Computable Analysis and Classification Problems

Rodney G. Downey[1] and Alexander G. Melnikov[2(✉)]

[1] Victoria University of Wellington, Wellington, New Zealand
`rod.downey@vuw.ac.nz`
[2] Massey University, Auckland, New Zealand
`alexander.g.melnikov@gmail.com`

1 Introduction

Suppose we are given a collection of mathematical objects such as the class of connected compact Polish groups or the set of all real numbers which are normal to some base. Is there a *reasonable* classification of these objects (e.g. by invariants)? One nice property expected from a useful classification is that it makes the classified objects easier to handle *algorithmically*. Even if the general classification of a class of objects is impossible, sometimes there is a useful *hierarchy* of objects in this class, e.g., the Cantor-Bendixson rank of a scattered Polish space, the Ulm type of a reduced abelian p-group, etc. We would expect that the algorithmic, algebraic, or topological complexity of objects increase from lower to higher levels of a hierarchy. Can you make this intuition formal? Is it possible to formally *measure* the algorithmic complexity of a given classification or a hierarchy, or perhaps show that there is no reasonable classification at all? Can algorithmic tools help us to define useful hierarchies?

In the present article we discuss several recent works in computable analysis which are related to classification. The main underlying theme here is applying computability theory to a classification problem or a hierarchy; neither the problem nor the hierarchy has to be computability-theoretic. The discussed results can be informally split into three categories:

I. *Local effective classifications.* Given some classical metric or Banach space we ask which elements of the space satisfy a certain property. For example: Which real numbers are normal? Which continuous functions are regular? And so on.

II. *Global effective classifications.* How hard is it to classify, say, compact Polish groups? What about compact Polish spaces? To answer these questions formally we extend methods of computable structure theory to computable separable spaces.

III. *Applications of computability to hierarchies.* Usually elements of higher levels of a given hierarchy are expected to be more complex. Computability theory can be used to make this intuition formal. Such results are clearly related to (I), but there are also explicit connections with Weihrauch reducibility and with theme II.

These three themes above are closely related and no firm line can be drawn between any two of them. We begin our discussion with the second global scheme (II) and a certain related hierarchy. Then we discuss the local theme (I) and finish with more on local hierarchies (III).

2   Global Effective Classifications

There has been a lot of applications of computability theoretic techniques to classification problems in *countable* algebra; see [19,31]. A few years ago Melnikov [26] proposed that methods of computable structure theory can be extended to computable separable spaces. We will see that some of these methods can be used to measure the classification problems for classes of separable spaces. We first look at the simpler case of Polish metric spaces, and then we discuss Banach spaces and Polish groups.

Computable Structures. To apply methods of computable structure theory we need to make the standard notion of a computable Polish space [5,34,37] look more familiar to a computable algebraist. A *structure* on a Polish space $\mathcal{M} = (M, d)$ is any map $\alpha : \omega \to \mathcal{M}$ such that its range is a dense subset of $\mathcal{M}$. The *open diagram* $D^+(\alpha)$ of a structure $\alpha : \omega \to \mathcal{M}$ is the collection of Gödel numbers of all elementary facts of the form $d(\alpha(i), \alpha(j)) < r$ and $d(\alpha(i), \alpha(j)) > q$ which hold on $\mathcal{M}$, where i, j range over ω and q, r over $\mathbb{Q}$. We say that α is *computable* if $D^+(\alpha)$ is a computably enumerable set[1].

We return to classification. Fix some class $\mathcal{K}$ of Polish spaces. First, suppose our task is to classify *computable* members of $\mathcal{K}$. Fix a uniformly computable list of all partially computable structures on Polish spaces: $\alpha_0, \alpha_1, \alpha_2, \ldots$, where each computable structure α_i is identified with its c.e. open diagram $D^+(\alpha_i)$.

Definition 1. The complexity of the classification problem for computable members of $\mathcal{K}$ is measured using the following two *index sets*:

1. The characterisation problem for computable members of $\mathcal{K}$:

$$I(\mathcal{K}) = \{e : \alpha_e \in \mathcal{K}\}.$$

2. The isomorphism problem for computable members of $\mathcal{K}$:

$$E(\mathcal{K}) = \{(e, j) : e, j \in I(\mathcal{K}) \text{ and } cl(\alpha_e) \cong cl(\alpha_j)\},$$

where $\cong$ stands for isometric isomorphism. Computable members of a class $\mathcal{K}$ of Polish metric spaces admit an *effective classification* (up to isometric isomorphim) if both $I(\mathcal{K})$ and $E(\mathcal{K})$ are hyperarithmetic.

---

[1] A standard trick can be used to (computably) remove repetitions and replace α with an injective α'; see, e.g., [20]. Thus, without loss of generality the map $\alpha : \omega \to \mathcal{M}$ will be assumed injective.

Remark 2. Note that ≅ does *not* stand for *computable* isometric isomorphism. Of course, if we wish to classify our spaces up to homeomorphism or quasi-isometry (etc.) then we should adjust the interpretation of ≅ accordingly. Definition 1 is an adaptation of a similar method proposed in [19] for countable discrete computable structures. Cenzer and Remmel [10] used index sets to measure the complexity of various properties of Π_1^0-classes. Although there is not much in common between their results and the theorems discussed in the present article, it is important that index sets had been used in computable analysis long before us.

Relativisation. To extend these methods beyond computable Polish spaces we use relativisation. A structure $\alpha : \omega \to \mathcal{M}$ is computable relative to X if $D^+(\alpha)$ is computably enumerable relative to X. Using structures computable relative to an oracle X, relativise the definitions of $I(\mathcal{K})$ and $E(\mathcal{K})$ to X; the resulting sets will be denoted $I^X(\mathcal{K})$ and $E^X(\mathcal{K})$. Usually, when we establish that, say, $I(\mathcal{K}) \in \Sigma_n^0$, we can apply relativisation to show $I^X(\mathcal{K}) \in \Sigma_n^X$, and similarly for $E^X(K)$. Recall this means that, for some recursive scheme R, we have

$$i \in I^X(\mathcal{K}) \iff (\exists x_1)\dots(Q_n x_n)R(X; x_1, \dots, x_n, i),$$

where X is a set-parameter. It follows that the number of alternations of quantifiers in $(\exists x_1)\dots(Q_n x_n)R(X; x_1, \dots, x_n, i)$ is an *invariant* of the whole class. This motivates the following:

Definition 3. We say that that $\mathcal{K}$ *admits an effective classification* if, for every oracle X, both $I^X(\mathcal{K})$ and $E^X(\mathcal{K})$ belong to a some (fixed) level of the hyper-arithmetical hierarchy relativised to X.

The First Results. Perhaps, the first non-trivial illustration of the proposed approach to classification in the literature is the theorem below.

Theorem 4 (Melnikov and Nies [29]). *The class $\mathcal{K}_{comp}$ of compact Polish metric spaces admits an effective classification.*

The proof of Theorem 4 relies on ideas of of Gromov [21] and $\mathcal{L}_{\omega_1\omega}$-definability. The most important step of the proof is establishing that every computable compact Polish metric space is Δ_3^0-*categorical*; the categoricity hierarchy will be discussed in the next subsection. Theorem 4 contrasts with:

Theorem 5 (Nies and Solecki [33]). *The characterisation problem for computable locally compact Polish metric spaces is Π_1^1-complete.*

It follows that for a given Polish space $\mathcal{M}$, deciding if it is locally compact is as hard as just checking the Π_1^1 definition of local compactness. Thus, the class of locally compact Polish metric spaces does not admit an effective classification. The result formally confirms our intuition that locally compact spaces are very hard to classify.

The Categoricity Hierarchy. We use the standard notion of a computable map between computable Polish spaces; see, e.g., [26]. The definition below extends the classical notion of computable categoricity [1] to Polish spaces.

Definition 6 ([26]). A Polish space $\mathcal{M}$ is computably categorical if for any two computable structures α and β on $\mathcal{M}$ there is an isometric isomorphism between $cl(\alpha)$ and $cl(\beta)$ computable with respect to α and β[2].

Examples of computably categorical spaces include [26]: every Polish space associated to a separable Hilbert space, Cantor space with the usual ultrametric, and the Urysohn space. If we generalise Definition 6 by allowing the isomorphism to be Δ_α^0, the resulting notion of a Δ_α^0-categorical space is a direct adaptation of the classical notion of Δ_α^0-*categoricity* from computable algebra [1].

How is this technical notion related to classification? If every member of $\mathcal{K}$ is Δ_α^0-categorical and $I(\mathcal{K})$ is hyperarithmetical then $E(\mathcal{K})$ is hyperarithmetical too[3]. Our results tend to be easily relativizable to any oracle X, and thus both $I^X(\mathcal{K})$ and $E^X(\mathcal{K})$ will usually be hyperarithmetical relative to X as well. This approach was used by Melnikov and Nies [29] to prove Theorem 4. Clearly, Δ_α^0-categoricity induces a hierarchy, and using a transformation from graphs to Polish spaces [20] it is not hard to show that the hierarchy is proper in general.

We will return to discussing Δ_α^0-categorcity later. First we discuss another natural relativisation of Definition 6. It reveals a connection between first-order definability and computable categoricity of spaces, in the spirit of [1]. We say that a computable Polish space $\mathcal{M} = cl(\alpha_i)_{i\in\omega}$ is *relatively computably categorical* if any (not necessarily computable) structure $(\beta_i)_{i\in\omega}$ computes an isometry from $cl(\beta_i)_{i\in\omega}$ onto $cl(\alpha_i)_{i\in\omega}$. Greenberg, Knight, Melnikov, and Turetsky [20] showed:

Theorem 7. *A computable Polish metric space $\mathcal{M}$ is relatively computably categorical iff for every $\epsilon \in \mathbb{Q}$ it admits a c.e. ϵ-Scott family consisting of first-order positive $\exists$-formulae[4].*

It is crucial in the proof of Theorem 7 that the positive $\exists$-formulae define open sets.

Problem 8. Extend Theorem 7 to relative Δ_α^0-categoricity. For that, an adequate formal definition of relative Δ_α^0-categoricity must be designed.

Banach spaces. All Banach spaces in this section are separable. What is the most natural general approach to computability on Banach spaces? Pour El and Richards [34] restrict themselves only to computable structures which also compute the standard vector space operations. The well-known result of Mazur

---

[2] Here cl stands for the completion operator. Recall that the inverse of a computable surjective isometry is computable too, thus the notion is symmetric.

[3] Since an isometric image of a complete metric space is closed, saying that a Δ_α^0 map is "onto" is arithmetical.

[4] This means that ϵ-automorphism orbits of tuples in $\mathcal{M}$ can be described by a (uniformly in ϵ) c.e. family of positive first-order existential positive formulae in the language $(d_{<r}, d_{>r})_{r\in\mathbb{Q}}$ with finitely many "stable" parameters; a parameter is stable if there exist a ball such that any point from the ball can be used as a parameter without changing the formula.

and Ulam (can be found in [35]) implies that there is only one way to define the operations consistently with a given norm so that we get a complete normed space. Does the result of Mazur and Ulam hold computably? If the answer was "yes" then the definition from Pour El and Richards [34] would be an overkill. Interestingly, Melnikov and Ng [30] showed that the effective version of Mazur-Ulam fails for the space of continuous functions on the unit interval.

Problem 9. Give an optimal effective analysis of Mazur-Ulam.

In the context of the present article, the above-mentioned result of Melnikov and Ng justifies the use of the definition below which is of course equivalent to the approach in Pour-El and Richards [34].

Definition 10. A separable Banach space $\mathcal{B}$ is computable if the associated metric space admits a computable structure which computes the vector space operations on $\mathcal{B}$.

In the case of Banach spaces (over $\mathbb{R}$) we modify Definitions 1 and 3 using the natural enumeration of all (partially) computable countable normed spaces $B_0, B_1, \ldots$. The relativisation principle still applies to this approach. To save space we omit the definition; see [6]. Now to the results.

Lebesgue Spaces. Suppose we are given a computable Banach space $\mathcal{B}$. How hard is it to determine whether $\mathcal{B}$ is a Lebesgue space? In other words, what is the complexity of the characterisation problem (Definition 1) for Lebesgue spaces? The crude upper bound involves searching for a (separable) measure space Ω and a real p which makes the space look like $L^p(\Omega)$. The well-known Kakutani–Bohnenblus characterisation of Lebesgue spaces in terms of Banach lattices [4,22] does not seem to be of much help either. Brown, McNicholl and Melnikov [6] have proven the following rather surprising result:

Theorem 11 (Brown et al. [6]). *The characterisation problem for Lebesgue spaces is Π_3^0.*

How to reduce the unclassifiable brute-force complexity down to Π_3^0? Using a non-trivial and novel technique, McNicholl [24] proved that for a computable real p, ℓ^p is Δ_2^0-categorical[5]. The proof of Theorem 11 extends these techniques to arbitrary Lebesgue spaces and combines them with new ideas. We see that Δ_α^0-categoricity helps again, though indirectly.

Question 12 ([6]). Is the bound Π_3^0 from Theorem 11 tight?

The main obstacle in simplifying the upper bound is related to:

---

[5] In the case of Banach spaces, we of course must restrict Definition 6 to computable structures which also compute the vector space operations. See [8,12] for further results on categoricity of Lebesgue spaces.

Question 13 (McNicholl). Suppose $\mathcal{B} \cong L^p(\Omega)$ is computable. Does p have to be a computable real? If yes, is it *uniformly* computable[6]?

It also follows from the main results of [6] that the isomorphism problem for computable Lebesgue spaces is arithmetical (see Remark 17). Although we do not know if the upper bounds are tight if p is not held fixed, the results are relativizable to any oracle. Thus, we have:

Corollary 14. *Separable Lebesgue spaces are effectively classifyable.*

Question 15. Estimate the complexity of the effective classification problem for: (1) separable Hardy spaces, and (2) separable Sobolev spaces.

Question 16. For each n, is there Δ^0_{n+1}-categorical but not Δ^0_n-categorical Banach space? Same for Polish groups[7].

A Computable Characterisation of C[0,1]. Suppose we are given a description of a (separable) Banach space $\mathcal{B}$. How hard is it to determine whether it is isomorphic to some fixed Banach space $\mathcal{C}$? Within the proposed framework, we can set $\mathcal{K} = \{\mathcal{C}\}$ and measure the complexity of the effective characterisation problem $\{e : cl(B_e) \cong \mathcal{C}\}$. For example, the separable Hilbert space ℓ_2 admits a low level arithmetical characterisation: use the parallelogram law and compute a basis [34]. Also, various natural Lebesgue spaces such as ℓ_3 admit arithmetical characterisations, with some index sets complete at proper difference levels such as d-Σ^0_2 [6].

Remark 17. In effective algebra there are very few natural examples of index sets of structures which are *not* complete at some level of the hyperarithmetical hierarchy; see, e.g., Problem 1 in [19]. Nonetheless, such estimates seem to be more common in computable analysis. For example, suppose $p \geq 1$ is a computable real other than 2. Then the isomorphism problem for the class of L^p spaces is co-3-Σ^0_3-complete [6].

Recall that $C[0,1]$ is universal among all separable Banach spaces. Building on the earlier work in [30] and [7], Franklin et al. [17] have recently announced the following unexpected result:

Theorem 18 (Franklin et al. [17]). *The Banach space $C[0,1]$ admits an effective (arithmetical) characterisation.*

Again, the result can be relativised to any oracle, but here it is not that important since $C[0,1]$ is computable. The main technical lemma in the proof states that $C[0,1]$ is Δ^0_5-categorical.

Question 19 Calculate optimal bounds for the characterisation problem and Δ^0_n-categoricity of $C[0,1]$.

---

[6] Currently, the best known uniform upper bound is Δ^0_2 [6]. McNicholl [25] has recently announced a partial positive solution for the case when $p > 2$, but he also announced that the uniformity of computing p fails. Thus, we conjecture that the upper bound in Theorem 11 is tight.

[7] Such examples exist for Polish spaces, as follows from [20].

Polish Groups. Following Melnikov and Montalbàn [28], we say that a (metrized) Polish group is computable if it admits a computable structure that computes the standard $\cdot$ and $^{-1}$ on the group[8]. Fix a uniform enumeration $(G_e)_{e \in \omega}$ of all partially computable Polish groups. The definitions of the characterisation problem and the isomorphism problem (Definition 1) should be adjusted accordingly. We note however that this approach seems best suited for *compact* groups. This is because compactness and totality of a (potential) group operation are both low-level arithmetical properties [27]. Thus, in the compact case the index sets $I(\mathcal{K})$ and $E(\mathcal{K})$ will reflect the complexity of $\mathcal{K}$ rather than some pathologies of coding.

Recall that compact Polish spaces admit an effective classification (Theorem 4). How hard is it to classify compact Polish groups? Every compact Polish group G contains the largest connected subgroup H which makes G/H profinite. Thus, the classes of connected and profinite groups are central to the general theory of Polish groups.

Theorem 20 (Melnikov [27]).

1. *The characterisation problems for profinite and connected compact Polish groups are both arithmetical.*
2. *The topological isomorphism problems for profinite abelian groups and for connected compact abelian groups are both Σ_1^1-complete.*

As usual, the result can be relativised to any oracle. It follows that recognising whether a given group is profinite or connected compact is not that hard (Π_2^0- and Π_3^0-complete, resp.), but the isomorphism problem is too hard even in the abelian case. In contrast with the previous results, the main tool in the proof of Thm 20 is not Δ_α^0-categoricity[9]. Instead, Melnikov proves a computable version of the celebrated Pontryagin duality which is then used to apply effective algebraic results [14,16] to topological groups. Provably, the duality is effective only when passing from discrete to compact groups [27], but this half-effectivity was sufficient.

Problem 21. (1.) Study Δ_α^0-categoricity of Polish groups. (2.) Measure the effective classification problem for natural subclasses of compact Polish groups. (3.) What is the complexity of the Pontryagin dual of a computable compact connected abelian group? (The profinite case is known.)

---

[8] Computability of the metric is obviously not enough. Unlike the Banach space case, there is nothing like the Mazur-Ulam theorem for topological groups. For example, every separable profinite group is homeomorphic to the computable Cantor space, but obviously not every profinite group is computable. Perhaps, computability of $^{-1}$ can be dropped at least in some cases, but this has not yet been explored.

[9] But there is again a tight connection with categoricity. Melnikov [27] showed that the characterisation problem for computably categorical *recursive* profinite abelian groups Π_4^0-complete; see [27] for the definitions.

A Local Approach to Global Classification. We must emphasise that the subdivision of the discussed results into local and global is somewhat subjective. For instance, Melnikov and Montalbán [28] have suggested an intermediate approach. Recall that a transformation space is a Polish space together with a smooth action of a Polish group on the space. The key observation is that the standard S_∞-transformation space of countable structures in a given finite language is actually *computable* [28]. Thus, the global classification problem for countable structures becomes a local classification problem for points in a transformation space.

The use of topological groups sometimes makes proofs easier. For instance, a technical result of Montalbán [32] characterises uniformly computably categorical structures (categorical relative to all oracles) in terms of infinitary Scott sentences. Using transformation groups and the old result of Effros [15], the rather tricky argument from [32] can be simplified and generalised at the same time:

Theorem 22. *Let (X, G, a) be a (not necessarily computable) transformation group, and $x \in X$. The following are equivalent:*

1. x is uniformly computably categorical on a cone;
2. the orbit of x is $\mathbf{\Pi}_2^0$.

The theorem above is an explicit link between Polish groups, the Borel hierarchy, the categoricity hierarchy, and classification problems.

Question 23. Is it possible to extend Theorem 22 to a (computable) categoricity-theoretic description of $\mathbf{\Sigma}_3^0$-orbits in a transformation space[10]?

3   Local Classifications and Hierarchies of Functions

Suppose $\mathcal{M}$ is a computable Polish space, such as $C[0,1]$ or the reals[11]. Fix some property P of points in $\mathcal{M}$, and consider the index set $I(P) = \{e : x_e \text{ satisfies } P\}$ of P, where $(x_e)_{e \in \omega}$ is an effective enumeration of all (partially) computable points in $\mathcal{M}$. If the complexity of $I(P)$ is no simpler than the naive brute-force upper bound derived from the definition of P, then we can conjecture that members of the space having property P do not admit any reasonable classification. Such results are usually relativisable to any oracle.

For example, Becher, Heiber, and Slaman [3] showed that the index set of all computable real numbers normal to base 2 is Π_3^0-complete[12]. Their proof is

---

[10] Melnikov and Montalbán's initial strong conjecture was that such a characterisation in terms of non-uniform categoricity exists; alas, their proof contained an error (spotted by Solecki).

[11] We usually fix a "natural" computable structure on the space, such as the rationals in $\mathbb{R}$. For a computably categorical space (such as $\mathbb{R}$) this assumption is of course not necessary, while for spaces like $C[0,1]$ it is essential [30].

[12] A number is said to be normal to base b if, for every positive integer n, all possible strings n digits long have density b^{-n}. A number is (absolutely) normal if it is normal to base b for every integer b greater than 1. Equivalently, a sequence is normal if and only if there is no finite-state gambler that succeeds on it [36].

relativizable, and thus implies the earlier result of Ki and Linton [23] who showed that the set of reals normal to base 2 is $\mathbf{\Pi}_3^0$-complete. Becher and Slaman [9] later extended their techniques to show that the index set of numbers computable to *some* base is Σ_4^0-complete, and again the result can be fully relativised to any oracle.

Westrick [38] gives a detailed index set analysis of Kechris and Woodin's differentiability hierarchy for continuous functions in $C[0,1]$. She showed that the index set of rank $\leq \alpha$ computable functions in $C[0,1]$ is $\Pi_{2\alpha+1}^0$-complete, where α is any computable ordinal.

The index set $I(P)$ does not have to use the enumeration of *all* computable members in a class. For example, we could instead start with a uniform enumeration of all polynomial time computable points $(l_e)_{e\in\omega}$ of the space; if we can still prove a completeness result restricted to $(l_e)_{e\in\omega}$ then this means that the property "x is polynomial time computable" does not help to characterise when P holds on x. We illustrate this approach by a non-trivial example. Following [11], we call $f \in C[0,1]$ regular to base 2 if the graph of f coded as a pair of binary strings is recognised by a Büchi automation. Regular continuous functions are Lipschitz and also map rationals to rationals in linear time. Using quantifier elimination one can express "f is regular" as a Σ_2^0-statement about f. Franklin et al. [17] have recently announced:

Theorem 24. *Given a linear time computable Lipschitz $f : [0,1] \to \mathbb{R}$ with $f(\mathbb{Q} \cap [0,1]) \subseteq \mathbb{Q}$, checking whether f is regular is a Σ_2^0-complete problem.*

Remark 25. Working independently, Gorman et al. [18] have announced that every continuous regular function has to be affine outside a measure 0 nowhere dense set. This property can be added to the list of properties in the theorem above too.

It follows that none of the properties of regular functions known to date helps to reduce the complexity of the definition of regularity.

3.1   Baire Hierarchy and Parallel Weihrauch Reducibility

So far our "local" results were restricted to continuous functions. If we want to extend the ideas beyond continuity, we need more ideas. We might have some hope for reasonable classes of functions such as the *Baire* classes. Recall that f is Baire class 0 iff it is continuous, and f is called Baire $n+1$ iff there is a collection $\{f_i \mid i \in \mathbb{N}\}$ of Baire class n functions such that $f(x) = \lim_n f_n(x)$, and this extends to limit ordinals in the obvious way. Baire 1 functions are precisely the derivatives of differentiable functions. Baire functions are all Lebesgue measureable, and any Lebesgue measureable function is the same as a Baire 2 function except on a set of measure 0. Can we say more? In the below, we will concentrate on *classification* for the Baire class functions.

Recall that *Weihrauch* reducibility [2] is defined as $f \leq_W g$ to mean there are computable h, k such that $f(x) = h(x, g(k(x)))$. A more general version of this is called *parallel* Weihrauch reducibility. The idea is that to compute $f(x)$ to within 2^{-n} we might explore $g(y)$, yet for $f(x)$ to within 2^{-m} perhaps $g(y')$. For this new reduction we will write $f \leq_T g$.

Remark 26. This is motivated as follows. It is not reasonable to expect that for a general reduction we should have a uniform map like $\leq_W$ which takes arguments of f pointwise to arguments of g determining the reduction. Turing reduction on sets, $A \leq_T B$, can use many queries to B to determine $A(n)$. In the case of continuous functions this can be viewed as as the "parallelisation" of g; replace g with ω many copies of g. See [13] for a clarification.

Example 27. Consider the α-th jump function, $j^{\alpha}(x)(i) = x^{(\alpha)}(i)$, where $x^{(\alpha)}(i)$ is 0 if $\Phi_i^x(i)$ does not halt and 1 otherwise. Then it is possible to show that $j^1(x) \leq_T S_0$ where S_0 is the step function which is 0 below 0 and 1 above.

In [13] Day, Downey and Westrick instigated an analysis of the classification of Baire functions using $\leq_T$. In that paper, they allowed real parameters, the *boldface* version $\leq_\mathbf{T}$. It is easy to show that f is Baire α iff $f \leq_\mathbf{T} j^{\alpha}$. Using recursion-theoretic techniques, Day, Downey and Westrick [13] have proved:

Theorem 28. *If α is a constructive ordinal and a Baire function f is not Baire α then $j^{\alpha+1} \leq_\mathbf{T} f$, and if f is limit, and f is not Baire β for $\beta < \alpha$, either $f \equiv_\mathbf{T} j^{\alpha}$, or $j^{\alpha} <_\mathbf{T} f$.*

In [13], Day, Downey and Westrick refined $\leq_\mathbf{T}$ to look at analogs of m- and tt-reductions; we omit the definitions[13].The idea being that Post's Theorem (for example) puts Σ_m^0 complete sets above the other Σ_m^0 sets via an m-reduction. We state the following satisfying classification result

Theorem 29 ([13])**.**

1. *For all Baire functions f and g, either $f \leq_\mathbf{m} g$ or $g \leq_\mathbf{m} - f$. Hence if f is Baire α then $f \leq_\mathbf{m} j^{\alpha+1}$.*
2. *If f is Borel and f is not Baire α, then either $j^{\alpha+1} \leq_\mathbf{m} f$ or $-j^{\alpha+1} \leq_\mathbf{m} f$.*

These new reductions had reflections in results from classical analysis to give computable "explanations" for classical results. In [13] Day et al. showed (classical) Baire and Bourgain hierarchies of functions intertwine with the degree structures above. Kihara (to appear) has recently extended the results above.

As we see the boldface version gives real insight into classical investigations, but it might be argued that a finer classification could be obtained using no parameters. This is still an open challenge. Lots of open problems remain, particularly what would be the correct notion for lightface versions.

References

1. Ash, C., Knight, J.: Computable Structures and the Hyperarithmetical Hierarchy. Studies Logic and the Foundations of Mathematics, vol. 144. North-Holland Publishing Co., Amsterdam (2000)
2. Brattka, V., Gherardi, G.: Borel complexity of topological operations on computable metric spaces. J. Logic Comput. **19**(1), 45–76 (2009)

---

[13] This is not immediate, as the idea of "positive information" in this setting needs clarification. Day et al. use *separation names.*

3. Becher, V., Heiber, P.A., Slaman, T.A.: Normal numbers and the Borel hierarchy. Fund. Math. **226**(1), 63–78 (2014)
4. Bohnenblust, H.F.: An axiomatic characterization of L_p-spaces. Duke Math. J. **6**, 627–640 (1940)
5. Brattka, V., Hertling, P., Weihrauch, K.: A tutorial on computable analysis. In: Cooper, S.B., Löwe, B., Sorbi, A. (eds.) New Computational Paradigms, pp. 425–491. Springer, New York (2008). https://doi.org/10.1007/978-0-387-68546-5_18
6. Brown, T., McNicholl, T., Melnikov, A.: On the complexity of classifying lebesgue spaces. To appear
7. Brown, T.A.: Computable Structure Theory on Banach Spaces. ProQuest LLC, Ann Arbor, MI (2019). Thesis (Ph.D.)-Iowa State University
8. Brown, T., McNicholl, T.: Analytic computable structure theory and L^p spaces-part 2 Archive for Mathematical Logic (to appear)
9. Becher, V., Slaman, T.A.: On the normality of numbers to different bases. J. Lond. Math. Soc. **90**(2), 472–494 (2014)
10. Cenzer, D., Remmel, J.B.: Index sets in computable analysis. Theoret. Comput. Sci. **219**(1–2), 111–150 (1999). Computability and complexity in analysis (Castle Dagstuhl, 1997)
11. Chaudhuri, S., Sankaranarayanan, S., Vardi, M.Y.: Regular real analysis. In: 2013 28th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS 2013), pp. 509–518. IEEE Computer Society, Los Alamitos (2013)
12. Clanin, J., McNicholl, T.H., Stull, D.M.: Analytic computable structure theory and L^p spaces. Fundamenta Mathematicae **244**, 255–285 (2019)
13. Day, A., Downey, R.G., Westrick, L.B.: Three topological reducibilities for discontinuous functions. To appear
14. Downey, R., Montalbán, A.: The isomorphism problem for torsion-free abelian groups is analytic complete. J. Algebra **320**(6), 2291–2300 (2008)
15. Effros, E.G.: Transformation groups and C^*-algebras. Ann. Math. **2**(81), 38–55 (1965)
16. Ershov, Y., Goncharov, S.: Constructive Models. Siberian School of Algebra and Logic. Consultants Bureau, New York (2000)
17. Franklin, J., Adam, R.H., Khoussainov, B., Melnikov, A., Ng, K.M.: Continuous functions and effective classification (2020)
18. Gorman, A., et al.: Continuous regular functions. Preprint (2019)
19. Goncharov, S., Knight, J.: Computable structure and antistructure theorems. Algebra Logika **41**(6), 639–681, 757 (2002)
20. Greenberg, N., Melnikov, A.G., Knight, J.F., Turetsky, D.: Uniform procedures in uncountable structures. J. Symb. Log. **83**(2), 529–550 (2018)
21. Gromov, M.: Metric structures for Riemannian and non-Riemannian spaces. Modern Birkhäuser Classics. Birkhäuser Boston Inc., Boston, MA, english edition (2007). Based on the 1981 French original, With appendices by M. Katz, P. Pansu and S. Semmes, Translated from the French by Sean Michael Bates
22. Kakutani, S.: Concrete representation of abstract L-spaces and the mean ergodic theorem. Ann. Math. **42**, 523–537 (1941)
23. Ki, H., Linton, T.: Normal numbers and subsets of **N** with given densities. Fund. Math. **144**(2), 163–179 (1994)
24. McNicholl, T.H.: Computable copies of ℓ^p. Computability **6**(4), 391–408 (2017)
25. McNicholl, T.H.: Computing the exponent of a Lebesgue space. To appear
26. Melnikov, A.G.: Computably isometric spaces. J. Symbolic Logic **78**(4), 1055–1085 (2013)

27. Melnikov, A.: Computable topological groups and Pontryagin duality. Trans. Amer. Math. Soc. **370**(12), 8709–8737 (2018)
28. Melnikov, A., Montalbán, A.: Computable Polish group actions. J. Symb. Log. **83**(2), 443–460 (2018)
29. Melnikov, A.G., Nies, A.: The classification problem for compact computable metric spaces. In: Bonizzoni, P., Brattka, V., Löwe, B. (eds.) CiE 2013. LNCS, vol. 7921, pp. 320–328. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39053-1_37
30. Melnikov, A.G., Ng, K.M.: Computable structures and operations on the space of continuous functions. Fund. Math. **233**(2), 101–141 (2016)
31. Montalbán, A.: Computability theoretic classifications for classes of structures. In: Proceedings of the International Congress of Mathematicians-Seoul 2014, vol. II, pp. 79–101. Kyung Moon Sa, Seoul (2014)
32. Montalbán, A.: A robuster Scott rank. Proc. Am. Math. Soc. **143**(12), 5427–5436 (2015)
33. Nies, A., Solecki, S.: Local compactness for computable polish metric spaces is π_1^1 -complete. In: Evolving Computability - 11th Conference on Computability in Europe, CiE 2015, Bucharest, Romania, 29 June–3 July 2015, pp. 286–290 (2015)
34. Pour-El, M.B., Richards, J.I.: Computability in Analysis and Physics. Perspectives in Mathematical Logic. Springer, Berlin (1989)
35. Sierpinski, W.: General topology. Mathematical Expositions, No. 7. University of Toronto Press, Toronto (1952). Translated by C. Cecilia Krieger
36. Schnorr, C.-P., Stimm, H.: Endliche Automaten und Zufallsfolgen. Acta Informat. **1**(4), 345–359 (1971)
37. Weihrauch, K.: Computable Analysis. Texts in Theoretical Computer Science. An EATCS Series. Springer, Berlin (2000). https://doi.org/10.1007/978-3-642-56999-9
38. Westrick, L.B.: A lightface analysis of the differentiability rank. J. Symb. Log. **79**(1), 240–265 (2014)