \ldots, d_0$; those at the next level are numbered $d_0+1, d_0+2, \ldots, d_0+(d_0+1)(d_1+1)$, and so on. The only important aspect of this computable numbering is that each node has a label greater than its predecessor's label.

Given the atomic diagrams of algebraic fields $F_0, F_1, \ldots$, we construct models $K_0, K_1, \ldots$ of $\mathbf{ACF}_0$ to satisfy, for each i:

- If there exists $j < i$ such that $p_{F_i} = p_{F_j}$, then K_i has the same transcendence degree (over $\mathbb{Q}$) as K_j.
- Otherwise, there exists some least n such that $(\forall j < i)\ p_{F_i}\restriction n \neq p_{F_j}\restriction n$. Let d be the label of the node $p_F\restriction n$. Then K_i will have transcendence degree d.

Clearly, satisfying these conditions will ensure that we have a countable reduction from Alg to ACF. To satisfy them, we guess effectively at the path p_{F_i} for each i, at each stage s, with the guesses converging to the actual path p_{F_i}. If our guesses produce an n as described in the second item, then K_i at this stage has transcendence degree n; if not, then it is isomorphic to K_j at this stage.

This is not really a generalization of the binary reduction constructed above. Here we use the fact that transcendentals can be destroyed as well as created in the construction of a computable field: we build only finitely much of K_i at any stage, and therefore any element previously considered transcendental in K_i can consistently be turned into a large rational number at the next stage. (The 1-type of a transcendental element is a nonprincipal type.) For a given K_i, once the guesses stabilize on the true value n (if one exists), the transcendentals in K_i at that stage remain independent forever, and any more transcendentals subsequently added to K_i are later destroyed this way. On the other hand, if $F_i \cong F_j$ for some smaller j, then for the n belonging to the least such j, there is some stage after which we always have the same guess $p_{F_i} \upharpoonright n = p_{F_j} \upharpoonright n$ for both paths. From then on, the independent elements in K_i corresponding to those in F_j will stay independent forever, and any subsequent ones will later be destroyed. (Notice that this means that every K_i will have finite transcendence degree. So we have actually given a countable reduction of Alg to a slightly smaller class than ACF.)

On the other hand, there is no computable reduction, not even a binary reduction, from ACF to Alg. This follows from the Π^0_3-completeness of the isomorphism problem for the set of indices for computable algebraically closed fields of characteristic 0: any such reduction would show this Π^0_3-complete set to be Π^0_2, by Lemma 1. Thus the non-reducibility result for computable structures carries over to the general case.

4 $\mathbf{ACF_0}$ and Equivalence Structures

For further insights about $\mathbf{ACF}_0$, we consider another class of structures: the class $\mathfrak{E}$ of countable equivalence structures with no infinite equivalence classes. (An *equivalence structure* consists of a single equivalence relation R on the domain, with equality also in the language.) For computable members of $\mathfrak{E}$, the isomorphism problem is Π^0_3-complete, the same level of complexity as for $\mathbf{ACF}_0$. (If we had allowed infinite equivalence classes, the complexity level would be Π^0_4 instead.) However, there are 2^ω-many nonisomorphic structures in $\mathfrak{E}$. Our goal is to distinguish these two classes using the new notions of this article. First, we show that the class $\mathfrak{C}$ of countable models of $\mathbf{ACF}_0$ is no harder than $\mathfrak{E}$.

Proposition 2. *$ACF \leq_0 Eq$, where Eq is the isomorphism relation on the reals in the class $\mathfrak{E}$.*

Proof. Given an algebraically closed field K as oracle, our reduction Φ builds an equivalence relation R, beginning by creating a single R-equivalence class of size $(2n-1)$ and infinitely many R-equivalence classes of size $2n$ for each $n > 0$.

Then it begins to guess (separately for each $n > 0$) whether K has transcendence degree $\geq n$.

At each stage $s+1$, for each $n \leq s$, we find the least n-tuple $\boldsymbol{x} \in K$ with all $x_i \leq s$ such that, for all $i \leq s$, $q_{n,i}(\boldsymbol{x}) \neq 0$ in K. (Here we use a fixed ordering of ω^n and a fixed list $\{q_{n,i}\}_{i\in\omega}$ of $\mathbb{Q}[X_1, \ldots, X_n]$.) If this is the same tuple as at stage s, we do nothing. If it is a new tuple, then we take the unique equivalence class of size $2n-1$ currently in R, add one more element to it, and create a new R-equivalence class of size $(2n-1)$ to replace it. This is the entire construction.

Now if K has transcendence degree $< n$, then every R-class of size $(2n-1)$ ever created will eventually become a class of size $2n$. On the other hand, if K has transcendence degree $\geq n$, then eventually an independent n-tuple will be found in K, and from that stage on, the unique R-class of size $(2n-1)$ will never have another element added to it. Thus R has exactly one class of size $(2n-1)$ for each $n \leq$ the transcendence degree of K, along with infinitely many classes of each even size. Thus we have a computable full reduction from ACF to Eq. $\square$

The next proposition is also not surprising. In fact, given that isomorphism on computable models of $\mathbf{ACF}_0$ is Π^0_3-complete as a set (and that isomorphism on computable sturctures in $\mathfrak{E}$ is Π^0_3, the proposition holds immediately for computable structures. This is because a computable binary reduction from E to F (where these are equivalence relations on ω) is in fact simply a many-one reduction from the set E to the set F. Since we wish to establish it for all of $\mathfrak{E}$ and $\mathfrak{C}$, rather than just for computable structures, we give the entire proof.

Proposition 3. *$Eq \leq^2_0 ACF$. That is, there is a computable binary reduction from Eq to ACF.*

Proof. Given two equivalence relations R_0 and R_1, we must produce algebraically closed fields K_0 and K_1, isomorphic if and only if R_0 and R_1 are. To do so, we wish to test, for each k and n, whether

$$R_0 \text{ has} \geq k \text{ classes of size exactly } n \iff R_1 \text{ has} \geq k \text{ classes of size exactly } n.$$

A pair $\langle k, n\rangle$ for which this may fail will be assigned a number $t_{n,k}$. If R_0 turns out to have k classes of size n while R_1 does not, then K_0 will have transcendence degree $\geq t_{k,n}$ and K_1 will not. If no such k and n exist, then both fields will have infinite transcendence degree. To determine what to do, we will search first for a finite subset $X_{k,n}$ forming k-many R_0-classes of size n, and then for a corresponding subset $\tilde{Y}_{k,n}$ of R_1-classes. If the R_1-classes appear first, then they will form $Y_{k,n}$ and we will then search for $\tilde{X}_{k,n}$ instead.

At stage 0 we begin with $\mathbb{Q}$ as both K_0 and K_1. At each stage, finitely many steps are taken so that each field will be algebraically closed at the end of the construction, but at no stage will $\overline{\mathbb{Q}}$ yet be a subfield of either K_0 or K_1. Thus, putative transcendentals can always be "destroyed," by being turned into elements algebraic over $\mathbb{Q}$. We write $R_{i,s}$ for the restriction of R_i to $\{0, \ldots, s\}$.

At stage $s+1$ we first address that pair $\langle k, n\rangle \leq s$ for which $t_{k,n,s}$ is smallest; then that for which $t_{k,n,s}$ is second-smallest, and so on. If none of these steps ends the stage, we will subsequently address those $\langle k, n\rangle$ with $t_{k,n,s}$ undefined.

If $t_{k,n,s}$ is defined, then so is one (but not both) of the finite subsets $X_{k,n,s}$ ($\subseteq R_{0,s}$) or $Y_{k,n,s}$ ($\subseteq R_{1,s}$). The instructions are symmetric; we give them here with $X_{k,n,s}$ defined, in which case its elements formed k distinct R_0-classes of size n when it was first chosen.

1. If any of these k R_0-classes contains more than n elements in $R_{0,s+1}$, then $X_{k,n,s+1}$ and $t_{k,n,s+1}$ become undefined, and we destroy enough transcendentals in both K_0 and K_1 to ensure that both have transcendence degree $t_{k',n',s+1}$, for that pair $\langle k', n'\rangle$ with the greatest $t_{k',n',s+1} < t_{k,n,s}$ (If there is no such $\langle k', n'\rangle$, then K_0 and K_1 both get transcendence degree 0.) The stage ends here, with all remaining values becoming undefined as well. Otherwise, $X_{k,n,s+1} = X_{k,n,s}$ and K_0 remains the same, and we consider (2)-(5) below.
2. Otherwise, if $\tilde{Y}_{k,n,s}$ was undefined, and $R_{1,s+1}$ contains k distinct classes of size exactly n, then the elements of the first k of these classes are defined to form $\tilde{Y}_{k,n,s+1}$, and we add just as many transcendentals to K_1 as needed to make its transcendence degree $\geq t_{k,n,s}$.
3. If $\tilde{Y}_{k,n,s}$ was undefined, but (2) does not apply, then nothing changes.
4. If $\tilde{Y}_{k,n,s}$ was defined, then its elements formed k distinct R_1-classes. If all these classes still have size exactly n in $R_{1,s+1}$, then nothing changes.
5. Otherwise $\tilde{Y}_{k,n,s}$ was defined, but one of its R_1-classes now has size $> n$. In this case $\tilde{Y}_{k,n,s+1}$ is undefined, and we destroy just enough transcendentals in K_1 to make its transcendence degree $< t_{k,n,s+1} = t_{k,n,s}$. ($K_0$ still has trancendence degree $\geq t_{k,n,s}$.)

If either (1), (2), (3), or (5) applies, then the stage ends here. If (4) applies, then we continue to the pair $\langle k, n\rangle$ with the next-smallest $t_{k,n,s}$. If no more pairs have $t_{k,n,s}$ defined, then we now go in order through those pairs $\langle k, n\rangle \leq s$ for which $t_{k,n,s}$ is undefined. For the least pair $\langle k, n\rangle$ (if any) among these such that either $R_{0,s+1}$ or $R_{1,s+1}$ contains at least k distinct classes of size exactly n, we define $t_{k,n,s+1} = s + 1$, and either

- let $X_{k,n,s+1}$ contain the (kn) elements of $R_{0,s+1}$ forming those R_0-classes, and add transcendentals to K_0 so that it has transcendence degree $t_{k,n,s+1}$; or else
- let $Y_{k,n,s+1}$ contain the (kn) elements of $R_{1,s+1}$ forming those R_1-classes, and add transcendentals to K_1 so that it has transcendence degree $t_{k,n,s+1}$.

This completes the stage.

If $R_0 \ncong R_1$, then fix the least stage s_0 at which, for some $\langle k, n\rangle$, we have found k classes truly of size n in R_{0,s_0} (WLOG) and set them to equal X_{k,n,s_0}, and R_1 does not possess k classes of this this size, and for all $\langle k', n'\rangle$ with $t_{k',n',s_0} < t_{k,n,s_0}$, X_{k',n',s_0} and $\tilde{Y}_{k',n',s_0}$ are defined and have stabilized. Such an s_0 must exist, since some $\langle k, n\rangle$ do exist. At this stage, K_0 will be given transcendence degree t_{k,n,s_0}, which will equal $t_{k,n} = \lim_s t_{k,n,s}$, while K_1 will have lesser transcendence degree at that stage. Moreover, every $\tilde{Y}_{k,n,s}$ ever subsequently found will later become undefined, with the transcendence degree of K_1 threrefore dropping back below $t_{k,n}$ infinitely often, and so $K_1 \ncong K_0$.

However, if $R_0 \cong R_1$, then for every $\langle k, n\rangle$ for which $t_{k,n,s}$ stabilizes, both K_0 and K_1 will have transcendence degree $\geq \lim_s t_{k,n,s}$. Moreover, there will be infinitely many such pairs $\langle k, n\rangle$, since every element of R_0 lies in a finite R_0-class, and likewise for R_1. Therefore, both K_0 and K_1 will have infinite transcendence degree, leaving them isomorphic. $\square$

Proposition 4. *$Eq \leq_0^3 ACF$, but $Eq \not\leq_0^4 ACF$. That is, there is a computable ternary reduction from Eq to ACF, but no 4-ary computable reduction.*

The details of the proof are too extensive to present here; they will be described in the author's talk at the C.i.E. meeting.

In light of the Π_3^0-completeness of isomorphism for computable algebraic fields, this proposition seems like a surprise. Section 4.2 of [11] makes it more plausible. The discussion there centers on the fact that, since isomorphism on models of $\mathbf{ACF}_0$ is given by transcendence degree, it is essentially just a matter (for a computable model K) of counting the elements in the following Σ_2^0 basis for K:

$$\{x \in K : (\forall \text{ nonzero } h \in \mathbb{Z}[X_0, \ldots, X_x])\ h(0, 1, \ldots, x) \neq 0 \text{ in } K\}.$$

It is shown in [11] that, if $E_{\text{card}}^{\emptyset'}$ is the relation (on indices e of Σ_2^0 sets $W_e^{\emptyset'}$) of having the same cardinality, then $E_{\text{card}}^{\emptyset'}$ is complete under ternary reducibility among Π_3^0 equivalence relations on ω, but not complete among them under 4-ary reducibility. Since the relation (for indices of computable fields in general) of having the same transcendence degree over the prime subfield is computably reducible to $E_{\text{card}}^{\emptyset'}$, it is not so surprising that isomorphism on ACF in general loses its power at the same specific finitary level of reduction.

The arguments in [11] do prove the following, using the notion of *jump-reduction* from [10], with a functional whose oracle is the jump of the inputs.

Lemma 2. *$Eq \leq_1^3 ACF$. That is, there is a Turing functional Γ such that $\Gamma^{(E_0 \oplus E_1 \oplus E_2)'} = K_0 \oplus K_1 \oplus K_2$ is a ternary reduction from Eq to ACF.*

In light of Proposition 4, it is natural to enquire into other theories admitting similar notions of dimension. Baldwin and Lachlan showed in [1] that, if T is an ω_1-categorical theory that is not ω-categorical (such as $\mathbf{ACF}_0$), the countable models of T form an $(\omega{+}1)$-sequence under elementary embedding. One suspects, therefore, that the class of such models might be complete among Π_α^0-definable equivalence relations on 2^ω under ternary computable reducibility but not under 4-ary computable reducibility, just as holds for models of $\mathbf{ACF}_0$ with $\alpha = 3$.

5 Transformations and Functors

As a final remark, we note that these computable transformations, as in Definition 1, form part of the larger concept of a *computable functor*. These were defined by Poonen, Schoutens, Shlapentokh, and the author in [12], and subsequently, in [6, 7], he and Harrison-Trainor, Melnikov, and Montalbán broadened their applicability. We give their definition here.

Definition 3. *Let $\mathfrak{C}$ and $\mathfrak{D}$ be categories of structures with domain ω, for which the morphisms from $\mathcal{S}$ to $\mathcal{T}$ are maps from the domain ω of $\mathcal{S}$ to the domain ω of $\mathcal{T}$. A* computable functor *is a functor $\mathcal{F} : \mathfrak{C} \to \mathfrak{D}$ for which there exist Turing functionals Φ and Φ_* such that*

- *for every $\mathcal{S} \in \mathfrak{C}$, the function $\Phi^{\mathcal{S}}$ computes (the atomic diagram of) the structure $\mathcal{F}(\mathcal{S})$; and*
- *for every morphism $g : \mathcal{S} \to \mathcal{T}$ in $\mathfrak{C}$, we have $\Phi_*^{\mathcal{S} \oplus g \oplus \mathcal{T}} = \mathcal{F}(g)$ in $\mathfrak{D}$.*

It would be natural to examine how close the computable transformations defined earlier in this article come to being computable as functors. The functional Φ_* computing the functor on morphisms is likely to require one or more jumps of $\mathcal{S}$ and $\mathcal{T}$ as oracle, but if not, then the conclusions in [12] and [6] about computable functors would all apply here as well.

References

1. J.T. Baldwin & A.H. Lachlan; On strongly minimal sets, *Journal of Symbolic Logic* **36** 1 (1971), 79–96.
2. W. Calvert; The isomorphism problem for classes of computable fields, *Archive for Mathematical Logic* **43** (2004), 327–336.
3. W. Calvert, D. Cummins, J.F. Knight, & S. Miller; Comparing classes of finite structures, *Algebra and Logic* **43** (2004), 365–373.
4. W. Calvert & J.F. Knight; Classification from a computable viewpoint, *Bulletin of Symbolic Logic* **12** (2006), 191–218.
5. H. Friedman & L. Stanley; A Borel reducibility for classes of countable structures. *Journal of Symbolic Logic* **54** (1989), 894–914.
6. M. Harrison-Trainor, A. Melnikov, R. Miller, & A. Montalbán; Computable functors and effective interpretability, to appear in the *Journal of Symbolic Logic*.
7. M. Harrison-Trainor, R. Miller, & A. Montalbán; Borel functors and infinitary interpretations, submitted for publication.
8. J.F. Knight, S. Miller, & M. Vanden Boom; Turing computable embeddings. *Journal of Symbolic Logic* **72** 3 (2007), 901–918.
9. L. Kronecker, Grundzüge einer arithmetischen Theorie der algebraischen Größen, *J. f. Math.* **92** (1882), 1–122.
10. R. Miller; Computable reducibility for Cantor space, submitted for publication.
11. R. Miller & K.M. Ng; Finitary reducibility on equivalence relations, *Journal of Symbolic Logic* **81** (2016) 4, 1225–1254.
12. R. Miller, B. Poonen, H. Schoutens, & A. Shlapentokh; A computable functor from graphs to fields, submitted for publication.
13. R. Miller & A. Shlapentokh; Computable Categoricity for Algebraic Fields with Splitting Algorithms, *Trans. of the Amer. Math. Soc.* **367** 6 (2015), 3981–4017.