An approach to intrinsic complexity of uniform learning
Sandra Zilles

DFKI GmbH, Postfach 2080, 67608 Kaiserslautern, Germany

Abstract

Inductive inference is concerned with algorithmic learning of recursive functions. In the model of learning in the limit a learner successful for a class of recursive functions must eventually find a program for any function in the class from a gradually growing sequence of its values. This approach is generalised in uniform learning, where the problem of synthesising a successful learner for a class of functions from a description of this class is considered.

A common reduction-based approach for comparing the complexity of learning problems in inductive inference is intrinsic complexity. Informally, if a learning problem (a class of recursive functions) A is reducible to a learning problem (a class of recursive functions) B, then a solution for B can be transformed into a solution for A. In the context of intrinsic complexity, reducibility between two classes is expressed via recursive operators transforming target functions in one direction and sequences of corresponding hypotheses in the other direction.

The present paper is concerned with intrinsic complexity of uniform learning. The relevant notions are adapted and illustrated by several examples. Characterisations of complete classes finally allow for various insightful conclusions. The connection to intrinsic complexity of non-uniform learning is revealed within several analogies concerning first the structure of complete classes and second the general interpretation of the notion of intrinsic complexity.

© 2006 Elsevier B.V. All rights reserved.

Keywords: Inductive inference; Learning theory; Recursion theory

1. Introduction

Inductive inference is concerned with algorithmic learning of recursive functions. In the model of learning in the limit, see Gold [9], a learner successful for a class of recursive functions must eventually find a correct program for any function in the class from a gradually growing sequence of its values. The learner is understood as a machine—called inductive inference machine or IIM—reading finite sequences of input–output pairs of a target function, and returning programs as its hypotheses, see also Blum and Blum [4]. The underlying programming system is then called a hypothesis space. For a survey on inductive inference of recursive functions, the reader is referred to Angluin and Smith [1].

For some special learning problem, i.e., some special class of recursive functions, a learner in general is successful, because it has some prior intrinsic knowledge about the class of target functions. Analogously, if a class of target functions is not learnable, then the required background knowledge is presumably not expressible adequately to be exploited by a learner.

* Tel.: +49 631 205 3464.
E-mail address: sandra.zilles@dfki.de.
For instance, consider a phenomenon often witnessed in abstract learning models: the existence of two classes of target objects (here classes of recursive functions), such that each of the target classes can be learned easily, but the union of the two classes is not learnable in the given model.

That means, a learner for any of the two target classes has enough information (intrinsic knowledge about the class of target functions), but it is impossible to acquire the knowledge needed for learning the union of the two classes. Any learner will fail in this problem. How can a learner acquire enough intrinsic knowledge to learn the union of two, three, or even infinitely many classes?

As the program for a learner is finite, it is impossible to store infinitely many pieces of information (about infinitely many classes of target functions) within a single learner. A quite natural approach to tackling this weakness of inductive learning is to communicate the required background knowledge to the learner instead of storing it in the learners in advance. In other words, the idea is to provide the learner with two kinds of information during the learning process:

- information about the target function (as usual), plus
- information about which class the unknown target function belongs to (extra information).

The additional information may help the learner to cope with several classes, the union of which is not learnable in the initial model. The benefit of this extra knowledge is that it reduces the hypothesis space, because it restricts the amount of possible target functions and thus the pool of functions the current unknown function may belong to.

This point of view can also be expressed in other terms: if intrinsic knowledge of learners about the target classes or hypothesis spaces is assumed, can this knowledge be exploited in a uniform way? That means, we ask for common preconditions in learnable classes of target functions, allowing for a common (uniform) method of induction for all these classes. The idea is to aim at some kind of meta-learner \( M \) simulating several (perhaps infinitely many) learners for special classes \( C_0, C_1, C_2, \ldots \) of target functions. This realises an approach to merging several intelligent systems into a single machine able to cope with the tasks of any of the systems, which is not a trivial task, if the resulting machine is required to represent a computable device. In other words, a single creative learning procedure shall be used for numerous learning problems. This approach is referred to by the term uniform learning. In summary, an analysis of uniform learning is of interest for several reasons, for example:

- it concerns the general problem of designing learning systems capable of simulating numerous expert learners for special target classes;
- it concerns common principles of solvable learning problems and common principles for possible corresponding successful learners;
- it concerns the general problem of representing learning problems adequately, and thus of appropriately communicating background knowledge on the particular target classes to the learner.

In particular, the latter aspect has to be explained in detail. Recall that the crucial component of uniform learning is supposed to be some kind of meta-learner \( M \) simulating several (perhaps infinitely many) learners for special classes \( C_0, C_1, C_2, \ldots \) of target functions—which may be seen as a decomposition of a class \( C = C_0 \cup C_1 \cup \cdots \). As the intrinsic knowledge to be used by the meta-learner \( M \) may depend on the class \( C_i \) of target functions currently considered, there must be some way to communicate this special knowledge about \( C_i \) to \( M \). This is done via some description \( d_i \) representing the class \( C_i \) of recursive target functions. As an input, a meta-IIM gets such a description \( d_i \) of one of the learning problems, i.e., classes of recursive functions, \( C_i \) in the collection. The meta-IIM is then supposed to develop a successful IIM for \( C_i \). That means, the IIM resulting from the meta-IIM on input of the extra information \( d_i \) is required to act as an appropriate learner for \( C_i \) when given the usual information on target functions in \( C_i \). Besides studies on uniform learning of classes of recursive functions, see [24,26] and Jantke [15], this topic has also been investigated in the context of learning formal languages, see in particular Baliga et al. [2], Kapur and Bilardi [16], and Osherson et al. [21].

Since we consider IIMs as tackling a given problem, namely the problem of identifying all elements in a particular class of recursive functions, the complexity of such IIMs might express, how hard a learning problem is. For instance, the class of all constant functions allows for a simple and straightforward identification method; for other classes successful methods might seem more complicated. But this does not involve any rule allowing us to compare two learning problems with respect to their difficulty. So a formal approach for comparing the complexity of learning problems (i.e., of classes of recursive functions) is desirable.

Different aspects have been analysed in this context. One approach is, e.g., mind change complexity measured by the maximal number of hypothesis changes a machine needs to identify a function in the given class, see Case and
Smith [5]. But since in general this number of mind changes is unbounded, other notions of complexity might be of interest.

Various subjects in theoretical computer science deal with comparing the complexity of decision problems, e.g., regarding decidability as such, see Rogers [22], or the possible efficiency of decision algorithms, see Garey and Johnson [7]. In general Problem A is no harder than Problem B, if A is reducible to B under a given reduction. Each such reduction involves a notion of complete (hardest solvable) problems. Besides studies concerning language learning—see Jain et al. [12] or Jain and Sharma [13,14]—Freivalds et al. [6] introduce an approach for reductions in the context of learning recursive functions. This subject, intrinsic complexity, has been further analysed by Kinber et al. [17] and Jain et al. [11] with a focus on complete classes. They have shown that, for learning in the limit, a class is complete, if and only if it contains an r.e. subclass, which is dense respecting the Baire metric, see Rogers [22]. Here the aspect of high topological complexity (density) contrasts with the aspect of low algorithmic complexity of r.e. sets.

In the context of uniform learning an approach to measuring the learning complexity is conceivable, as well. Here a learning problem is a problem of synthesising learners for classes of recursive functions from corresponding descriptions. Jantke [15] has shown that sets of descriptions representing rather trivial classes of recursive functions can form unsolvable learning problems, whereas sets of descriptions representing much more complex classes can form solvable problems (see also [24]). So, apparently, uniform learnability does not only depend on the classes of recursive functions represented, but to a large extent also on the respective descriptions for these classes. This raises the question of how to tell by the particular descriptions how hard a uniform learning problem is. Is there any rule or criterion concerning uniform learning problems allowing for a predication about their complexity? More precisely:

• How should a suitable notion of reducibility be defined in order to express relations concerning intrinsic complexity of uniform learning?
• How can complete learning problems be characterised in this context?
• Are there any analogies between intrinsic complexity in the non-uniform approach and intrinsic complexity in the uniform approach?

Below a notion of intrinsic complexity for uniform learning is developed and the corresponding complete classes are characterised. The obtained structure of degrees of complexity matches recent results on uniform learning: it has been shown that even decompositions into singleton classes can yield problems too hard for uniform learning in Gold’s model. This suggests that collections representing singleton classes may sometimes form hardest problems in uniform learning. Indeed, the notion developed below expresses this intuition, i.e., collections of singleton sets may constitute complete classes in uniform learning.

Comparing completeness here to completeness in the original notion of intrinsic complexity reflects strong relations between the uniform and the non-uniform approach. As in the non-uniform case, our characterisations reveal that a high topological complexity and a low algorithmic complexity are common and characteristic properties for complete learning problems in uniform learning. Additionally, the characterisations allow for proving that hardest problems in non-uniform learning can be formulated as uniform learning problems of lower complexity. In other words, each class C of recursive functions which is complete in the original notion can be decomposed into a uniformly learnable collection \( C \), \( C_1, \ldots \), which is not a hardest problem in uniform learning. Informally, this simply reflects how a very complex problem can be split into several subproblems, which can be solved in a uniform way summing up to a less complex problem. Here it is important to choose some appropriate way of decomposing \( C \), because—contrastingly—it will be demonstrated that \( C \) may also be decomposed into a collection of highest complexity in uniform learning.

All in all, this shows that intrinsic complexity as defined by Freivalds et al. [6] can be adapted to match the intuitively desired results in uniform learning.

A preliminary version of this paper has already appeared, see [25].

2. Preliminaries

2.1. Notations

Knowledge of basic notions used in mathematics and computability theory is assumed, see Rogers [22]. \( \mathbb{N} \) is the set of natural numbers. The cardinality of a set \( X \) is denoted by \( \text{card}(X) \). Partial-recursive functions always operate on natural numbers. If \( f \) is a function, \( f(n) \uparrow \) indicates that \( f(n) \) is undefined. Our target objects for learning will always be recursive functions, i.e., total partial-recursive functions. \( \mathcal{R} \) denotes the set of all recursive functions.
If \( x \) is a finite tuple of numbers, then \(|x|\) denotes its length. Finite tuples are coded, that is, if \( f(0), \ldots, f(n) \) are defined, a number \( f[n] \) represents the tuple \( (f(0), \ldots, f(n)) \), called an initial segment of \( f \). \( f[n] \uparrow \) means that \( f(x) \uparrow \) for some \( x \leq n \). For convenience, a function may be written as a sequence of values or as a set of input–output pairs. A sequence \( \sigma = x_0, x_1, x_2, \ldots \), converges to \( x \), iff \( x_n = x \) for all but finitely many \( n \); we write \( \lim(\sigma) = x \). For example let \( f(n) = 7 \) for \( n \leq 2 \), \( f(n) \uparrow \) otherwise; \( g(n) = 7 \) for all \( n \). Then \( f = \{ (0, 7), (1, 7), (2, 7) \} \), \( g = \{ (n, 7) \mid n \in \mathbb{N} \} \); \( \lim(g) = 7 \), and \( f \subseteq g \). For \( n \in \mathbb{N} \), the notion \( f =_n g \) means that for all \( x \leq n \) either \( f(x) \uparrow \) and \( g(x) \uparrow \) or \( f(x) = g(x) \).

As it will turn out, topological aspects of classes of recursive functions will play a decisive role in the context of learning complexity. Here a metric space topology is induced by the so-called Baire metric, see Rogers [22]. If \( C \) is a set of functions, then the Baire metric \( \delta \) on \( C \) is defined as follows:

\[
\delta(f, g) = \begin{cases} 
0 & \text{if } f = g, \\
\frac{1}{\min\{x \mid f(x) \neq g(x)\} + 1} & \text{otherwise,}
\end{cases}
\]

for all \( f, g \in C \). Note that \( C \) is dense with respect to the Baire metric, iff for any \( f \in C, n \in \mathbb{N} \) there is some \( g \in C \) (and thus infinitely many \( g \in C \)) satisfying \( f =_n g \), but \( f \neq g \).

Recursive functions—our target objects for learning—require appropriate representation schemes, to be used as hypothesis spaces. Partial-recursive enumerations serve for that purpose: any \( n \)-place partial-recursive function \( \psi \) enumerates the set \( \mathcal{P}_\psi := \{ \psi_i \mid i \in \mathbb{N} \} \) of \( n \)-place partial-recursive functions, where \( \psi_i(x) := \psi(i, x) \) for all \( x = (x_1, \ldots, x_n) \). Then \( \psi \) is called a numbering. Given \( f \in \mathcal{P}_\psi \), any index \( i \) satisfying \( \psi_i = f \) is a \( \psi \)-program of \( f \). As a special case, we consider acceptable numberings, such as, e.g., programming systems derived from an enumeration of all Turing machines, see Rogers [22].

Following Gold [8], we call a family \((d_i)_{i \in \mathbb{N}}\) of natural numbers limiting r.e., iff there is a recursive numbering \( \mathcal{D} \) such that \( \lim(\mathcal{D}) = d_i \) for all \( i \in \mathbb{N} \).

2.1. Learning in the limit

The crucial components of a model of inductive learning are the learner, the class of possible target objects, as well as a representation scheme to be used as a hypothesis space. The target objects in inductive inference considered here are always recursive functions; as a representation scheme some adequate partial-recursive numbering is chosen. It remains to specify the type of learners to be used. Each learner can be considered as some kind of machine, called inductive inference machine or IIM for short. An IIM \( M \) is an algorithmic device working in time steps. In step \( n \) it gets some input \( f[n] \) corresponding to an initial segment of a graph of some recursive function \( f \). If \( M \) returns an output on \( f[n] \), then this output is a natural number to be interpreted as a program in the given numbering serving as a hypothesis space, see Gold [9]. As usual, an IIM which is defined on any input will be called a total IIM. Subsequently, the term 'hypothesis space' will always refer to a two-place partial-recursive numbering.

In Gold’s basic model of identification in the limit, see Gold [9], the IIM working on the graph of some recursive target function \( f \) is required to produce guesses converging to a correct program for \( f \).

Definition 1 (Gold [9]). Let \( C \subseteq \mathcal{R} \). \( C \) is identifiable in the limit, \(^1\) iff there is some hypothesis space \( \psi \) and an IIM \( M \), such that for any \( f \in C \) the following conditions are fulfilled:

1. \( M(f[n]) \) is defined for all \( n \in \mathbb{N} \),
2. there is some \( i \in \mathbb{N} \), such that \( \psi_i = f \) and \( M(f[n]) = i \) for all but finitely many \( n \in \mathbb{N} \).

In this case, \( M \) is called an Ex-learner for \( C \) with respect to \( \psi \). Ex denotes the collection of all Ex-learnable classes \( C \subseteq \mathcal{R} \).

Each finite class \( C \subseteq \mathcal{R} \) is trivially Ex-learnable. In general, each class \( C \subseteq \mathcal{R} \) of functions enumerated by a recursive numbering belongs to Ex, see Gold’s method of identification by enumeration [9]. As an illustration

\(^1\) This is also referred to by the term Ex-identifiable associated with the phrase explanatory identification; subsequently, the phrases identifiable and learnable will be used synonymously.
for such enumerable classes $C$ consider the class of all primitive recursive functions or the class

$$C_{\text{zero}} := \{x0^n \mid x \text{ a finite tuple over } \mathbb{N}\}$$

of all recursive finite variants of the zero function. The latter will be used for several examples below.

In contrast to that, there is no Ex-learner successful for the whole class $R$ of recursive functions, no matter which hypothesis space is used.

Finally, note that each class $C \in \text{Ex}$ is Ex-learnable with respect to any acceptable numbering.

3. Uniform learning in the limit

Uniform learning views the approach of Ex-learning on a meta-level; it is not only concerned with the existence of methods solving specific learning problems, but with the problem of synthesising such methods. So the focus is on families of learning problems (here families of classes of recursive functions). Given a representation or description of a class of recursive functions, the aim is to effectively determine an adequate learner, that is, to compute a program for a successful IIM learning the class.

For a formal definition of uniform learning it is necessary to agree on a method for describing classes of recursive functions (i.e., describing learning problems). For that purpose we fix a three-place acceptable numbering $\varphi$. If $d \in \mathbb{N}$, the numbering $\varphi^d$ is the function resulting from $\varphi$, if the first input is fixed by $d$. Then any number $d$ corresponds to a two-place numbering $\varphi^d$ enumerating the set $\mathcal{P}_{\varphi^d} = \{\varphi^d_i \mid i \in \mathbb{N}\}$ of partial-recursive functions. Now it is possible to consider the subset of all total functions in $\mathcal{P}_{\varphi^d}$ as a learning problem which is uniquely determined by the number $d$. Thus each number $d$ acts as a description of the set $\mathcal{R}_d$, where

$$\mathcal{R}_d := \{\varphi^d_i \mid i \in \mathbb{N} \text{ and } \varphi^d_i \text{ is recursive} \} = \mathcal{P}_{\varphi^d} \cap \mathcal{R} \text{ for any } d \in \mathbb{N}. $$

$\mathcal{R}_d$ is called the recursive core of the numbering $\varphi^d$. So any set $D = \{d_0, d_1, \ldots\}$ can be regarded as a set of descriptions, i.e., a collection of learning problems $\mathcal{R}_{d_0}, \mathcal{R}_{d_1}, \ldots$. In this context, $D$ is called a description set.

A meta-IIM $M$ is an IIM with two inputs: (i) a description $d$ of a recursive core $\mathcal{R}_d$, and (ii) an initial segment $f[n]$ of some $f \in \mathcal{R}$. Then $M_d$ is the IIM resulting from $M$, if the first input is fixed by $d$. A meta-IIM $M$ can be seen as mapping descriptions $d$ to IIMs $M_d$; it is a successful uniform learner for a set $D$, in case $M_d$ learns $\mathcal{R}_d$ for all $d \in D$; that means, given any description in $D$, $M$ develops a suitable learner for the corresponding recursive core.

As a first approach, it seems reasonable to choose one hypothesis space to be used for identifying all the recursive cores $\mathcal{R}_d$ described by a set $D$. Since each Ex-learnable class can be Ex-identified with respect to any acceptable numbering, it is possible to choose an acceptable numbering for this hypothesis space.

**Definition 2.** Let $D \subseteq \mathbb{N}$ be a description set and let $\tau$ be an acceptable numbering. $D$ is uniformly Ex-learnable with respect to $\tau$, iff there is a meta-IIM $M$, such that, for any description $d \in D$, the IIM $M_d$ is an Ex-learner for the class $\mathcal{R}_d$ with respect to $\tau$. In this case, $M$ is called a UniEx-learner for $D$ with respect to $\tau$. $D$ is uniformly Ex-learnable, iff there is an acceptable numbering, such that $D$ is uniformly Ex-learnable with respect to that numbering. UniEx denotes the collection of all uniformly Ex-learnable description sets.

Again, note that—similar to Ex-learnable classes—each description set $D \in \text{UniEx}$ can be UniEx-identified with respect to any acceptable numbering.

Our second approach starts with the observation that, as a numbering, $\varphi^d$ enumerates a superset of $\mathcal{R}_d$. Thus a meta-IIM might also use $\varphi^d$ as a hypothesis space for $\mathcal{R}_d$. This approach yields just a special (restricted) case of uniform Ex-learning, because $\varphi^d$-programs can be uniformly translated into $\tau$-programs for any acceptable numbering $\tau$.

**Definition 3.** Let $D \subseteq \mathbb{N}$ be a description set. $D$ is strongly uniformly Ex-learnable, iff there is a meta-IIM $M$, such that, for any description $d \in D$, the IIM $M_d$ is an Ex-learner for the class $\mathcal{R}_d$ with respect to $\varphi^d$. In this case, $M$ is called a $\text{resUniEx}$-learner for $D$. resUniEx denotes the class of all description sets which are strongly uniformly Ex-learnable.

---

2 Note that, by intuition, it seems adequate to talk of uniformly learnable collections of recursive cores represented by description sets, rather than of uniformly learnable description sets themselves. Yet, for convenience, the latter notion is preferred.

3 The notion resUniEx symbolises a restricted variant of the model UniEx.
Now that we have defined two approaches to uniform learning, let us first have a look at some simple examples.

First, if \( D \) is a finite description set and \( \mathcal{R}_d \in \text{Ex} \) for all \( d \in D \), then \( D \) is \( \text{UniEx} \)-identifiable (but not necessarily \( \text{resUniEx} \)-identifiable, see for instance [24]). Second, if \( D \) is chosen such that the union of all recursive cores described by \( D \) is \( Ex \)-learnable, then \( D \) is also \( \text{UniEx} \)-identifiable (but not necessarily \( \text{resUniEx} \)-identifiable, see for instance [24]). Of course, \( Ex \)-learnability of each recursive core described by \( D \) is a necessary condition for \( \text{UniEx} \)-learnability of \( D \).

Note that in the examples above the witnessing IIMs can be chosen to be total. As the following remark states, it is the case that we may assume without any loss of generality that witnessing meta-IIMs are always total.

**Remark 4 (see Zilles [26]).** Let \( D \subseteq \mathbb{N} \) be a description set and let \( \tau \) be an acceptable numbering.

1. If \( D \in \text{UniEx} \), then there exists a total meta-IIM \( M \) which is a \( \text{UniEx} \)-learner for \( D \) with respect to \( \tau \).
2. If \( D \in \text{resUniEx} \), then there exists a total meta-IIM \( M \) which is a \( \text{resUniEx} \)-learner for \( D \).

From the discussion above we have a few simple examples of uniform learning. In order to illustrate that uniform learning is not a trivial task, it is also necessary to look at some negative results, i.e., description sets not uniformly learnable or not strongly uniformly learnable. For instance, Theorem 5 states that the set of all descriptions representing only singleton recursive cores is not uniformly learnable. Moreover, given any fixed recursive function \( r \), the set of all descriptions representing only the singleton recursive core \( \{ r \} \) is not strongly uniformly learnable.

**Theorem 5 (Jantke [15], Zilles [24,26]).** (1) \( \{ d \in \mathbb{N} \mid \text{card}(\mathcal{R}_d) = 1 \} \notin \text{UniEx} \).

(2) Fix \( r \in \mathcal{R} \). Then \( \{ d \in \mathbb{N} \mid \mathcal{R}_d = \{ r \} \} \notin \text{resUniEx} \).

At first glance, these results seem very discouraging. Finite classes are trivially \( Ex \)-learnable, particularly singletons. Still, in the context of uniform learning they may form unsolvable learning problems. Does Theorem 5 say that not even the most simple recursive cores can be learned uniformly? After reconsideration, it does not quite say so. In fact, all singleton recursive cores—even all \( Ex \)-learnable recursive cores—can be identified uniformly, if the descriptions representing these recursive cores are chosen more specifically.

**Theorem 6 (Jantke [15]).** There is a description set \( D \subseteq \mathbb{N} \), such that the following conditions are fulfilled:

1. For any \( C \in \text{Ex} \) there is some \( d \in D \) satisfying \( C \subseteq \mathcal{R}_d \).
2. \( C \in \text{resUniEx} \).

The contrast between Theorems 5 and 6 deserves closer attention. On the one hand we have had very negative results concerning the learnability of rather simple classes of recursive functions in case a meta-learner is supposed to cope with any corresponding description. On the other hand, Theorem 6 impressively shows that a more selective choice of descriptions may yield global positive results.

Apparently, the learnability of a description set \( D \) does not only depend on the recursive cores represented by \( D \), but to a large extent also on the respective descriptions for these recursive cores. Considering each recursive core as a learning problem and \( D \) as a uniform learning problem, the complexity of \( D \) is not only influenced by the complexity of the recursive cores. Here the notion of complexity informally is used to express how hard a learning problem is.

This raises the question of how to tell by the particular descriptions in a set \( D \) how hard the uniform learning problem \( D \) is. Is there any rule or criterion concerning description sets allowing for a predication about their uniform learning complexity? This is the central question discussed in the following.

4. **Intrinsic complexity**

Fortunately, the approach of intrinsic complexity for non-uniform inductive inference—as propagated by Freivalds et al. [6]—will turn out to be applicable for uniform learning, too. For that purpose, this section gives a short formal introduction into intrinsic complexity, which will afterwards be adapted for our purposes.

The basic conception needed for the transformation of problems in the reductions is that of recursive operators, see in particular Rogers [22].
Definition 7 (Rogers [22], Kinber et al. [17], Jain et al. [11]). Let $\Theta$ be a total function mapping functions to functions. $\Theta$ is called a recursive operator, iff the following three properties are satisfied for all functions $f$ and $f'$ and all numbers $n$, $y \in \mathbb{N}$:

1. if $f \subseteq f'$, then $\Theta(f) \subseteq \Theta(f')$;
2. if $\Theta(f)(n) = y$, then there exists some initial subfunction $g \subseteq f$ such that $\Theta(g)(n) = y$;
3. if $f$ is finite, then one can effectively (in $f$) enumerate $\Theta(f)$.

Reducing one class $C_1 \subseteq \mathcal{R}$ to another class $C_2 \subseteq \mathcal{R}$ involves two recursive operators: the first one transforms each function in $C_1$ into a function in $C_2$; the second operator transforms any successful hypothesis sequence for the obtained function in $C_2$ into a successful hypothesis sequence for the original function in $C_1$. This requires a notion of successful hypothesis sequences, called admissible sequences in the context of learning in the limit.

Definition 8 (Freivalds et al. [6]). Fix an acceptable numbering $\tau$ and let $f \in \mathcal{R}$. An infinite sequence $\sigma$ is called $Ex$-admissible for $f$ with respect to $\tau$, iff $\sigma$ converges to a $\tau$-program for $f$.

Suppose $\tau$, $\tau'$ are two acceptable numberings. Obviously, there is an effective procedure for transforming each $Ex$-admissible sequence for any function $f$ with respect to $\tau$ into an $Ex$-admissible sequence for $f$ with respect to $\tau'$. Thus Definition 8 can be generalised: a sequence $\sigma$ is $Ex$-admissible for $f$, iff it is $Ex$-admissible for $f$ with respect to some acceptable numbering, which is fixed a priori.

This finally allows us to define the desired reducibility relation.

Definition 9 (Freivalds et al. [6], Kinber et al. [17], Jain et al. [11]). Let $C_1 \in Ex$ and $C_2 \in Ex$. Then $C_1$ is called $Ex$-reducible to $C_2$, iff there exist recursive operators $\Theta$ and $\Xi$ such that each function $f$ in $C_1$ satisfies the following two conditions:

1. $\Theta(f) \in C_2$,
2. if $\sigma$ is an $Ex$-admissible sequence for $\Theta(f)$, then $\Xi(\sigma)$ is an $Ex$-admissible sequence for $f$.

Note that, if $C_1$ is $Ex$-reducible to $C_2$, then an $Ex$-learner for $C_1$ can be computed from any $Ex$-learner for $C_2$.

For instance, each $Ex$-learnable class is $Ex$-reducible to the class

$$C_{\text{zero}} = \{x0^\infty \mid x \text{ is a finite tuple over } \mathbb{N}\}$$

of all recursive functions of finite support, see Freivalds et al. [6]. In other words, $C_{\text{zero}}$ is an $Ex$-complete class, i.e., an $Ex$-learnable class of highest complexity respecting the notion of $Ex$-reducibility, see Definition 10.

Definition 10 (Freivalds et al. [6], Kinber et al. [17], Jain et al. [11]). A class $C \subseteq \mathcal{R}$ is $Ex$-complete, iff $C$ is $Ex$-identifiable and each $Ex$-identifiable class is $Ex$-reducible to $C$.

$C_{\text{zero}}$ is an r.e. class and each initial segment of any function in $C_{\text{zero}}$ is an initial segment of even infinitely many functions in $C_{\text{zero}}$. As it turns out, these properties are somehow characteristic for $Ex$-complete classes; the following corresponding characterisation has been verified by Kinber et al. [17].

Theorem 11 (Kinber et al. [17], Jain et al. [11]). Let $C \in Ex$. $C$ is $Ex$-complete, iff there is a recursive numbering $\psi$, such that

1. $\mathcal{P}_\psi \subseteq C$;
2. for each $i$, $n \in \mathbb{N}$ there are infinitely many $j \in \mathbb{N}$ satisfying $\psi_i =_n \psi_j$ and $\psi_i \neq \psi_j$.

So $Ex$-complete classes have subsets, which are on the one hand topologically complicated (in terms of density respecting the Baire metric, as demanded in the second property in Theorem 11) but on the other hand algorithmically simple (being r.e. as demanded in the first property in Theorem 11). The latter is astonishing, in particular, since there is a subset $C'$ of $C_{\text{zero}}$, which is of the same topological complexity, but of lower intrinsic complexity than $C_{\text{zero}}$.

4 Thus, with respect to the Baire metric, the class $\mathcal{P}_\psi$ is dense.
see Kinber et al. [17]. That means, \( C' \) contains subsets which are dense with respect to the Baire metric, none of these subsets are r.e., but \( C' \) is \textit{not} \( \text{Ex} \)-complete. In other words, \( C' \) is in some sense algorithmically more complex than \( C_{\text{zero}} \), but belongs to a lower degree of intrinsic complexity.

The existence of r.e. subclasses in an \( \text{Ex} \)-complete class \( C \), as in Theorem 11, is due to the fact that the recursive enumerability of certain \( \text{Ex} \)-complete sets (such as \( C_{\text{zero}} \)) is ‘transferred’ by the operator \( \Theta \) mapping \( C_{\text{zero}} \) to \( C \). So perhaps this approach of intrinsic complexity just makes a class complete, if it is a suitable ‘target’ for recursive operators. This might lead to the interpretation that only those classes are complete, which are adequate for transferring successful learners into learners successful for other \( \text{Ex} \)-learnable classes. Of course, this might not exactly express the reader’s intuition of the complexity of learning problems, so it is possible to consider also other ideas of intrinsic complexity. Still the approach proposed by Freivalds et al. [6] has been widely analysed and is accepted as a suitable conception of complexity in the context of inductive inference. Here it is important to note that a ‘hardest’ learning problem does not necessarily demand a learner which is intuitively ‘harder to define’. A learning problem is ‘hardest’, if it has certain structural properties allowing for the required \( \text{Ex} \)-reductions.

5. Intrinsic complexity of uniform learning

5.1. Definition

In order to adapt these conceptions for uniform learning, the crucial step is to define the notion of reducibility in uniform learning between two description sets \( D_1 \) and \( D_2 \). As in the non-uniform model, reducibility ought to express intrinsic complexity in the sense that a meta-learner for \( D_1 \) can be computed from a meta-learner for \( D_2 \), if \( D_1 \) is reducible to \( D_2 \). In order to reduce the amount of formal notions used for adapting the approach of intrinsic complexity, we shall first focus exclusively on \( \text{UniEx} \)-identification and deal with \( \text{resUniEx} \)-identification a bit later.

A first idea for \( \text{UniEx} \)-reducibility might be to demand the existence of operators \( \Theta \) and \( \Xi \), such that for \( d_1 \in D_1 \) and \( f_1 \in R_{d_1} \):
- the operator \( \Theta \) transforms \((d_1, f_1)\) into a pair \((d_2, f_2)\);
- \( d_2 \in D_2 \) and \( f_2 \in R_{d_2} \);
- the operator \( \Xi \) transforms any \( \text{Ex} \)-admissible sequence for \( f_2 \) into an \( \text{Ex} \)-admissible sequence for \( f_1 \).

Unfortunately, this approach has a severe drawback. Whereas the construction of the function \( f_2 \) may work piecemeal and may depend on \( d_1 \) and the whole function \( f_1 \), the value \( d_2 \) can be computed only from \( d_1 \) and some finite initial segment of \( f_1 \). If \( n \in \mathbb{N} \) satisfies \( \Theta(d_1, f_1[n]) = (d_2, f_2[m]) \) for some \( m \in \mathbb{N} \) (such an \( n \) must exist), then none of the values \( f_1(n') \) for \( n' > n \) affects the computation of \( d_2 \). Informally speaking, the function \( f_2 \) may be determined in the limit, while the value \( d_2 \) must not.

That means, we would prefer an operator \( \Theta \), which computes not only \( f_2 \), but also \( d_2 \) in the limit. In other words, the operator \( \Theta \) should be allowed to return a \textit{sequence} of descriptions, when fed a pair \((d_1, f_1)\). As an improved approach based on this argumentation, it is conceivable to demand the existence of operators \( \Theta \) and \( \Xi \), such that for \( d_1 \in D_1 \) and \( f_1 \in R_{d_1} \):
- the operator \( \Theta \) transforms \((d_1, f_1)\) into a pair \((\delta_2, f_2)\);
- \( \delta_2 \) is a sequence of descriptions converging to some description \( d_2 \in D_2 \) with \( f_2 \in R_{d_2} \);
- the operator \( \Xi \) transforms any \( \text{Ex} \)-admissible sequence for \( f_2 \) into an \( \text{Ex} \)-admissible sequence for \( f_1 \).

At first glance, this approach seems reasonable, but it bears a problem. In the usual notion of reductions and intrinsic complexity, the reducibility predicate should be transitive. That means, if \( D_1 \) is reducible to \( D_2 \) and \( D_2 \) is reducible to \( D_3 \), then also \( D_1 \) should be reducible to \( D_3 \). In general, such a transitivity is achieved by connecting the operators of the first reduction with the operators of the second reduction. The approach above cannot guarantee such a transitivity: assume \( D_1 \) is reducible to \( D_2 \) via \( \Theta_1 \) and \( \Xi_1 \); \( D_2 \) is reducible to \( D_3 \) via \( \Theta_2 \) and \( \Xi_2 \). If \( \Theta_1 \) transforms \((d_1, f_1)\) into \((\delta_2, f_2)\), then which description \( d \) in the sequence \( \delta_2 \) has to be chosen to form an input \((d, f_2)\) for \( \Theta_2 \)? It is in general impossible to detect the limit \( d_2 \) of the sequence \( \delta_2 \), and any description \( d \) different from \( d_2 \) might change the output of \( \Theta_2 \).

Based on the above considerations, we require that the operator \( \Theta \) operate on \textit{sequences} of descriptions and on functions. That means, \( \Theta \) transforms each pair \((\delta_1, f_1)\), where \( \delta_1 \) is a sequence of descriptions, into a pair \((\delta_2, f_2)\). As usual, monotonicity of \( \Theta \) is required, and hence, if \( \delta_1' \subseteq \delta_1, f_1' \subseteq f_1 \), then

\[
\Theta(\delta_1', f_1') = (\delta_2', f_2'), \quad \Theta(\delta_1, f_1) = (\delta_2, f_2) \quad \text{implies} \quad \delta_2' \subseteq \delta_2, f_2' \subseteq f_2.
\]
This leads us to a formal definition of the required operators; they will be called *meta-operators* to indicate that they are needed for the reductions in uniform learning.

**Definition 12.** Let \( \Theta \) be a total function mapping pairs of functions to pairs of functions. \( \Theta \) is called a recursive meta-operator, iff the following properties are satisfied for all functions \( \delta_1, \delta'_1 \) and \( f_1, f'_1 \) and all numbers \( n, y \in \mathbb{N} \):

1. if \( \delta_1 \subseteq \delta'_1, f_1 \subseteq f'_1 \), as well as \( \Theta(\delta_1, f_1) = (\delta_2, f_2) \) and \( \Theta(\delta'_1, f'_1) = (\delta'_2, f'_2) \), then \( \delta_2 \subseteq \delta'_2 \) and \( f_2 \subseteq f'_2 \);
2. if \( \Theta(\delta_1, f_1) = (\delta_2, f_2) \) and \( \delta_2(n) = y \) (or \( f_2(n) = y \), respectively), then there exist subfunctions \( \gamma_1 \subseteq \delta_1 \) and \( g_1 \subseteq f_1 \) such that \( (\gamma_2, g_2) = \Theta(\gamma_1, g_1) \) fulfils \( \gamma_2(n) = y \) (or \( g_2(n) = y \), respectively);
3. if \( \delta_1 \) and \( f_1 \) are finite and \( \Theta(\delta_1, f_1) = (\delta_2, f_2) \), then one can effectively enumerate \( \delta_2 \) and \( f_2 \).

This finally allows for the following definition of *UniEx*-reducibility.

**Definition 13.** Let \( D_1, D_2 \subseteq \mathbb{N} \) be description sets in UniEx. Fix a recursive meta-operator \( \Theta \) and a recursive operator \( \Xi. D_1 \) is UniEx-reducible to \( D_2 \) via \( \Theta \) and \( \Xi \), iff for any description \( d_1 \in D_1 \), any function \( f_1 \in R_{d_1} \), and any finite tuple \( \delta_1 \) over \( \mathbb{N} \) there is a sequence \( \delta_2 \) and a function \( f_2 \) satisfying the following properties:

1. \( \Theta(d_1, f_1) = (\delta_2, f_2) \);
2. \( \delta_2 \) converges to some description \( d_2 \in D_2 \) such that \( f_2 \in R_{d_2} \);
3. if \( \sigma \) is an Ex-admissible sequence for \( f_2 \), then \( \Xi(\sigma) \) is Ex-admissible for \( f_1 \).

\( D_1 \) is UniEx-reducible to \( D_2 \), iff there exist a recursive meta-operator \( \Theta \) and a recursive operator \( \Xi \), such that \( D_1 \) is UniEx-reducible to \( D_2 \) via \( \Theta \) and \( \Xi \).

Note that this definition expresses intrinsic complexity in the sense that a meta-IIM for \( D_1 \) can be computed from a meta-IIM for \( D_2 \), if \( D_1 \) is UniEx-reducible to \( D_2 \).

Moreover, as has been demanded in advance, the resulting reducibility relation is transitive. That means, if \( D_1, D_2, D_3 \) are description sets, such that \( D_1 \) is UniEx-reducible to \( D_2 \) and \( D_2 \) is UniEx-reducible to \( D_3 \), then \( D_1 \) is UniEx-reducible to \( D_3 \). To verify this, assume \( D_1 \) is UniEx-reducible to \( D_2 \) via \( \Theta_1 \) and \( \Xi_1 \); \( D_2 \) is UniEx-reducible to \( D_3 \) via \( \Theta_2 \) and \( \Xi_2 \). If a meta-operator \( \Theta \) is defined by \( \Theta(\delta, f) = \Theta_1(\Theta_2(\delta, f)) \) for all \( \delta, f \in R \) and an operator \( \Xi \) is given by \( \Xi(\sigma) = \Xi_1(\Xi_2(\sigma)) \) for all sequences \( \sigma \), then \( D_1 \) is UniEx-reducible to \( D_3 \) via \( \Theta \) and \( \Xi \).

In order to define a reducibility relation in the sense of Definition 13 also for strong meta-learning, it is first of all inevitable to adapt the notion of Ex-admissible sequences correspondingly. The required definition is quite straightforward.

**Definition 14.** Let \( d \in \mathbb{N} \) be any description and let \( f \in R_d \). An infinite sequence \( \sigma \) of natural numbers is called resUniEx-admissible for \( d \) and \( f \), iff \( \sigma \) converges to a \( \phi^d \)-program for \( f \).

Adapting the formalism of intrinsic complexity for strong uniform learning, we have to be careful concerning the operator \( \Xi \). In UniEx-learning, the current description \( d \) has no effect on whether a sequence is admissible for a function or not. For strong uniform learning this is different. Therefore, to communicate the relevant information to \( \Xi \), it is inevitable to include a description from \( D_2 \) in the input of \( \Xi \). That means, \( \Xi \) should operate on pairs \( (\delta_2, \sigma) \) rather than on sequences \( \sigma \) only. Since only the limit of the function output by \( \Xi \) is relevant for the reduction, this idea can be simplified. It suffices, if \( \Xi \) operates correctly on the inputs \( d_2 \) and \( \sigma \), where \( d_2 \) is the limit of \( \delta_2 \). Then an operator on the pair \( (\delta_2, \sigma) \) is obtained from \( \Xi \) by returning the sequence \( (\Xi(\delta_2(0)\sigma[0]), \Xi(\delta_2(1)\sigma[1]), \ldots) \). Its limit will equal the limit of \( \Xi(d_2) \).

**Definition 15.** Let \( D_1, D_2 \subseteq \mathbb{N} \) be description sets in resUniEx. Fix a recursive meta-operator \( \Theta \) and a recursive operator \( \Xi. D_1 \) is resUniEx-reducible to \( D_2 \) via \( \Theta \) and \( \Xi \), iff for any any description \( d_1 \in D_1 \), any function \( f_1 \in R_{d_1} \), and any finite tuple \( \delta_1 \) over \( \mathbb{N} \) there is a sequence \( \delta_2 \) and a function \( f_2 \) satisfying the following properties:

1. \( \Theta(d_1, f_1) = (\delta_2, f_2) \);
2. \( \delta_2 \) converges to some description \( d_2 \in D_2 \) such that \( f_2 \in R_{d_2} \);
3. if \( \sigma \) is a resUniEx-admissible sequence for \( d_2 \) and \( f_2 \), then \( \Xi(\delta_2(\sigma)) \) is resUniEx-admissible for \( d_1 \) and \( f_1 \).

\( D_1 \) is resUniEx-reducible to \( D_2 \), iff there exist a recursive meta-operator \( \Theta \) and a recursive operator \( \Xi \), such that \( D_1 \) is resUniEx-reducible to \( D_2 \) via \( \Theta \) and \( \Xi \).
Though the definitions of UniEx-reducibility and resUniEx-reducibility are a bit involved, the conception of complete description sets can be adapted from the usual definitions in the obvious way. That means, a description set $D \subseteq \mathbb{N}$ is UniEx-complete, iff $D \in \text{UniEx}$ and each description set in UniEx is UniEx-reducible to $D$. $D$ is resUniEx-complete, iff $D \in \text{resUniEx}$ and each description set in resUniEx is resUniEx-reducible to $D$. Moreover, note that the resUniEx-reducibility relation is transitive, as well.

The transitivity of the reducibility relations of Definitions 13 and 15 is the key fact in establishing the following lemma.

**Lemma 16.** Let $D_1, D_2 \subseteq \mathbb{N}$:

1. Assume $D_1, D_2 \in \text{UniEx}$. If $D_1$ is UniEx-complete and UniEx-reducible to $D_2$, then $D_2$ is UniEx-complete.
2. Assume $D_1, D_2 \in \text{resUniEx}$. If $D_1$ is resUniEx-complete and resUniEx-reducible to $D_2$, then $D_2$ is resUniEx-complete.

**Proof.** Assertion (1) follows from transitivity of UniEx-reducibility: as $D_1$ is UniEx-complete, each description set in UniEx is UniEx-reducible to $D_1$, and thus—with transitivity—also UniEx-reducible to $D_2$. Similarly, Assertion (2) follows from transitivity of resUniEx-reducibility. \( \Box \)

As soon as we have found an example of a UniEx-complete (or resUniEx-complete) description set, this lemma can be used to verify the UniEx-completeness (or resUniEx-completeness) of others.

It remains to note that, as in the non-uniform context, hardest (i.e., complete) learning problems do not necessarily require a meta-learner which is intuitively hard to define. By definition, a uniform learning problem is hardest, if its structural properties allow for transforming its solution into solutions for all other solvable learning problems. The purpose of the following analysis is to characterise these structural properties.

### 5.2. Examples of complete description sets

The previous section has provided all the notions we need to study intrinsic complexity of uniform learning. Let us first illustrate the new notions with an example.

This example states that there is a description $d$ of an Ex-complete set, such that the description set $\{d\}$ is UniEx-complete. On the one hand, this might be surprising, because a description set consisting of just one index representing an Ex-learnable class might be considered rather simple and thus not complete for uniform learning. But, on the other hand, this result is not contrary to the intuition that the hardest problems in non-uniform learning may remain hardest, when considered in the context of meta-learning. The reason is that the complexity is still of highest degree, if the corresponding class of recursive functions is not decomposed into several recursive cores appropriately. Intuitively this reflects the trivial fact that, in general, the complexity of a problem does not decrease unless the problem is split up into several subproblems.

**Example 17.** Let $d \in \mathbb{N}$ fulfil $R_d = C_{\text{zero}}$. Then the set $\{d\}$ is UniEx-complete.

**Proof.** Obviously, the set $\{d\}$ is UniEx-learnable. Thus it has to be proven that each description set in UniEx is UniEx-reducible to $\{d\}$.

For that purpose fix a description set $D_1 \in \text{UniEx}$ and some acceptable numbering $\tau$. By Remark 4, there is a total meta-IIM $M$, such that $M_{d_1}$ is an Ex-learner for $R_{d_1}$ with respect to $\tau$, whenever $d_1 \in D_1$. It remains to define a recursive meta-operator $\Theta$ and a recursive operator $\Xi$ witnessing that $D_1$ is UniEx-reducible to $\{d\}$.

The idea for the recursive meta-operator $\Theta$ operating on $\delta_1$ and $f_1$ is to return a pair $(d^\infty, f_2)$. $f_2$ results from a modification of the sequence of hypotheses returned by the meta-IIM $M$, if the first parameter of $M$ is gradually taken from the sequence $\delta_1$ and the second parameter of $M$ is gradually taken from the sequence of initial segments of $f_1$. The intended modification is to turn each hypothesis agreeing with its predecessor into zero. Thus each converging sequence of hypotheses will be turned into a finite variant of the zero function. In order to allow for coding the hypotheses of $M$ into the obtained sequence, each hypothesis different from its predecessor is increased by 1.
Given $\delta_1, f_1 \in \mathcal{R}$, let $\Theta(\delta_1, f_1) = (d^\infty, f_2)$, where $f_2$ is defined as follows. For any $n \in \mathbb{N}$:

- compute $M_{\delta_1(n)}(f_1[n])$;
- if $n > 0$ and $M_{\delta_1(n)}(f_1[n]) = M_{\delta_1(n-1)}(f_1[n-1])$, then let $f_2(n) := 0$;
- otherwise let $f_2(n) := M_{\delta_1(n)}(f_1[n]) + 1$.

The idea for an operator $\Xi$ operating on a sequence $\sigma$ is to seek for the limit $i$ of $\sigma$ and—assuming $\tau_i \in C_{\text{zero}}$—to search for the last non-zero value $j + 1$ of $\tau_i$. Thereby $\Xi$ returns a sequence converging to $j$.

Given $n \in \mathbb{N}$ and $\sigma \in \mathcal{R}$, define $\Xi(\sigma[n])$ as follows:

- Compute $\tau_{\sigma(n)}(x)$ for all $x \leq n$, for $n$ steps each. Let $z \leq n$ be maximal, such that $\tau_{\sigma(n)}(z)$ is defined within $n$ steps and $\tau_{\sigma(n)}(z) > 0$.
- If the value $z$ does not exist, let $\Xi(\sigma[n]) := \Xi(\sigma[n-1])$ (\Xi maps the empty sequence to the empty sequence).
- If the value $z$ exists, let $q = \tau_{\sigma(n)}(z) - 1$ and $\Xi(\sigma[n]) := \Xi(\sigma[n-1])q$.

To verify that $D_1$ is UniEx-reducible to $\{d\}$ via $\Theta$ and $\Xi$, fix $d_1 \in D_1$, $f_1 \in \mathcal{R}_d$, and a finite sequence $\delta_1$.

First, since $M_{\delta_1}$ is an Ex-learner for $f_1$ respecting $\tau$, the sequence of hypotheses $M_{\delta_1}(f_1[n])$, $n \in \mathbb{N}$, converges to some $j$ with $\tau_j = f_1$. So $\Theta(\delta_1 d^\infty, f_1) = (d^\infty, f_2)$, where $f_2 = f_2[k](j+1)0^\infty$ for some $k \in \mathbb{N}$. In particular, $f_2 \in C_{\text{zero}} = \mathcal{R}_d$.

Second, if $\sigma$ is Ex-admissible for $f_2$, then $\sigma$ converges to some $i$ satisfying $\tau_i = f_2 = f_2[k](j+1)0^\infty$. If $z \in \mathbb{N}$ is maximal with $\tau_i(z) > 0$, then $\tau_i(z) = j + 1$, so $\Xi(\sigma)$ converges to $\tau_i(z) - 1 = j$. Hence $\Xi(\sigma)$ is Ex-admissible for $f_1$.

So $D_1$ is UniEx-reducible to $\{d\}$ and finally $\{d\}$ is UniEx-complete. \(\square\)

Now with the help of Example 17 and Lemma 16 we can prove the UniEx-completeness of other description sets. Note that, as in Example 17, the UniEx-complete description sets below have in common that the union of all recursive cores described contains an Ex-complete class. It will turn out that this condition is necessary for UniEx-completeness, see Corollary 24.

**Example 18.** (1) Let $(z_i)_{i \in \mathbb{N}}$ be an r.e. family of all initial segments. Let $e \in \mathcal{R}$ fulfil $\varphi^e_{0}(i) = z_i0^\infty$ and $\varphi^e_{x+1}(i) = \uparrow^\infty$ for $i, x \in \mathbb{N}$. Then the description set $\{e(i) \mid i \in \mathbb{N}\}$ is UniEx-complete.

(2) Let $\tau$ be an acceptable numbering. Let $e \in \mathcal{R}$ fulfil $\varphi^e_{0}(i) = \tau_i$ and $\varphi^e_{x+1}(i) = \uparrow^\infty$ for $i, x \in \mathbb{N}$. Then the description set $\{e(i) \mid i \in \mathbb{N}\}$ is UniEx-complete.

**Proof.** Subsequently, both assertions are verified with the help of Lemma 16, using the fact that the description set $\{d\}$ in Example 17 is UniEx-complete.

ad (1). Obviously, the set $\{e(i) \mid i \in \mathbb{N}\}$ is UniEx-identifiable. In order to prove that this set is even UniEx-complete, we reduce the set $\{d\}$ from Example 17 to $\{e(i) \mid i \in \mathbb{N}\}$. As $\{d\}$ is UniEx-complete, Lemma 16 then implies that $\{e(i) \mid i \in \mathbb{N}\}$ is UniEx-complete, too.

Given $\delta_1, f_1 \in \mathcal{R}$, let $\Theta(\delta_1, f_1) = (\delta_2, f_1)$, where $\delta_2$ is defined as follows. For any $n \in \mathbb{N}$:

- compute the minimal index $i$, such that $z_i \subseteq f_1[n]$ is the longest initial segment of $f_1[n]$ not ending with 0; i.e., $f_1[n] = z_i0^y$ for some $y \in \mathbb{N}$, and there is no $\beta$ satisfying $f_1[n] = \beta0^{y+1}$;
- let $\delta_2(n) := e(i)$.

Clearly, the output of $\Theta$ depends only on the second input parameter. If $f_1[n]$ is a finite tuple not ending with 0, then $\Theta(\delta_1, f_1[n]0^\infty)$ equals $(\delta_2, f_1[n]0^\infty)$, where $\delta_2$ is a sequence converging to some number $e(i)$ satisfying $z_i = f_1[n]$.

Moreover, let $\Xi$ be the identity operator, i.e., $\Xi(\sigma) = \sigma$ for all sequences $\sigma$.

Obviously, $\Theta$ is a recursive meta-operator and $\Xi$ is a recursive operator. It remains to verify that $\{d\}$ is UniEx-reducible to $\{e(i) \mid i \in \mathbb{N}\}$ via $\Theta$ and $\Xi$.

Fix $\delta_1 \in \mathcal{R}$ and $f_1 \in \mathcal{R}_d$.

First, there exists some $n \in \mathbb{N}$, such that $f_1[n]$ does not end with 0 and $f_1 = f_1[n]0^\infty$. Let $i$ be the minimal index of $f_1[n]$ in our fixed family, i.e., $z_i = f_1[n]$. By definition, $\Theta(\delta_1, f_1) = (\delta_2, f_1) = (\delta_2, z_i0^\infty)$, where $\delta_2$ is a sequence converging to $e(i)$. Moreover, the function $f_1 = z_i0^\infty$ belongs to $\mathcal{R}_{\delta_1}$(i).

Second, assume $\sigma$ is an Ex-admissible sequence for $f_1$. Then obviously, $\Xi(\sigma) = \sigma$ is an Ex-admissible sequence for $f_1$.

Hence $D_1$ is UniEx-reducible to the set $\{d\}$ via $\Theta$ and $\Xi$. This finally proves Assertion (1).
By definition, the function 

\[ \varphi_x^{(h(i))} = \tau_{h(i)} = \alpha_i 0^\infty \]

is \(\alpha_i\)-identifiable. Thus it remains to prove that every description set in \(\text{resUniEx}\) is \(\text{UniEx}\)-reducible to \(\{e(i) \mid i \in \mathbb{N}\}\). For that purpose let \(D_1\) be any description set in \(\text{resUniEx}\). Then there exists a total meta-IIM \(M\), such that \(M_{d_1}\) is a successful \(\text{Ex}\)-learner for \(R_{d_1}\) with respect to \(\varphi^{d_1}\), whenever \(d_1\) belongs to \(D_1\). With the help of \(M\) it is possible to define appropriate operators.

Given \(\delta_1, f_1 \in R\), let \(\Theta(\delta_1, f_1) = (\delta_2, r)\), where \(\delta_2\) is defined by \(\delta_2(n) := e(M_{\delta_1(n)}(f_1(n)))\) for all \(n \in \mathbb{N}\). Clearly, if \(\delta_1\) converges to some description \(d_1 \in D_1\) and \(f_1\) belongs to \(R_{d_1}\), then the sequence of descriptions in the first output component of \(\Theta(\delta_1, f_1)\) converges to \(e(i)\), where \(i\) is the limit hypothesis of \(M_{d_1}\) on \(f_1\).

Moreover, let \(\Xi\) be the identity operator, i.e., \(\Xi(\sigma) = \sigma\) for all sequences \(\sigma\).

Obviously, \(\Theta\) is a recursive meta-operator and \(\Xi\) is a recursive operator. Next we establish that \(D_1\) is \(\text{resUniEx}\)-reducible to \(\{e(i) \mid i \in \mathbb{N}\}\) via \(\Theta\) and \(\Xi\):

Fix a finite tuple \(\delta_1\) over \(\mathbb{N}\). Let \(d_1 \in D_1\) and \(f_1 \in R_{d_1}\).

First, \(M_{d_1}\) is an \(\text{Ex}\)-learner for \(f_1\) with respect to \(\varphi^{d_1}\), so \(M_{d_1}(f_1[n])\) is a \(\varphi^{d_1}\)-number for \(f_1\) for all but finitely many \(n \in \mathbb{N}\). Hence \(\Theta(\delta_1 d_1^\infty, f_1) = (\delta_2, r)\), where \(\delta_2\) converges to \(e(i)\) for some \(\varphi^{d_1}\)-program \(i\) for \(f_1\). In particular, \(r\) belongs to \(R_{e(i)}\).

Second, if \(\sigma\) is a \(\text{resUniEx}\)-admissible sequence for \(e(i)\) and \(r\), then \(\sigma\) converges to the index \(i\). This is the only possibility, because \(i\) is the only \(\varphi^{d_1}\)-number for the function \(r\). Therefore also \(\Xi(e(i)\sigma)\) converges to \(i\), which is a \(\varphi^{d_1}\)-number for \(f_1\). Hence \(\Xi(e(i)\sigma)\) is \(\text{resUniEx}\)-admissible for \(d_1\) and \(f_1\).

All in all, \(D_1\) is \(\text{resUniEx}\)-reducible to \(\{e(i) \mid i \in \mathbb{N}\}\) via \(\Theta\) and \(\Xi\). As \(D_1 \in \text{resUniEx}\) was chosen arbitrarily, this proves the claim. □

Thus—just as Example 18 shows that description sets for singletons can form hardest problems in \(\text{UniEx}\)-learning—Example 20 shows that description sets for a fixed singleton can form hardest problems in \(\text{resUniEx}\)-learning, cf. Theorem 5. So the definition of \(\text{resUniEx}\)-reducibility explains the fact that such description sets can often be used to illustrate difficulties in strong uniform learning, see [26].
Corollary 21. Let $D \in \text{resUniEx}$ and $r \in \mathcal{R}$. Then $D$ is resUniEx-reducible to some subset of $\{d \in \mathbb{N} \mid \mathcal{R}_d = \{r\}\}$.

5.3. Characteristic properties of complete description sets

Considering the above examples of UniEx-complete description sets one observes an interesting connection to intrinsic complexity in the non-uniform case: Examples 17 and 18 have illustrated that UniEx-complete description sets may represent a decomposition of a superclass of an Ex-complete class. In other words, the union of all recursive cores described by a UniEx-complete set may contain an Ex-complete class. Indeed, one can prove that the latter condition is necessary for UniEx-completeness, that is, each UniEx-complete description set represents a decomposition of a superclass of an Ex-complete class.\footnote{For now, we omit the proof of this statement—it will be an immediate consequence of Corollary 24 below.} The next theorem provides us with a helpful criterion for deciding the UniEx-completeness of a given description set.

Theorem 22. Let $D \in \text{UniEx}$. $D$ is UniEx-complete, iff there is a recursive numbering $\psi$ and a limiting r.e. family $(d_i)_{i \in \mathbb{N}}$ of descriptions in $D$ such that

1. $\psi_i \in \mathcal{R}_d$ for all $i \in \mathbb{N}$;
2. For each $i, n \in \mathbb{N}$ there are infinitely many $j \in \mathbb{N}$ satisfying $\psi_i =_n \psi_j$ and $\psi_i \neq \psi_j$.

Proof. Fix a description set $D$ in UniEx.

Necessity: Assume $D$ is UniEx-complete. Fix any one–one recursive numbering $\chi$, such that $\mathcal{P}_\chi$ equals the class $C_{\text{zero}}$ of all recursive finite variants of the zero function. Moreover, fix a recursive function $e$ which, given any $i, x \in \mathbb{N}$, fulfills $\varphi^{e(i)}_0 = \chi_i$ and $\varphi^{e(i)}_x = \uparrow^\infty$, if $x > 0$. Then the description set $\{e(i) \mid i \in \mathbb{N}\}$ is UniEx-complete, as can be verified by similar means as in the proof of Example 18. The choice of $D$ then implies that $\{e(i) \mid i \in \mathbb{N}\}$ is UniEx-reducible to $D$, say via $\Theta$ and $\mathcal{S}$.

Fix any one–one r.e. family $(\chi_i)_{i \in \mathbb{N}}$ of all finite tuples over $\mathbb{N}$. Given $i \in \mathbb{N}$, $i = (x, y)$, define $(\delta_i, \psi_i) := \Theta(\chi_y e(x)^\infty, \chi_x)$. By definition of reducibility, $\psi$ is a recursive numbering and, for all $i \in \mathbb{N}$, the sequence $\delta_i$ converges to some $d_i \in D$, such that $\psi_i \in \mathcal{R}_{d_i}$. Hence $(d_i)_{i \in \mathbb{N}}$ is a limiting r.e. family of descriptions in $D$.

It remains to verify the two required properties.

ad (1). This assertion has already been verified; it follows immediately from the definition of $\psi$.

ad (2). Now fix $i, n \in \mathbb{N}$. If $i = (x, y)$, we obtain $\Theta(\chi_y e(x)^\infty, \chi_x) = (\delta_i, \psi_i)$. As $\Theta$ is a recursive meta-operator, there must be some number $m \in \mathbb{N}$, such that $\Theta(\chi_y e(x)^m, \chi_x[m]) = (\delta'_i, \beta)$ for sequences $\delta'_i \subseteq \delta_i$ and $\psi_i[n] \subseteq \beta \subseteq \psi_i$.

Because of the particular properties of $\chi$, there exists some $x' \in \mathbb{N}$, $x' \neq x$, such that $\chi_{x'} =_m \chi_x$, but $\chi_x \neq \chi_{x'}$. Moreover, there is some $y' \in \mathbb{N}$, such that $\chi_{y'} = \chi_y e(x)^m$. Now if $j = (x', y')$, this yields $\Theta(\chi_y e(x)^m e(x')^\infty, \chi_{x'}) = (\delta_j, \psi_j)$, where $\beta \subseteq \psi_j$. In particular, $\psi_j =_n \psi_j$.

Assume $\psi_j = \psi_j$. Suppose $\sigma$ is any Ex-admissible sequence for $\psi_j$. Then $\sigma$ is Ex-admissible for $\psi_j$. This implies that $\mathcal{E}(\sigma)$ is Ex-admissible for both $\chi_x$ and $\chi_{x'}$. As $\chi_x \neq \chi_{x'}$ for the one–one numbering $\chi$, this is impossible. So $\psi_j \neq \psi_j$.

We have shown that, for all $i, n \in \mathbb{N}$, there is some $j \in \mathbb{N}$ with $\psi_i =_n \psi_j$ and $\psi_i \neq \psi_j$. This implies, in addition, that for each $i, n \in \mathbb{N}$ there must be infinitely many such numbers $j$.

Sufficiency: Assume $D, \psi$, and $(d_i)_{i \in \mathbb{N}}$ fulfil the conditions of Theorem 22. Let $\mathcal{A}$ denote the numbering associated to the limiting r.e. family $(d_i)_{i \in \mathbb{N}}$. The results of the previous section, respecting intrinsic complexity of non-uniform learning, will help to show that $D$ is UniEx-complete.

By assumption, $\mathcal{P}_\psi$ is an infinite r.e. set of recursive functions, such that for each $i, n \in \mathbb{N}$ there are infinitely many $j \in \mathbb{N}$ satisfying $\psi_i =_n \psi_j$ and $\psi_i \neq \psi_j$. Theorem 11 then implies that $\mathcal{P}_\psi$ is Ex-complete. As the class $C_{\text{zero}}$ of all recursive finite variants of the zero function is Ex-learnable (even Ex-complete), $C_{\text{zero}}$ is Ex-reducible to $\mathcal{P}_\psi$. Let $\Theta', \mathcal{S}'$ be the corresponding recursive operators according to Definition 9.

With the help of $\Theta'$ and $\mathcal{S}'$ it is possible to show that the description set $\{d\}$ from Example 17 is UniEx-reducible to $D$. Since $\{d\}$ is UniEx-complete and $D$ belongs to UniEx, this implies that $D$ is UniEx-complete, too. Note that $\mathcal{R}_d = C_{\text{zero}}$, that means, the recursive core described by $d$ equals the class of recursive finite variants of the zero function.
So it remains to define $\Theta$ and $\Xi$ appropriately, where $\Theta$ is a recursive meta-operator and $\Xi$ a recursive operator.

Given $\delta_1, f_1 \in \mathcal{R}$, let $\Theta(\delta_1, f_1) = (\delta_2, \Theta'(f_1))$, where $\delta_2$ is defined as follows: Assume $f_2 = \Theta'(f_1)$. For any $n \in \mathbb{N}$,

- let $i_n$ be minimal, such that $f_2[n] \subseteq \psi_{i_n}$;
- let $\delta_2(n) := \overline{d}_{i_n}(n)$.

Moreover, let $\Xi := \Xi'$. Obviously, $\Theta$ is a recursive meta-operator and $\Xi$ is a recursive operator. Finally, we verify that $\{d\}$ is $\text{UniEx}$-reducible to $D$:

Fix $\delta_1 \in \mathcal{R}$, $f_1 \in \mathcal{R}_d$.

First, note that $f_2 = \Theta'(f_1) \in \mathcal{P}_\psi$. Let $i$ be the minimal $\psi$-number of $f_2$. As $\psi$ is a recursive numbering, for all $n \in \mathbb{N}$ the minimal number $i_n$ satisfying $f_2[n] \subseteq \psi_{i_n}$ can be computed. Additionally, $i_n$ will equal $i$ for all but finitely many $n \in \mathbb{N}$. Note that $\overline{d}_{i_1}(n)$ will equal $d_i$ for all but finitely many $n \in \mathbb{N}$. Hence, $\Theta(\delta_1, f_1) = (\delta_2, f_2)$, where $f_2 \in \mathcal{P}_\psi$ and $\delta_2$ converges to $d_i$, given $f_2 = \psi_i$. In particular, $f_2$ belongs to $\mathcal{R}_{d_i}$.

Second, if $\sigma$ is $\text{Ex}$-admissible for $f_2$, then $\Xi(\sigma)$ is $\text{Ex}$-admissible for $f_1$ and thus $\Xi(\sigma)$ is $\text{Ex}$-admissible for $f_1$.

Hence $\{d\}$ is $\text{UniEx}$-reducible to $D$ via $\Theta$ and $\Xi$, and finally $D$ is $\text{UniEx}$-complete. \hfill $\Box$

The following example illustrates how Theorem 22 can be utilised to simplify the verification of completeness of particular description sets in uniform learning.

**Example 23.** Fix a recursive numbering $\psi$ such that for each $i, n \in \mathbb{N}$ there are infinitely many $j \in \mathbb{N}$ satisfying $\psi_i =_n \psi_j$ and $\psi_i \neq \psi_j$. Let $e \in \mathcal{R}$ fulfil $\phi_0^{e(i)} = \psi_j$ and $\phi_{n+1}^{e(i)} = \uparrow$ for $i, x \in \mathbb{N}$. Then $\{e(i) \mid i \in \mathbb{N}\}$ is $\text{UniEx}$-complete.

**Proof.** Let $D := \{e(i) \mid i \in \mathbb{N}\}$ and define $d_i := e(i)$ for all $i \in \mathbb{N}$. Note that $(d_i)_{i \in \mathbb{N}}$ is an r.e. family and thus in particular a limiting r.e. family.

Apparently, $D, \psi$, and $(d_i)_{i \in \mathbb{N}}$ fulfil the conditions of Theorem 22. Hence $D$ is $\text{UniEx}$-complete. \hfill $\Box$

Moreover, we will use Theorem 22 to establish that the description set from Example 20 is not $\text{UniEx}$-complete. Thus we have a description set, which is $\text{resUniEx}$-complete, but not $\text{UniEx}$-complete. To verify this, let $r, e \in \mathcal{R}$ such that $\phi_i^{e(i)} = r$ and $\phi_{n+1}^{e(i)} = \uparrow$, if $i, x \in \mathbb{N}, x \neq i$. Assume $\{e(i) \mid i \in \mathbb{N}\}$ is $\text{UniEx}$-complete. Then Theorem 22 implies the existence of some recursive numbering $\psi$, such that

- $\mathcal{P}_\psi$ is contained in the union of all recursive cores $\mathcal{R}_{\psi(i)}$ for $i \in \mathbb{N}$, and
- for each $i, n \in \mathbb{N}$ there are infinitely many $j \in \mathbb{N}$ satisfying $\psi_i =_n \psi_j$ and $\psi_i \neq \psi_j$.

Since the recursive cores described by $\{e(i) \mid i \in \mathbb{N}\}$ consist only of the function $r$, this is impossible. Hence the initial assumption must be wrong. Therefore $\{e(i) \mid i \in \mathbb{N}\}$ is not $\text{UniEx}$-complete.

Obviously, there is a connection between $\text{Ex}$-completeness and $\text{UniEx}$-completeness concerning the characteristic properties of complete classes and complete description sets. The following alternative version of Theorem 22 emphasises this connection a bit more.

**Corollary 24.** Let $D \in \text{UniEx}$. $D$ is $\text{UniEx}$-complete, iff there is a recursive numbering $\psi$ and a limiting r.e. family $(d_i)_{i \in \mathbb{N}}$ of descriptions in $D$ such that

1. $\psi_i \in \mathcal{R}_{d_i}$ for all $i \in \mathbb{N}$;
2. $\mathcal{P}_\psi$ is $\text{Ex}$-complete.

**Proof.** Let $D$ be a description set in $\text{UniEx}$.

**Necessity:** Assume $D$ is $\text{UniEx}$-complete. Then there are (i) a recursive numbering $\psi$ and (ii) a limiting r.e. family $(d_i)_{i \in \mathbb{N}}$ of descriptions in $D$, such that Properties (1) and (2) of Theorem 22 are fulfilled. Now Property (1) of Theorem 22 says that $\psi_i \in \mathcal{R}_{d_i}$ for all $i \in \mathbb{N}$. Applying Theorem 11 to Property (2) of Theorem 22 implies that $\mathcal{P}_\psi$ is $\text{Ex}$-complete.

**Sufficiency:** Assume $\psi$ and $(d_i)_{i \in \mathbb{N}}$ fulfil the conditions above. Let $\overline{d}$ be a recursive numbering associated to the limiting r.e. family $(d_i)_{i \in \mathbb{N}}$. By Property (2), $\mathcal{P}_\psi$ is $\text{Ex}$-complete. In particular, by Theorem 11, there exists a recursive
numbering \( \psi' \), such that \( \mathcal{P}_{\psi'} \subseteq \mathcal{P}_\psi \) and for each \( i, n \in \mathbb{N} \) there are infinitely many \( j \in \mathbb{N} \) satisfying \( \psi'_i =_n \psi'_j \) and \( \psi'_i \neq \psi'_j \). Without loss of generality, choose \( \psi' \) such that \( \psi' \) is one–one.\(^6\)

It remains to find a limiting r.e. family \( (d'_i)_{i \in \mathbb{N}} \) of descriptions in \( D \), such that \( \psi'_i \) belongs to \( \mathcal{R}_{d'_i} \) for all \( i \in \mathbb{N} \).

For that purpose define a corresponding numbering \( \overline{d'_i} \) to be associated to such a limiting r.e. family.

For \( i, n \in \mathbb{N} \) define \( \overline{d'_i}(n) \) as follows:

- Let \( j \in \mathbb{N} \) be the minimal number satisfying \( \psi'_i =_n \psi_j \).
- (*) Note that, for all but finitely many \( n \), the index \( j \) will be the minimal \( \psi' \)-number of the function \( \psi'_i \), (*)
- Let \( \overline{d'_i}(n) := \overline{d}_j(n) \).
- (*) If \( j \) is minimal, such that \( \psi'_i = \psi_j \), then \( \overline{d'}_i \) will converge to \( d_j \). (*)

Finally, let \( d'_i \) be given by the limit of the function \( \overline{d'}_i \), in case such a limit exists.

Fix \( i \in \mathbb{N} \). Then there is some minimal index \( j \) satisfying \( \psi'_i = \psi_j \). By the notes in the definition of \( \overline{d}_j \), the limit of \( \overline{d}_j \) exists and equals \( d_j \), i.e., \( d'_i = d_j \). This implies \( d'_i \in D \). Moreover, since \( \psi_j \) belongs to \( \mathcal{R}_{d_j} \), the function \( \psi'_i \) belongs to \( \mathcal{R}_{d'_i} \).

Since \( \psi' \) and \( (d'_i)_{i \in \mathbb{N}} \) fulfill the properties required for application of Theorem 22, the set \( D \) is thus \( \text{UniEx-complete} \). \( \square \)

Thus certain decompositions of \( \text{Ex-complete} \) classes remain \( \text{UniEx-complete} \), and \( \text{UniEx-complete} \) description sets always represent decompositions of superset of \( \text{Ex-complete} \) classes.

Theorem 22 reveals a further connection to the non-uniform case: though \( \text{UniEx-complete} \) sets involve a topologically complicated structure, expressed by Property (2), this goes along with the demand for a limiting r.e. subset combined with a r.e. subset \( \mathcal{P}_\psi \) of the union of all represented recursive cores. The latter again can be seen as a simple algorithmic structure. So, as in non-uniform learning, complete learning problems can be characterised by a contrast between a simple algorithmic and a complicated topological structure. Note that the demands concerning the topological structure affect the union of all recursive cores described, whereas the demands concerning the algorithmic structure affect the particular descriptions.

Thus the intrinsic complexity of a uniform learning problem depends on both

- the intrinsic complexity of the union of all recursive cores described, and
- the particular descriptions for these recursive cores.

Both of these have to fulfil certain conditions to make a set \( \text{UniEx-complete} \). In particular, we cannot trade any demands concerning the union of all recursive cores described for any demands concerning the particular descriptions.

The properties used in the proof of Example 20 now help to formulate a characterisation of \( \text{resUniEx-completeness} \).

Of course, since there are complete sets representing just one singleton recursive core, the demand for a numbering with a recursive core which is dense with respect to the Baire metric has to be dropped. Some weaker condition is needed instead.

**Theorem 25.** Let \( D \in \text{resUniEx} \). \( D \) is \( \text{resUniEx-complete} \), iff there is a recursive numbering \( \psi \) and a limiting r.e. family \( (d_i)_{i \in \mathbb{N}} \) of descriptions in \( D \) such that

1. \( \psi_i \in \mathcal{R}_{d_i} \) for all \( i \in \mathbb{N} \);
2. for each \( i, n \in \mathbb{N} \) there are infinitely many \( j \in \mathbb{N} \) satisfying \( \psi_i =_n \psi_j \) and \( d_i \neq d_j \) or \( \psi_i \neq \psi_j \).

**Proof.** Fix a description set \( D \) in \( \text{resUniEx} \).

**Necessity:** Assume \( D \) is \( \text{resUniEx-complete} \). This implies that the description set \( \{ e(i) \mid i \in \mathbb{N} \} \) from Example 20 is \( \text{resUniEx-reducible} \) to the set \( D \), say via \( \Theta \) and \( \Xi \).

Fix any one–one r.e. family \( (z_i)_{i \in \mathbb{N}} \) of all finite tuples over \( \mathbb{N} \). Given \( i \in \mathbb{N} \), \( i = (x, y) \), define \( (\delta_i, \psi_i) := \Theta(z_i, e(x)^\infty, r) \). By definition of reducibility, \( \psi \) is a recursive numbering and, for all \( i \in \mathbb{N} \), the sequence \( \delta_i \) converges to some \( d_i \in D \), such that \( \psi_i \in \mathcal{R}_{d_i} \). Hence \( (d_i)_{i \in \mathbb{N}} \) is a limiting r.e. family of descriptions in \( D \).

It remains to verify the two required properties.

ad (1). This assertion has already been verified; it follows immediately from the definition of \( \psi \).

\(^6\) If \( \psi' \) is not one–one, then it is not hard to construct a one–one numbering enumerating \( \mathcal{P}_{\psi'} \) which fulfils the conditions of Theorem 11.
ad (2). Now fix $i, n \in \mathbb{N}$. If $i = (x, y)$, we obtain $\Theta(x, e(x)^\infty, r) = (\delta_i, \psi_i)$. As $\Theta$ is a recursive meta-operator, there must be some number $m \in \mathbb{N}$, such that $\Theta(x, e(x)^m, r) = (\delta'_i, \beta)$ for sequences $\delta'_i$ and $\beta$ satisfying $\delta'_i \subseteq \delta_i$ and $\psi_i[n] \subseteq \beta \subseteq \psi_i$.

Now choose any $x' \in \mathbb{N}$, such that $x' \neq x$. Also take $y' \in \mathbb{N}$ to be so that $x_{y'} = x_{y'} e(x)^m$. Now if $j = (x', y')$, this yields $\Theta(x_{y'} e(x)^m e(x')^\infty, r) = (\delta_j, \psi_j)$, where $\beta \subseteq \psi_j$. In particular, $\psi_j =_n \psi_i$.

Assume $d_i = d_j$ and $\psi_i = \psi_j$. Suppose $\sigma$ is any $\text{resUniEx}$-admissible sequence for $d_i$ and $\psi_i$. Then $\sigma$ is a $\text{resUniEx}$-admissible sequence for $d_j$ and $\psi_j$. This implies that $\Xi(d_i, \sigma)$ is $\text{resUniEx}$-admissible for both $e(x)$ and $r$ and $e(x')$ and $r$. As $x$ is the only $\varphi^e(x)$-number for $r$ and $x'$ is the only $\varphi^e(x')$-number for $r$, the latter is impossible. So $\psi_j \neq \psi_j$ or $d_i \neq d_j$.

Repeating this argument for infinitely many $x'$ with $x' \neq x$ yields the desired property.

Sufficiency: First note: if $D$, $\psi$, and $(d_i)_{i \in \mathbb{N}}$ fulfill the conditions of Theorem 22, then there also exist $\psi'$ and $(d'_i)_{i \in \mathbb{N}}$ fulfilling these conditions, such that $\{(d'_i, \psi'_i) : i \in \mathbb{N}\} \subseteq \{(d_i, \psi_i) : i \in \mathbb{N}\}$ and, additionally, $i \neq j$ implies $(d'_i, \psi'_i) \neq (d'_j, \psi'_j)$. This can be verified by standard recursion-theoretic methods.

So assume $D$, $\psi$, and $(d_i)_{i \in \mathbb{N}}$ fulfill the conditions of Theorem 22 and assume that $i \neq j$ implies $d_i \neq d_j$ or $\psi_i \neq \psi_j$. Let $\bar{d}$ denote the numbering associated to the limiting r.e. family $(d_i)_{i \in \mathbb{N}}$. Fix a recursive function $r$ and a one–one recursive function $e$ as in Example 20. The aim is to verify that the description set $\{e(i) : i \in \mathbb{N}\}$ from Example 20 is $\text{resUniEx}$-reducible to $D$. Lemma 16 then implies that $D$ is $\text{resUniEx}$-complete.

For that purpose fix a one–one recursive numbering $\eta$, such that $\mathcal{P}_\eta$ equals the set

$$C_{\text{const}} := \{x^i : x \text{ is a finite tuple over } \mathbb{N} \text{ and } i \in \mathbb{N}\}$$

of all recursive finite variants of constant functions.

Using a construction from Kinber et al. [17] we define an operator $\Theta'$, which maps $\mathcal{P}_\eta$ into $\mathcal{P}_\psi$. In parallel, a function $\theta$ is constructed to mark used indices.

Let $\Theta' (\eta_0) := \psi_0$ and $\theta(0) = 0$. If $i > 0$, let $\Theta' (\eta_i)$ be defined as follows.

- For all $x < i$, let $m_x$ be minimal, such that $\eta_i(m_x) \neq \eta_x(m_x)$.
- Let $m := \max\{m_x : x < i\}$.
- Let $k < i$ be minimal, such that $m_k = m$.

(* That means, among the functions $\eta_0, \ldots, \eta_{i-1}$, none agree with $\eta_i$ on a longer initial segment than $\eta_k$ does. *)

- Compute the set

$$H := \{j \in \mathbb{N} : j \notin \{\theta(0), \ldots, \theta(i-1)\} \text{ and } \psi_j =_m \Theta'(\eta_k)\}.$$  

(* $H$ is the set of unused $\psi$-numbers of functions agreeing with $\Theta' (\eta_k)$ on the first $m + 1$ values. *)

- Choose $h = \min(H)$ and return $\Theta'(\eta_i) := \psi_h$, moreover, let $\theta(i) := h$.

(* Because of Property (2), the index $h$ must exist. Moreover, since $\psi$ is recursive, $h$ can be found effectively. *)

Note that $\Theta'$ is a recursive operator mapping $\mathcal{P}_\eta$ into $\mathcal{P}_\psi$. $\theta$ is a one–one recursive function that maps each number $i$ to the index $h$, which is used in the construction of $\Theta'$ for defining $\Theta'(\eta_i) = \psi_h$. It may happen that $\Theta'(\eta_i) = \Theta'(\eta_j)$, but $\theta(i) \neq \theta(j)$ for some $i, j \in \mathbb{N}$.

It remains to define a recursive meta-operator $\Theta$ and a recursive operator $\Xi$, such that $\{e(i) : i \in \mathbb{N}\}$ is $\text{resUniEx}$-reducible to $D$ via $\Theta$ and $\Xi$.

If $\delta_1 \in \mathcal{R}$, let $\Theta(\delta_1, r) = (\delta_2, \Theta'(\delta_1))$, where $\delta_2$ is defined as follows. For each $n \in \mathbb{N}$:

- let $j_n$ be the minimal number satisfying $\eta_{j_n} =_n \delta_1$;
- let $i_n := \theta(j_n)$;
- let $\delta_2(n) := \bar{d}_{i_n}(n)$.

Clearly, the output of $\Theta$ depends only on $\delta_1$. If $\delta_1$ converges, then $\Theta(\delta_1, r) = (\delta_2, f_2)$, where $f_2 \in \mathcal{P}_\psi$ and $\delta_2$ converges to some description $d_i$, such that $i = \theta(j)$ for the minimal number $j$ satisfying $\eta_j = \delta_1$.

To define an operator $\Xi$, let $\sigma \in \mathcal{R}$ and $d \in \mathbb{N}$ any description. Then define $\Xi(d, \sigma)$ according to the following instructions.

- Let $i_0 := 0$ and $b_0 := 0$.
- For each $n \geq 1$, compute $\varphi_{\sigma (n)}^d [b_{n-1}]$ for $n$ steps. Moreover, compute $\bar{d}_{\Sigma}(n)$ and $\psi_{\Sigma}[b_{n-1}]$ for all $z \leq n$. If $\varphi_{\sigma(n)}^d [b_{n-1}]$ is defined within $n$ steps and there is some $z \leq n$ with $\bar{d}_{\Sigma}(n) = d$ and $\psi_{\Sigma}[b_{n-1}] = \varphi_{\sigma(n)}^d [b_{n-1}]$, then let $i_n$ be the
Let \( \mathcal{J}_n(n) = d \) and \( \psi_{n_i}[b_{n-1}] = \varphi_{\sigma(n)}[b_{n-1}] \). In this case, moreover, let \( b_n := b_{n-1} + 1 \).

Otherwise, let \( i_n := i_{n-1} \) and \( b_n := b_{n-1} \).

(* If \( \sigma \) converges to \( s \), then the sequence of the values \( i_n \) converges to the only number \( i \) satisfying \( d_i = d \) and \( \varphi^d_i = \psi_i \)—provided that \( i \) exists. *)

- For each \( n \in \mathbb{N} \), compute a number \( j_n \in \mathbb{N} \), such that \( \theta(j_n) = i_n \).

(* Thus, in the limit, a number \( j \) satisfying \( \theta(j) = i \) and \( \Theta'(\eta_j) = \psi_i \) is found—provided that \( j \) exists. *)

- Let \( \Xi(d) := (e^{-1}(\eta_j(0)), e^{-1}(\eta_j(1)), e^{-1}(\eta_j(2)), \ldots) \).

(* \( \Xi(d) \) converges to \( e^{-1}(l) \), where \( l \) is the limit of \( \eta_j \) with \( \Theta'(\eta_j) = \psi_i = \varphi^d_i \) and \( d_i \) equals \( d \)—provided that \( i \) and \( j \) exist. *)

As can be easily verified, \( \Theta \) is a recursive meta-operator and \( \Xi \) is a recursive operator. It remains to prove that \( \{ e(i) \mid i \in \mathbb{N} \} \) is \( \text{resUniEx-reducible to } D \) via \( \Theta \) and \( \Xi \).

Fix an infinite sequence \( \delta_1 \) converging to some value \( d \) in \( \{ e(i) \mid i \in \mathbb{N} \} \).

First, by the remarks below the definition of \( \Theta \), we obtain \( \Theta(\delta_1, r) = (\delta_2, f_2) \), where \( f_2 = \Theta'(\delta_1) \in \mathcal{P}_\psi \) and \( \delta_2 \) converges to some description \( d_i \), such that \( i = \Theta(j) \) for the minimal number \( j \) satisfying \( \eta_j = \delta_1 \). This implies \( f_2 = \psi_i \). In particular, \( f_2 \) belongs to the recursive core \( \mathcal{R}_{d_i} \).

Second, assume \( \sigma \) is \( \text{resUniEx-admissible for } d_i \) and \( \psi_i \). Note that the limit of \( \eta_j \) equals the limit of \( \delta_1 \), which is \( d \).

By the note in the definition of \( \Xi \), then \( \Xi(d_i, \sigma) \) converges to \( e^{-1}(d) \). Recall that \( e^{-1}(d) \) is the only \( \varphi^d \)-number of \( r \).

Hence, \( \Xi(d_i, \sigma) \) is \( \text{resUniEx-admissible for } d_i \) and \( r \).

All in all, the set \( \{ e(i) \mid i \in \mathbb{N} \} \) is \( \text{resUniEx-reducible to } D \) via \( \Theta \) and \( \Xi \). This finally implies that \( D \) is \( \text{resUniEx-complete} \).

Corollary 26. Let \( D \in \text{resUniEx} \). If \( D \) is \( \text{UniEx-complete} \), then \( D \) is \( \text{resUniEx-complete} \).

The previous theorems and corollaries can be used to further explicate the characteristic properties of complete description sets.

Examples 17 and 18(1) have shown that \( \text{UniEx-complete description sets may represent decompositions of } \text{Ex-complete classes} \). The following Theorem even states that this is possible for \( \text{any Ex-complete class} \). That means we can disintegrate any \( \text{Ex-complete class} \) and allocate the pieces to different recursive cores, such that some corresponding description set is \( \text{UniEx-complete} \). Intuitively, one might say that each most complex problem can be split into simple subproblems without decreasing the complexity of the overall problem. So in case one considers uniform learning as a way to cope with smaller problems in order to reduce the complexity of solving a bigger problem, one has to be careful to choose the representations of the smaller problems (here: descriptions of recursive cores) appropriately. Otherwise the resulting collection of subproblems will still form a most complex problem.

Theorem 27. Let \( C \subseteq \mathcal{R} \). Assume \( C \) is \( \text{Ex-complete} \). Then there is a set \( D \in \text{resUniEx} \) such that

1. \( C \) equals the union of all recursive cores described by \( D \),
2. \( D \) is \( \text{UniEx-complete} \) and \( \text{resUniEx-complete} \).

Proof. Let \( C \) be an \( \text{Ex-complete} \) class of recursive functions. By Theorem 11 there exists a recursive numbering \( \psi \), such that \( \mathcal{P}_\psi \subseteq C \) and for each \( i, n \in \mathbb{N} \) there are infinitely many \( j \in \mathbb{N} \) satisfying \( \psi_i =_n \psi_j \) and \( \psi_i \neq \psi_j \). (* Note that \( \mathcal{P}_\psi \) is also \( \text{Ex-complete} \). *)
The required description set $D$ can be defined as follows. First, for all $i \in \mathbb{N}$, let $d_i$ be a description, such that $\mathcal{R}_{d_i} = \{\varphi_i^d\} = \{\psi_i\}$. Apparently, the descriptions $d_i$ can be chosen such that $(d_i)_{i \in \mathbb{N}}$ is an r.e. family and thus in particular a limiting r.e. family. Second, for all $f \in C \setminus \mathcal{P}_\psi$, let $d_f$ be a description, such that $\mathcal{R}_{d_f} = \{\varphi_0^{d_f}\} = \{ f \}$. Then define

$$D := \{ d_i \mid i \in \mathbb{N} \} \cup \{ d_f \mid f \in C \setminus \mathcal{P}_\psi \}.$$ 

The meta-IIIM constantly returning zero evidences $D \in \text{resUniEx}$. Moreover, by definition, $C$ equals the union of all recursive cores described by $D$. It remains to show that $D$ is UniEx-complete and $\text{resUniEx}$-complete.

For that purpose, use Corollaries 24 and 26. By definition, $\psi$ is a recursive numbering and $(d_i)_{i \in \mathbb{N}}$ is a limiting r.e. family, such that $\psi_i \in \mathcal{R}_{d_i}$ for all $i \in \mathbb{N}$. Moreover, by the note above, $\mathcal{P}_\psi$ is Ex-complete. Corollary 24 then implies that $D$ is UniEx-complete. Applying Corollary 26 to the set $D \in \text{resUniEx}$, we obtain that $D$ is $\text{resUniEx}$-complete.  

So we know that any Ex-complete class has a UniEx-complete decomposition. Interestingly, there are also less complex decompositions for such Ex-complete classes, as Theorem 28 states. This reflects the idea that complex problems may be easier to solve when splitting them into simpler subproblems, if the representations of these subproblems are chosen appropriately.

**Theorem 28.** Let $C \subseteq \mathcal{R}$. Then there is a set $D \in \text{resUniEx}$ such that

1. $C$ equals the union of all recursive cores described by $D$,
2. $D$ is neither UniEx-complete nor $\text{resUniEx}$-complete.

**Proof.** Let $A_0, A_1, \ldots$, be a sequence of all infinite limiting r.e. sets, such that

$$\varphi_0^d \in C \quad \text{and} \quad \varphi_{x+1}^d \uparrow^\infty \quad \text{for all } i, x \in \mathbb{N} \text{ and } d \in A_i.$$ 

Let $A := \bigcup_{i \in \mathbb{N}} A_i$ and write $C = \{f_0, f_1, \ldots \}$. The aim is to construct a set $D \subset A$, such that the complement of $D$ intersects with each set $A_i, i \in \mathbb{N}$. This guarantees that $D$ does not contain any infinite limiting r.e. set. Additionally, $D$ will be defined in such a way that each function in $C$ occurs in a recursive core $\mathcal{R}_d$ for at least one description $d \in D$. $D$ is constructed in steps, as the union of sets $D_i, i \in \mathbb{N}$.

The set $D_0$ is defined according to the following instructions:

- Fix the least element $d_0^0$ of $A_0$.
- Let $D_0^0 := \{ d_0^0 \}$. (* Note that $D_0^0$ intersects with $A_0$. *)
- Let $d_0 \in A \setminus D_0^0$ be minimal, such that $f_0 \in \mathcal{R}_{d_0}$. (* $d_0$ exists, because $A$ contains infinitely many descriptions $d$ with $\varphi_0^d = f_0$. *)
- Let $D_0 := \{ d_0 \}$. (* The sets $D_0$ and $D_0^0$ are disjoint; some recursive core described by $D_0$ equals $\{ f_0 \}$. *)

Moreover, for any $k \in \mathbb{N}$, define $D_{k+1}$ as follows:

- Fix the least element $d_{k+1}^k$ of $A_{k+1} \setminus (D_k \cup D_k')$. (* $d_{k+1}^k$ has not been touched in the definition of $D_0, \ldots, D_k$ yet. *)
- Let $D_{k+1}' := D_k \cup \{d_{k+1}^k\}$. (* Note that $D_{k+1}'$ intersects with $A_{k+1}$. *)
- Let $d_{k+1} \in A \setminus D_{k+1}'$ be minimal, such that $f_{k+1} \in \mathcal{R}_{d_{k+1}}$. (* $d_{k+1}$ exists, because $A$ contains infinitely many descriptions $d$ with $\varphi_0^d = f_{k+1}$. *)
- Let $D_{k+1} := D_k \cup \{d_{k+1}\}$. (* The sets $D_{k+1}$ and $D_{k+1}'$ are disjoint; some recursive core described by $D_{k+1}$ equals $\{f_{k+1}\}$. *)

Define

$$D := \bigcup_{k \in \mathbb{N}} D_k \subset A.$$ 

Since $\varphi_{x+1}^d \uparrow^\infty$ for all $d \in D$ and $x \in \mathbb{N}$, we have $D \in \text{resUniEx}$ Moreover, $C$ equals the union of all recursive cores represented by descriptions in $D$. So it remains to prove that $D$ is not UniEx-complete and not $\text{resUniEx}$-complete.

First, suppose by way of contradiction that $D$ is $\text{resUniEx}$-complete. Then there is some limiting r.e. family $(d_i)_{i \in \mathbb{N}}$ of descriptions in $D$ and some recursive numbering $\psi$, such that

1. $\psi_i \in \mathcal{R}_{d_i}$ for all $i \in \mathbb{N}$;
2. for each $i, n \in \mathbb{N}$ there are infinitely many $j \in \mathbb{N}$ satisfying $\psi_i =_n \psi_j$ and $[d_i \neq d_j \text{ or } \psi_i \neq \psi_j]$.  

In particular, the set \( \{ (d_i, \psi_j) \mid i \in \mathbb{N} \} \) is infinite. As the complement of \( \mathcal{D} \) intersects with each set \( A_i, i \in \mathbb{N} \), the set \( \mathcal{D} \) does not contain any infinite limiting r.e. set. Therefore \( \{ d_i \mid i \in \mathbb{N} \} \) is finite. \( \text{card}(\mathcal{R}_{d_i}) = 1 \) for \( i \in \mathbb{N} \) then implies that \( \{ \psi_j \mid i \in \mathbb{N} \} \) is finite, too, and thus \( \{ (d_i, \psi_j) \mid i \in \mathbb{N} \} \) is finite—a contradiction. So \( \mathcal{D} \) is not resUniEx-complete.

Second, assume that \( \mathcal{D} \) is UniEx-complete. Corollary 26 then implies that \( \mathcal{D} \) is resUniEx-complete. As the latter has just been refuted, the set \( \mathcal{D} \) is not UniEx-complete. □

The contrast between Theorems 27 and 28 also illustrates the contrast between the algorithmic and the topological structure of UniEx-complete description sets. If \( \mathcal{C} \) is any Ex-complete class, then, on the one hand, there is a UniEx-complete description set representing a decomposition of \( \mathcal{C} \). On the other hand, there is a non-complete description set representing a decomposition of \( \mathcal{C} \). Note that, for both description sets, the topological structure of the union of all recursive cores stays the same, but the UniEx-complete description set has in a specific sense a simpler algorithmic structure. Hence, a more complicated algorithmic structure here yields a more simple learning problem!

6. Summary

We have investigated the problem of comparing learning problems in inductive inference with respect to their difficulty. Our investigation has built on the basic work on intrinsic complexity by Freivalds et al. [6] and on the subsequent results on classes that are complete under intrinsic complexity reductions (see [17,11]). We have adapted many of the key ideas of that work to the setting of uniform learning.

Our central questions have been:
- How should a suitable notion of reducibility be defined in order to express relations concerning intrinsic complexity of uniform learning?
- How can complete learning problems be characterised in this context?
- Are there any analogies between intrinsic complexity in the non-uniform approach and intrinsic complexity in the uniform approach?

We have suggested notions of reducibility for uniform learning, see Definition 13, and for strong uniform learning, see Definition 15. Besides several examples on complete learning problems in this context, we have provided characterisations of complete description sets for both models of uniform learning in Theorem 22, Corollary 24, and Theorem 25.

Our analysis has revealed several connections and analogies to intrinsic complexity in the non-uniform approach. For example we have seen that complete description sets in uniform learning may represent decompositions of Ex-complete classes. On the one hand, each Ex-complete class can be represented as the union of all recursive cores of a description set which is complete for uniform learning. On the other hand, in each complete description set for the non-restricted version of uniform learning, the union of all recursive cores contains an Ex-complete class. Especially Corollary 24 shows that the approach to intrinsic complexity for non-uniform learning is closely related to the approach to intrinsic complexity for uniform learning. A further analogy concerns the structure of complete learning problems: the characteristic properties of Ex-complete classes (namely a complicated topological structure and a simple algorithmic structure) are reflected in the characterisations of complete description sets in uniform learning.

In addition, it has turned out—in accordance with the negative results in Theorem 5—that collections of singleton recursive cores may form hardest problems in uniform learning. This meets our intuition based on many known results on uniform learning, which illustrate difficulties in uniform learning with examples of description sets representing only singleton recursive cores, see [26]. As has been indicated by Theorems 5 and 6, the difficulty of a learning problem in uniform inductive inference does not only depend on the difficulty of the recursive cores or their unions, but is strongly influenced by the particular choice of the corresponding descriptions. This is illustrated firstly by several examples of complete description sets representing simple recursive cores and secondly by the contrasting results of Theorems 27 and 28.

Finally, Theorems 27 and 28 reflect the idea of uniform learning problems as reformulations of original learning problems in the non-uniform setting. Assume a very complex (non-uniform) learning problem is given. Then it might be reasonable to try to reduce the complexity of the problem by splitting the whole problem into several smaller problems and solve these uniformly. The collection of these small subproblems is then a uniform learning problem—hopefully with a decreased complexity. As Theorem 28 shows, such a decrease in complexity can always be achieved for any hardest non-uniform learning problem, if the descriptions of the subproblems are chosen suitably.
Inappropriate descriptions—as is verified in Theorem 27—will remain ineffectual concerning the complexity of the overall problem.

Thus, the results on intrinsic complexity of uniform learning reaffirm the intuitive statements on the difficulties in uniform inductive inference (for example concerning the complexity of collections of singleton recursive cores). This finally corroborates the suitability of the new reducibility notions and hence the suitability of the suggested approach to intrinsic complexity of uniform learning.

Acknowledgments

The author feels very grateful towards the anonymous referees whose really thorough reviews helped to improve the paper significantly. Moreover, thanks are due to Frank Stephan for a helpful discussion on the background of intrinsic complexity.

References