UNIVERSAL COMPUTABLY ENUMERABLE EQUIVALENCE RELATIONS

URI ANDREWS, STEFFEN LEMPP, JOSEPH S. MILLER, KENG MENG NG,
LUCA SAN MAURO, AND ANDREA SORBI

Abstract. We study computably enumerable equivalence relations (ceers), under the reducibility $R \leq S$ if there exists a computable function f such that $x \; R \; y$ if and only if $f(x) \; S \; f(y)$, for every x, y. We show that the degrees of ceers under the equivalence relation generated by $\leq$ form a bounded poset that is neither a lower semilattice, nor an upper semilattice, and its first order theory is undecidable. We then study the universal ceers. We show that 1) the uniformly effectively inseparable ceers are universal, but there are effectively inseparable ceers that are not universal; 2) a ceer R is universal if and only if $R' \leq R$, where R' denotes the halting jump operator introduced by Gao and Gerdes (answering an open question of Gao and Gerdes); and 3) both the index set of the universal ceers and the index set of the uniformly effectively inseparable ceers are Σ^0_3-complete (the former answering an open question of Gao and Gerdes).

1. Introduction

Given equivalence relations R and S over the set ω of the natural numbers, define R *reducible to* S (notation: $R \leq S$) if there exists a computable function f such that $x \; R \; y \Leftrightarrow f(x) \; S \; f(y)$, for all x, y. This reducibility can be viewed as a natural computable version of Borel reducibility (in which the reduction function is Borel), widely studied in descriptive set theory to measure the relative complexity of Borel equivalence relations on Polish spaces, see for instance the textbooks [1, 19].

The reducibility $\leq$ was introduced by Ershov (see Ershov's monograph [13], translated into German in [10, 11, 12]), while dealing with the monomorphisms of a certain category. Following Ershov, one can introduce the category of equivalence relations on ω, in which a morphism from R to S is a function $\mu : \omega/_R \longrightarrow \omega/_S$ between the corresponding quotient sets, such that there is a computable function f with $\mu([x]/_R) = [f(x)]/_S$, or in other words, $x \; R \; y \Rightarrow f(x) \; S \; f(y)$.

2010 *Mathematics Subject Classification.* 03D25.

Key words and phrases. Computably enumerable equivalence relations, effectively inseparable sets.

Lempp's research was partially supported by an AMS-Simons Foundation Collaboration Grant 209087. Miller's research was partially supported by NSF grant DMS-1001847. Sorbi's research was partially carried out while he was visiting the Department of Mathematics of the University of Wisconsin-Madison in March 2011. The second and last author would like to thank the Isaac Newton Institute for Mathematical Sciences of Cambridge, England, for its hospitality during the final phase this paper was completed.

(In fact, Ershov's primary interest is in the category of numberings, of fundamental interest in the foundations of computability theory, computable algebra, and computable model theory. One can view the category of equivalence relations as a full subcategory of the category of numberings, such that every object in the category of numberings is isomorphic to an object in the subcategory of equivalence relations.) It is then easy to see that $R \leq S$ if and only if, in the terminology of category theory, there is a monomorphism from R to S, i.e., R is a subobject of S. In recent years, there has been a sudden burst of interest in the reducibility $\leq$, either to compare it and contrast it with Borel reducibility (see for instance [8, 18, 15]), or to study the complexity of various equivalence relations, such as the isomorphism relation, on classes of computable structures (see for instance [14, 16, 17]).

We study computably enumerable equivalence relations (for short, ceers) under $\leq$. Ceers appear frequently in mathematics (for instance as equality of words in finitely presented semigroups or groups), in computable model theory (see e.g., [6], where ceers are called Σ^0_1 equivalence structures), and in the theory of numberings, where the ceers are exactly the equivalence relations corresponding to the so called *positive* numberings (in fact, in the Russian literature, ceers are often called *positive* equivalence relations, as in [9]). Applications of the reducibility $\leq$ to ceers have also been motivated by proof theoretic interests in the relation of provable equivalence in formal systems (see for instance [4, 3, 25, 33]), and in universal ceers, i.e., ceers which are complete under $\leq$ with respect to the class of all ceers. An important result along these lines was the discovery that the class of universal ceers contain distinct computable isomorphism types, one of which is constituted by the precomplete ceers, proven by Lachlan [22].

We prove some useful facts about universal ceers and the poset of degrees of ceers under the reducibility $\leq$. In particular, in Section 2 we show that the poset of degrees of ceers under $\leq$ is neither an upper semilattice nor a lower semilattice (Corollary 2.5), and has undecidable first order theory (Theorem 2.4). We then turn to the study of universal ceers, and in Section 3 we show (Corollary 3.16) that every u.e.i. ceer is universal, where a u.e.i. ceer is a nontrivial ceer (i.e., with more than one equivalence class) providing a partition of ω into sets that are uniformly effectively inseparable; this extends all known universality results for ceers, and is a natural extension to ceers of classical results stating universality of creative sets, and pairs of effectively inseparable sets. We prove that uniformity is essential to conclude universality, since in Theorem 3.19 we give an example of a ceer that is not universal, but yields a partition into effectively inseparable sets; our proof of universality of u.e.i. ceers succeeds by showing the identity of the class of u.e.i. ceers with certain classes of ceers, obtained by refining classes already known in the literature. In Section 4 we study the halting jump operation on ceers introduced by Gao and Gerdes [20]: Answering an open question raised in [20], we show (Theorem 4.3) that a ceer is universal if and only if it is bireducible with its halting jump. In the final Section 5 we study some index sets of classes of universal ceers, and answering an open question raised again by Gao and Gerdes [20], we show (Theorem 5.1) that the index set of universal ceers is Σ^0_3-complete.

1.1. **Background.** Our terminology and notations about computability theory are standard, and can be found for instance in the textbooks [29, 32]. A clear and thorough introduction to ceers is provided by [20], which is currently the most complete attempt to give a systematic study of ceers under $\leq$. For later reference, we show how ceers can be computably numbered, and approximated. We say that a numbering ν of the ceers (i.e., a function ν from ω onto the ceers) is *computable* if $\{\langle e,x,y\rangle : x\ \nu(e)\ y\}$ is a c.e. set. One natural way to number all ceers is via the following construction. For every set of numbers X, let X^* denote the equivalence relation on ω generated by X, where of course we view X as a subset of ω^2, via the Cantor pairing function. It is easy to see that there exists a computable function γ such that, $W_{\gamma(e)} = W_e^*$, for every e, and if W_e is already an equivalence relation on ω, then $W_{\gamma(e)} = W_e$. Then the numbering of all ceers, $\nu(e) = W_{\gamma(e)}$, is computable. Moreover, it is easy to see that ν is *universal*, or *principal*, i.e., for every computable numbering ρ of all ceers, there exists a computable function f such that $\rho = \nu \circ f$. (For different universal computable numberings of all ceers, see e.g., [9, 20].) Throughout the rest of the paper, we denote $R_e = \nu(e)$.

We say that a sequence $\{R^s : s \in \omega\}$ of equivalence relations on ω is a *computable approximation to* a ceer R, if

(1) the set $\{\langle x,y,s\rangle : x\ R^s\ y\}$ is computable;
(2) $R^0 = \mathrm{Id}$;
(3) for all s, $R^s \subseteq R^{s+1}$; the equivalence classes of R^s are finite; there exists at most one pair $[x]_{R^s}, [y]_{R^s}$ of equivalence classes, such that $[x]_{R^s} \cap [y]_{R^s} = \emptyset$, but $[x]_{R^{s+1}} = [y]_{R^{s+1}}$ (we say in this case that the equivalence relation *R-collapses* x and y at stage $s+1$);
(4) $R = \bigcup_t R^t$.

Lemma 1.1. *There exists a sequence $\{R_e^s : e, s \in \omega\}$ of equivalence relations such that $\{\langle e,x,y,s\rangle : x\ R_e^s\ y\}$ is computable, and the sequence $\{R_e^s : s \in \omega\}$ is a computable approximation to R_e. Therefore an equivalence relation R is a ceer if and only if R can be computably approximated. Moreover if R is a ceer and $R \setminus \{\langle x,x\rangle : x \in \omega\}$ is infinite, then one can find an approximating sequence $\{R^s : s \in \omega\}$ to R satisfying that for every s, the relation R^{s+1} is obtained from R^s by the R-collapse of exactly one pair of equivalence classes of R^s.*

Proof. Straightforward. □

2. Universal ceers and the poset of degrees of ceers

The following definition plays a crucial role in this paper:

Definition 2.1. A ceer R is *universal* if $S \leq R$, for every ceer S.

Clearly, universal ceers do exist: For instance, the ceer R where $\langle i,x\rangle\ R\ \langle j,y\rangle$ if $i = j$ and $x\ R_i\ y$, is clearly universal. The first natural example of a universal ceer was given by Ershov [9, Proposition 8.2]. Subsequently, several classes of universal ceers appeared in the literature, see e.g. [4, 25, 22]; we will propose later, Corollary 3.16, yet a new and wider class of universal ceers.

Define now $R \equiv S$ if $R \leq S$ and $S \leq R$. Denote by $\deg(R)$ the $\equiv$-equivalence class, or *degree*, of R, and define

$$\deg(R) \leq \deg(S) \Leftrightarrow R \leq S.$$

Let $\mathcal{P} = \langle \text{ceers}/_{\equiv}, \leq \rangle$ denote the poset of degrees of ceers. For every $n \geq 1$, let Id_n denote the ceer

$$x \ \mathrm{Id}_n \ y \Leftrightarrow [x \equiv y \mod n]$$

(Id_1 is also called the *trivial* ceer); moreover, let Id be the identity equivalence relation. The following information about $\mathcal{P}$ is readily available:

1. $\mathcal{P}$ is a bounded poset: The least element is given by $\deg(\mathrm{Id}_1)$; the greatest element is given by the degree of universal ceers;
2. $\mathcal{P}$ has a linearly ordered initial segment of order type $\omega + 1$,
$$\deg(\mathrm{Id}_1) < \deg(\mathrm{Id}_2) < \cdots < \deg(\mathrm{Id}_n) < \cdots < \deg(\mathrm{Id}),$$
with the mapping $n \mapsto \deg(\mathrm{Id}_{n+1})$ providing the order-theoretic isomorphism of ω with $\mathcal{I}_\omega = \{\deg(\mathrm{Id}_n) : n \geq 1\}$.
3. Every ceer with n equivalence classes lies in $\deg(\mathrm{Id}_n)$, whereas $\deg(\mathrm{Id})$ consists of all decidable ceers with infinitely many equivalence classes.
4. Every $R \in \mathcal{P} \setminus \mathcal{I}_\omega$ is an upper bound of $\mathcal{I}_\omega$, i.e.,
$$(\forall S) \left[S \in \mathcal{I}_\omega \Rightarrow S \leq R \right].$$

However, $\deg(\mathrm{Id})$ is not a lower bound of $\mathcal{P} \setminus \mathcal{I}_\omega$, as follows from Lemma 2.3 below, for which we first give a preliminary definition:

Definition 2.2. Given a set A, define R_A by

$$x \ R_A \ y \Leftrightarrow [x, y \in A \text{ or } x = y].$$

Lemma 2.3. *If A is simple, then* $\mathrm{Id} \nleq R_A$.

Proof. If $\mathrm{Id} \leq R_A$ via a computable function f, then f is 1-1, so there is at most one number a such that $f(a) \in A$. Then either $f[\omega \setminus \{a\}] \subseteq A^c$, or $f[\omega] \subseteq A^c$ if no such a exists. In either case, A^c contains an infinite c.e. set, a contradiction. □

The main result of this section is

Theorem 2.4. *The first-order theory of $\mathcal{P}$, in fact its Π^0_3-fragment, is undecidable.*

Proof. Let $\mathbf{0}_1$ denote the 1-degree of any computable, infinite and coinfinite set, and let $\mathbf{0}'_1$ denote the 1-degree of K. We first claim that $[\deg(\mathrm{Id}), \deg(R_K)] \simeq [\mathbf{0}_1, \mathbf{0}'_1]$, where $\simeq$ denotes order isomorphism between the two intervals of degrees: Notice that $\mathrm{Id} \equiv R_A$ for every computable, infinite and coinfinite set A.

The claim follows from the following two observations. First of all, if A, B are c.e. sets with B infinite, then

$$A \leq_1 B \Leftrightarrow R_A \leq R_B.$$

(This result has been independently noticed by several authors (San Mauro [24]; Coskey, Hamkins, and Miller [8]; moreover, the left-to-right implication appears in [20]).

Next, we notice that if $\mathrm{Id} \leq R \leq R_A$ then there exists a c.e. set B such that $R \equiv R_B$. To see this, if $\mathrm{Id} \leq R \leq R_A$, and $R \leq R_A$ via a computable f, then the range of f is an infinite c.e. set, and thus computably isomorphic to ω; let $g : \mathrm{range}(f) \longrightarrow \omega$ be a computable bijection. Finally take $B = g[A \cap \mathrm{range}(f)]$. Then $R_B \leq R$ via h where

$$h(x) = \mu y. [g(f(y)) = x].$$

On the other hand, $R \leq R_B$ via the computable function $g \circ f$.

Finally, the undecidability of the first order theory follows from Lachlan's result [21] that the topped finite initial segments of $[\mathbf{0}_1, \mathbf{0}_1']$ are exactly the finite distributive lattices (see also [28, p. 584]); thus the same is true of the interval $[\deg(\mathrm{Id}), \deg(R_K)]$ of $\mathcal{P}$. Hence the first-order theory of the finite distributive lattices is Σ_1-elementarily definable with parameters (see [27] for the terminology) in $\mathcal{P}$. On the other hand, the Π_3^0-theory of the finite distributive lattices is hereditarily undecidable ([27, Theorem 4.8]). Hence by the Nies Transfer Lemma, [27], the Π_3^0-theory of $\mathcal{P}$ is undecidable. □

Corollary 2.5. *$\mathcal{P}$ is neither an upper semilattice nor a lower semilattice.*

Proof. The claim follows from the isomorphism $[\deg(\mathrm{Id}), \deg(R_K)] \simeq [\mathbf{0}_1, \mathbf{0}_1']$ and the fact that the poset of c.e. 1-degrees, $[\mathbf{0}_1, \mathbf{0}_1']$, is neither an upper semilattice nor a lower semilattice, [34]. □

Additional information about $\mathcal{P}$ is provided by the following proposition. For more on $\mathcal{P}$, see also [20].

Proposition 2.6. *$\mathcal{P}$ is upwards dense and its greatest element is join-irreducible.*

Proof. Upwards density will be a corollary to Theorem 4.3. As to the other claim, suppose that R and S are ceers such that their degrees join to the greatest element. Consider the ceer $R \oplus S$, defined by

$$x \; R \oplus S \; y \Leftrightarrow \begin{cases} u \; R \; v & \text{if } x = 2u \text{ and } y = 2v, \\ u \; S \; v & \text{if } x = 2u+1 \text{ and } y = 2v+1. \end{cases}$$

Being above both R and S we have that $R \oplus S$ is universal. Let now E be a universal ceer such that there is no decidable set $X \neq \emptyset, \omega$ which is E-closed (i.e., satisfying that $x \in X$ and $y \; E \; x$ imply $y \in X$: see again Ershov [9, Proposition 8.2] for an example of such a ceer, or, more generally, this property holds of all the universal ceers given by Corollary 3.16 below), and let f be a computable function reducing $E \leq R \oplus S$. If $E \nleq R$ and $E \nleq S$ then the set $X = \{x : f(x) \text{ even}\}$ is a nontrivial decidable set that is E-closed. □

3. Universal ceers and partitions of the natural numbers into effectively inseparable sets

Effective inseparability played an important role in the early investigations on universal ceers, see e.g. [4, 25]. We recall:

Definition 3.1. Two disjoint c.e. sets A and B are *effectively inseparable* if there is a computable function p (called a *productive function*) such that, for all pairs u, v,

$$[A \subseteq W_u \text{ and } B \subseteq W_v \text{ and } W_u \cap W_v = \emptyset] \Rightarrow p(u,v) \notin W_u \cup W_v.$$

(We notice that the usual definition of an e.i. pair requires the existence of a productive function which is just partial computable, and defined on all pairs u, v such that $A \subseteq W_u$, $B \subseteq W_v$, and $W_u \cap W_v = \emptyset$; this is, however, equivalent to requiring a total productive function: See e.g., [32, p. 44].)

Since every ceer yields a partition of ω into c.e. sets, the previous definition suggests the following definition (where we recall that a ceer R is nontrivial if $R \neq \mathrm{Id}_1$):

Definition 3.2. A nontrivial ceer R is

- *effectively inseparable* (or *e.i.* for short) if it yields a partition of ω into sets that are pairwise effectively inseparable;
- *uniformly effectively inseparable* (or *u.e.i.* for short) if it is e.i. and there is a uniform productive function, i.e., a computable function $p(a,b,u,v)$ such that if $[a]_R \cap [b]_R = \emptyset$ then $p(a,b,\_,\_)$ is a productive function for the pair $\big([a]_R, [b]_R\big)$.

Ceers yielding partitions into effectively inseparable sets had been previously studied for instance in [2, 4, 33]. The main result of this section shows that every u.e.i. ceer is universal; on the other hand, there exist e.i. ceers that are not universal. The proof that u.e.i. ceers are universal proceeds by showing that the u.e.i. ceers coincide with two classes of ceers, introduced in the next two definitions, that refine known classes in the literature.

Recall that an equivalence relation R is *uniformly finitely precomplete* (or *u.f.p.* for short) if there exists a total computable function $f(e, D, x)$ such that for every finite set D and every e, x,

$$\varphi_e(x){\downarrow} \in [D]_R \ \Rightarrow \varphi_e(x) \ R \ f(e, D, x).$$

This definition is due to Montagna [25], who showed that every u.f.p. ceer is universal. (A proof of this claim also follows from Theorem 3.7 below.) Two important subclasses of u.f.p. ceers (and thus universal ceers) are provided by the precomplete ceers and the extension complete ceers. Precomplete equivalence relations had been introduced much earlier by Malcev [23]. An equivalence relation R is *precomplete* if there exists a total computable function $f(e,x)$ such that, for all e, x,

$$\varphi_e(x){\downarrow} \Rightarrow \varphi_e(x) \ R \ f(e,x).$$

Finally, *extension complete* (or, simply, *e-complete*) equivalence relations were introduced by Montagna [25] and by Lachlan [22]: The name is due to Lachlan [22], whereas Montagna called them *uniformly finitely m-complete*. Following a characterization given by Bernardi and Montagna [3], a ceer R is e-complete if R is u.f.p. and R has a total diagonal function, i.e., a computable function d such that $d(x) \not\mathrel{R} x$, for all x. The precomplete ceers and the e-complete ceers form two distinct isomorphism types (Lachlan [22] proved that all precomplete ceers are isomorphic; Montagna [25] proved that all e-complete ceers are isomorphic: see also [22]), where we say that two equivalence relations R and S are *isomorphic* if there is a computable permutation f of ω such that f reduces R to S. It is, however, worth noting that there are u.f.p ceers that are neither precomplete nor e-complete, Shavrukov [30]. Examples of precomplete, u.f.p., and e-complete ceers coming from the relation of provable equivalence in formal systems can be found in [33, 4, 25, 3].

Definition 3.3. We say that a nontrivial ceer R is *weakly u.f.p.* if there exists a total computable function $f(e, D, x)$ such that for every finite set D, where $i \not\mathrel{R} j$ for every $i, j \in D$, and every e, x,

$$\varphi_e(x)\downarrow\in [D]_R \ \Rightarrow \varphi_e(x) \mathrel{R} f(e, D, x).$$

Note that the definition differs from that of a u.f.p. ceer in that f need only satisfy the condition when $i \not\mathrel{R} j$ for every $i, j \in D$. Clearly

Corollary 3.4. *Every u.f.p. ceer is weakly u.f.p.*

Proof. Immediate. □

The following definition is a strengthening of the definition of a uniformly m-complete ceer given by Bernardi and Sorbi [4]. Namely, a nontrivial ceer R is *uniformly m-complete* (abbreviated as *u.m.c.*) if for every ceer S and every assignment $a_0 \mapsto b_0$, $a_1 \mapsto b_1$ (also denoted by $(a_0, a_1) \mapsto (b_0, b_1)$) of numbers such that $a_0 \not\mathrel{S} a_1$ and $b_0 \not\mathrel{R} b_1$, there exists a computable function extending the assignment and reducing S to R.

Definition 3.5. We say that a nontrivial ceer R is *strongly u.m.c.* if for every ceer S, every assignment $(a_0, a_1) \mapsto (b_0, b_1)$ can be extended uniformly (in a_0, a_1, b_0, b_1) to a total computable function f reducing S to R, provided that $a_0 \not\mathrel{S} a_1$ and $b_0 \not\mathrel{R} b_1$. (Note that the uniformity extends also to the cases $a_0 \mathrel{S} a_1$ or $b_0 \mathrel{R} b_1$; however, then no claim is made as to f reducing S to R.)

We call a nontrivial ceer *weakly n-u.f.p.* if Definition 3.3 for weakly u.f.p. holds, but we replace "finite set D" with "finite set D where $|D| \leq n$".

Lemma 3.6. *Each weakly* 2*-u.f.p. ceer is weakly u.f.p.*

Proof. Let f_i be a computable function witnessing that R is weakly i-u.f.p., for $2 \leq i \leq n$. We describe how to effectively get a function f_{n+1} witnessing that R is weakly $n+1$-u.f.p. Let e, D be given, with $|D| = i$. If $i > n+1$ or $i \leq 1$ then $f_{n+1}(e, D, x)$ outputs 0 for every x; if $2 \leq i \leq n$ then $f_{n+1}(e, D, x) = f_i(e, D, x)$ for every x. We assume now $D = \{d_0, \dots, d_n\}$. By the Double Recursion Theorem (see, e.g., [29, Theorem X(a)]) assume that we build φ_a and φ_b for some a, b. Let $E_x = \{f_n(a, D \setminus \{d_n\}, x), d_n\}$, and $f_{n+1}(e, D, x) = f_2(b, E_x, x)$.

We now specify, for an x, how to compute $\varphi_a(x)$ and $\varphi_b(x)$. We initially start with both values undefined. We see which event happens first: If we find that $\varphi_e(x)\downarrow R\ d_n$, we define $\varphi_b(x) = d_n$. If we find that $\varphi_e(x)\downarrow R\ d_i$ for some $i < n$, we define $\varphi_b(x) = f_n(a, D \setminus \{d_n\}, x)$ and $\varphi_a(x) = \varphi_e(x)$. Finally, if we discover that $f_n(a, D \setminus \{d_n\}, x) \mathrel{R} d_n$, then we define $\varphi_a(x) = d_0$.

Clearly f_{n+1} is a total computable function, whose index can be found effectively in the indices for $f_2, \dots, f_n$, using the fact that the fixed points in the Double Recursion Theorem can be found effectively from the parameters.

Now we verify that f_{n+1} witnesses that R is weakly $n+1$-u.f.p. Fix e, D, x such that $D = \{d_0, \dots, d_n\}$ where $d_i \not R\, d_j$ for every pair $i \neq j$, and $\varphi_e(x)\downarrow R\; d_i$ for some $i \leq n$. First we claim that $f_n(a, D \setminus \{d_n\}, x) \not R\, d_n$. Suppose otherwise: Then by construction we would set $\varphi_a(x) = d_0$ unless it has previously been defined (to be $\varphi_e(x)\ R\ d_i$, for some $i < n$). In either case we have $\varphi_a(x)\ R\ d_i$ for some $i < n$, which implies that $d_n\ R\ f_n(a, D \setminus \{d_n\}, x)\ R\ d_i$, a contradiction. We have thus that E_x consists of two elements that are not R-equivalent. Since $\varphi_b(x)$ is defined only when $\varphi_e(x)$ converges, it is straightforward to see that $f_{n+1}(e, D, x)\ R\ \varphi_e(x)$. □

Theorem 3.7. *The following properties are equivalent for ceers:*

(i) *u.e.i.*
(ii) *weakly u.f.p.*
(iii) *strongly u.m.c.*

We prove Theorem 3.7 via Lemma 3.8, Lemma 3.10, and Lemma 3.11.

Lemma 3.8. *Each u.e.i. ceer is weakly u.f.p.*

Proof. Assume that R is u.e.i. via the uniform productive function $p(a, b, u, v)$ as in Definition 3.2. We argue that R is weakly 2-u.f.p. Given any $a \neq b$, and e, we uniformly build a function $f(x) = f(e, \{a, b\}, x)$ witnessing that R is 2-u.f.p. Note that if $a = b$ then we can let f be the constant function with output a. Again by the Double Recursion Theorem with parameters we build W_{a_x}, W_{b_x} for computable sequences of indices $\{a_x\}_{x\in\omega}, \{b_x\}_{x\in\omega}$, where the sequence is known to us during the construction.

Let $f(x) = p(a_x, b_x)$, where for simplicity we denote $p(a, b, \_, \_)$ by $p(\_, \_)$. Clearly f is a total computable function. Fix x, and let

$$W_{a_x} = \begin{cases} [a]_R, & \text{if } \varphi_e(x) \not R\, b \\ [a]_R \cup \{p(a_x, b_x)\}, & \text{if } \varphi_e(x)\ R\ b, \end{cases}$$

$$W_{b_x} = \begin{cases} [b]_R, & \text{if } \varphi_e(x) \not R\, a \\ [b]_R \cup \{p(a_x, b_x)\}, & \text{if } \varphi_e(x)\ R\ a. \end{cases}$$

Now assume that $a \not R\, b$, and fix e, x such that $\varphi_e(x)\downarrow \in [a]_R \cup [b]_R$. Without loss of generality suppose $\varphi_e(x)\ R\ a$. If $f(x) \not R\, a$ then $W_{a_x} \cap W_{b_x} = \emptyset$ and $p(a_x, b_x) \in W_{a_x} \cup W_{b_x}$, which contradicts p being a productive function. □

In the proof of Lemma 3.10 below we will use a computable sequence of fixed points. Since a computable sequence of indices can be viewed as the range of a computable function f, a formal justification to our argument is provided by the Case Functional Recursion Theorem:

Lemma 3.9 (Case Functional Recursion Theorem, Case [5])**.** *Given a partial computable functional F, there is a total computable function f such that, for* *every* e, x,

$$F(f, e, x) = \varphi_{f(e)}(x).$$

Lemma 3.10. *Each weakly u.f.p. ceer is strongly u.m.c.*

Proof. Assume that R is a weakly u.f.p. ceer, as witnessed by the computable function f. In order to show that R is strongly u.m.c., we show in fact that for every ceer S, every assignment $(0, 1, \dots, m) \mapsto (a_0, a_1, \dots, a_m)$ with $m > 0$, can be extended, uniformly in $a_0, a_1, \dots, a_m$, to a total computable function inducing a reduction from S to R, provided that $i \not S j$ and $a_i \not R a_j$ whenever $i \neq j$, with $i, j \leq m$. (Uniformity extends also to the cases in which there are pairs $i \neq j$ with $i\ S\ j$, or $a_i\ R\ a_j$.) Notice that it is no loss of generality considering an assignment $(0, 1, \dots, m) \mapsto (a_0, a_1, \dots, a_m)$, instead of $(a'_0, a'_1, \dots, a'_m) \mapsto (a_0, a_1, \dots, a_m)$: Indeed, given S and $(a'_0, a'_1, \dots, a'_m) \mapsto (a_0, a_1, \dots, a_m)$, one can consider the ceer S' and the new assignment $(0, 1, \dots, m) \mapsto (a_0, a_1, \dots, a_m)$, where we have picked a computable permutation f of ω with $f(i) = a'_i$, for all $i \leq m$, and we have defined $x\ S'\ y$ if and only if $f(x)\ S\ f(y)$. Clearly, we can extend the new assignment to a reduction of S' to R if and only if we can extend the original assignment to a reduction of S to R. The definition of a strongly u.m.c. ceer is thus just the case $m = 1$.

Our goal (under the assumption that the i's are pairwise not S-equivalent, and the a_i's are not R-equivalent, for $i \leq m$) is to extend this assignment to a total computable function yielding a reduction, by specifying a computable sequence of points $(a_{m+1}, a_{m+2}, \dots)$ where for every pair i, j such that one of i or j is larger than m, we can force a_i to R-collapse to a_j, i.e., to become $a_i\ R\ a_j$. By the Recursion Theorem (or, more precisely, the Case Functional Recursion Theorem), we assume that we control φ_{e_i} for a computable sequence $\{e_i\}_{i \in \omega}$ of indices.

We will define computable arrays $\{x_i^k, y_n\}_{i,k,n \in \omega}$ with the purpose that we can choose to cause R-collapses of pairs of y's, from $\{y_n\}_{n \in \omega}$, independently. We first informally describe the uses of the elements x_i^k and y_n. For each k, we will build the element x_0^k to be R-collapsible to any element of $\{a_0, \dots, a_m\}$. (This can be achieved by defining $x_0^k = f(e_1, \{a_0, \dots, a_m\}, k)$: When we need to R-collapse x_0^k to a_j we set the previously undefined $\varphi_{e_1}(k)$ to be $\varphi_{e_1}(k) = a_j$, and use properties of f to conclude that $x_0^k\ R\ a_j$, as long as a_j is not R-equivalent to any other a_i.) These x_0^k's will be used to R-collapse any other x_i^k or y_k into $\{a_0, \dots, a_m\}$, by previously R-collapsing it to x_0^k. The role of x_i^k will be to allow y_k to R-collapse with y_i. To be more precise, for $k > 0$, y_k will be built to be R-collapsible with any element in $Y_k = \{x_1^k, \dots, x_{k-1}^k\} \cup \{x_0^{2k}\}$ through its definition as $y_k = f(e_{2k}, Y_k, 0)$, and $x_i^k \in Y_k$ will be built to be R-collapsible to any element in $\{y_i, x_0^{2i+1}\}$ through its definition as $x_i^k = f(e_{2i+1}, \{y_i, x_0^{2i+1}\}, k)$. If for instance $k > m$ and we want to R-collapse y_k to y_i since we see that i and k S-collapse (we may assume that $i < k$), then we distinguish the two possible cases: If $i \leq m$, then define $\varphi_{e_{2k}}(0) \downarrow= x_0^{2k}$ and then $\varphi_{e_1}(k) \downarrow= a_i$; if $m < i < k$, then define $\varphi_{e_{2k}}(0) \downarrow= x_i^k$ and $\varphi_{e_{2i+1}}(k) \downarrow= y_i$. We must of course make sure that these are the only R-collapses that we allow in this way for the y's, and more generally between the elements of our arrays $\{x_i^k, y_n\}_{i,k,n \in \omega}$: This can be achieved by arguing that if this were not the case then we would be eventually able to apply a special procedure, called Action $\diamondsuit$, consisting in laying down suitable additional R-collapses, that would lead to the conclusion that there is a pair of distinct $i, j \leq m$ such that $a_i \mathrel{R} a_j$. Thus, if there is no such pair we in fact do not take Action $\diamondsuit$, excluding the possibility of unwanted R-collapses.

Formally, define the computable arrays $\{x_i^k\}_{i,k\in\omega}$ and $\{y_n\}_{n\in\omega}$ as follows: Let $x_0^k = f(e_1, \{a_0, \dots, a_m\}, k)$. Given $\{x_i^k\}_{i<n+1,k\in\omega}$, define

$$y_{n+1} = f(e_{2n+2}, Y_{n+1}, 0),$$

where $Y_{n+1} = \{x_i^{n+1} \mid 0 < i < n+1\} \cup \{x_0^{2n+2}\}$, and, for $k \in \omega$,

$$x_{n+1}^k = f(e_{2n+3}, \{y_{n+1}, x_0^{2n+3}\}, k).$$

We assume (see Lemma 1.1) that during each stage of the construction, exactly one pair of distinct S-equivalence classes collapses, and we always assume that $[i]_S$ represents an S-equivalence class with smallest member i. During the construction, to *identify* y_n *with* c means to define $\varphi_{e_{2n}}(0)\downarrow = c$, and similarly to *identify* x_n^k *with* c means to define $\varphi_{e_{2n+1}}(k)\downarrow = c$.

Definition of $\{a_j\}_{j>m}$. We let $a_j = y_j$ for $j > m$. The a_j's will be the markers that code S in R, the other numbers x_i^j, y_i are simply representatives of auxiliary classes which will assist in R-collapsing the a_j's.
Construction of φ_{e_i}. During the construction, if we ever discover that $a_i \mathrel{R} a_j$, for some $i < j \leq m$, then we can ignore the rest of the construction below, and continue the construction trivially for the sake of uniformity, since the working assumption that $a_i \not\mathrel{R} a_j$, for all $i < j \leq m$, is violated. At stage s of the construction, let $[i]_S$ and $[j]_S$ be the pair of collapsing S-classes. If $i, j \leq m$, we can ignore the rest of the construction below and continue trivially for the sake of uniformity, otherwise there are two cases.

Case 1: $i \leq m < j$. We identify y_j with x_0^{2j} (clearly y_j cannot have been previously identified) and wait for either $y_j \mathrel{R} x_0^{2j}$ or two elements of Y_j to R-collapse. If the latter happens first, then take Action $\diamondsuit$, otherwise we identify x_0^{2j} with a_i, and wait for $x_0^{2j} \mathrel{R} a_i$.

Case 2: $m < i < j$. Identify y_j with x_i^j, and wait for $y_j \mathrel{R} x_i^j$ or two elements of Y_j to R-collapse. Again, if the latter happens first, take Action $\diamondsuit$, otherwise we identify x_i^j with y_i. Wait for either $x_i^j \mathrel{R} y_i$ or $y_i \mathrel{R} x_0^{2i+1}$. If the latter happens first, take Action $\diamondsuit$, otherwise we achieve $y_j \mathrel{R} y_i$.

Action $\diamondsuit$: We arrived here because we found $y_i \mathrel{R} x_0^{2i+1}$ or two elements of Y_j have R-collapsed (and no element of Y_j has previously been identified). We describe two procedures P_k and Q_k which will call each other recursively until we force an R-collapse in $a_0, \dots, a_m$.

Procedure P_k: This is called when $y_k \mathrel{R} x_0^{2k+1}$. Perform the following steps.

(Step i) Check if y_k has been previously identified. If so, then by construction $y_k \mathrel{R} a_{k'}$ for some least $k' < k$. If $k' \leq m$ then go to Step (ii). Otherwise, $y_{k'}$ has not been previously identified, and we identify $y_{k'}$ with $x_0^{2k'}$. Wait for either $y_{k'}\ R\ x_0^{2k'}$ or two elements of $Y_{k'}$ to R-collapse. In the latter case we call $Q_{k'}$ (noting that no element in $Y_{k'}$ has been previously identified since $y_{k'}$ has not), otherwise we identify $x_0^{2k'}$ with a_0 and wait for $x_0^{2k'}\ R\ a_0$. Lastly, if y_k has not been previously identified, we identify y_k with x_0^{2k} and proceed as above, where we either call Q_k or we get $x_0^{2k}\ R\ a_0$. In either case, now continue with Step (ii).

(Step ii) If this step is reached then we have $y_k\ R\ a_{k'}$ for some $k' \leq m$. Clearly x_0^{2k+1} has not previously been identified. We now identify x_0^{2k+1} with $a_{k''}$ for any $k'' \neq k'$, $k'' \leq m$. We then obtain $a_{k'}\ R\ a_{k''}$, and continue the construction trivially for the sake of uniformity.

Procedure Q_k: This is called when two (least) elements in Y_k R-collapse (and no element of Y_k has been previously identified). There are two cases.

(Case i) The two elements are x_0^{2k} and x_i^k, $0 < i < k$. Identify x_i^k with x_0^{2i+1} and wait for $x_i^k\ R\ x_0^{2i+1}$ or $y_i\ R\ x_0^{2i+1}$. In the latter case call P_i, otherwise we now identify x_0^{2i+1} (not previously identified) with a_1 and wait for the R-collapse. Now identify x_0^{2k} with a_0 and wait for the R-collapse. We succeed in forcing $a_0\ R\ a_1$, and continue the construction trivially for the sake of uniformity.

(Case ii) The two elements are x_i^k and x_j^k, $0 < i < j < k$. Follow Case i to force that either $x_i^k\ R\ a_0$ and $x_j^k\ R\ a_1$ or one of P_i or P_j is called.

This ends the description of the procedures P_k and Q_k. Suppose we arrive at this action because we found $y_i\ R\ x_0^{2i+1}$. We call P_i. On the other hand, if we arrive because two elements of Y_j have R-collapsed then we call Q_j. Clearly only finitely many different procedures can be called, and we end up provoking an R-collapse within $a_0, \ldots, a_m$.

Enforcing non-collapse. At the end of stage s, check if there exist two elements of $\{a_n, x_n^k\}_{n,k\in\omega}$ which have R-collapsed but have not yet been identified.

(Case i) The two elements are a_i and a_j for $i < j \leq m$. Continue the construction trivially for the sake of uniformity.

(Case ii) The two elements are a_i and a_j for $i \leq m < j$. Identify y_j with x_0^{2j} and wait for the desired R-collapse, where we will identify x_0^{2j} with $a_{i'}$ for any $i' \neq i$, $i' \leq m$. If we instead find that two elements of Y_j have R-collapsed, we take Action $\diamondsuit$.

(Case iii) The two elements are a_i and x_j^k for $i \leq m$. If $j > 0$ we identify x_j^k with x_0^{2j+1} and wait for the R-collapse. We then identify x_0^{2j+1} with $a_{i'}$ for any $i' \leq m$ with $i' \neq i$. If we instead find that $y_j\ R\ x_0^{2j+1}$, we take Action $\diamondsuit$. The case $j = 0$ is trivial, by immediately identifying x_0^k with $a_{i'}$.

(Otherwise) For each of the remaining cases, we can follow Case ii or Case iii to force an R-collapse in $a_0, \ldots, a_m$.

Verification. We list some easy facts about the construction.

- If any element of Y_j is identified then the same action must identify y_j, or provoke an R-collapse among $a_0, \ldots, a_m$, and thus the construction is continued trivially for the sake of uniformity.
- If y_j is identified during the construction then we will either provoke an R-collapse among $a_0, \ldots, a_m$, or force $y_j \mathrel{R} a_i$ for some $i < j$ where $i \mathrel{S} j$.
- If x_0^{2k+1} is ever identified during the construction then the same action will provoke an R-collapse among $a_0, \ldots, a_m$.
- Therefore, any call to identify y_j or x_i^j during the construction must be successful.

Now we assume that $a_0, \ldots, a_m$ are in distinct R-equivalence classes. Then the construction is never continued trivially, and we never take Action $\diamondsuit$. At the end of every stage s, we have that $i \mathrel{S} j$ if and only if $a_i \mathrel{R} a_j$. The left to right direction is ensured by Case 1 and Case 2 of the construction, while the right to left is ensured by the "enforcing non-collapse" action. □

Lemma 3.11. *Every strongly u.m.c. ceer is u.e.i.*

Proof. Let R be a strongly u.m.c. ceer. Let U, V be a fixed pair of e.i. sets, and define S to be the ceer

$$x \mathrel{S} y \Leftrightarrow \left[x = y \text{ or } x, y \in U \text{ or } x, y \in V\right].$$

Fix $u \in U$, $v \in V$, and given a, b, consider the assignment $(u, v) \mapsto (a, b)$. Using the fact that R is strongly u.m.c., uniformly extend it to a computable function $f_{a,b}$. If $[a]_R \cap [b]_R = \emptyset$, then $f_{a,b}$ uniformly m-reduces the e.i. pair (U, V) to the pair $([a]_R, [b]_R)$, showing that the latter is e.i. (for this property of e.i. pairs, see, e.g., [29]). The fact that R is u.e.i. follows from the uniformity in this argument. □

The following is a natural companion to Lemma 3.6.

Corollary 3.12. *A nontrivial ceer R is strongly u.m.c. if for every ceer S and every assignment $(a_0', a_1', \ldots, a_m') \mapsto (a_0, a_1, \ldots, a_m)$, the assignment can be extended uniformly to a total computable function giving a reduction from S to R, provided the a_i' are pairwise not S-equivalent and the a_i are pairwise not R-equivalent, for every $i \leq m$.*

Proof. By Theorem 3.7 and the proof of Lemma 3.10. □

We note that in the above corollary and in the definition of a strongly u.m.c. ceer, the condition $m > 0$ is necessary as removing this condition implies that R has a total diagonal function g (see the definition of an e-complete ceer, given earlier) defined in the following way: Given a, consider the ceer Id_2 having only two equivalence classes $[0]_{\mathrm{Id}_2}$, $[1]_{\mathrm{Id}_2}$, and extend the assignment $0 \mapsto a$ to a total computable function f inducing a reduction from Id_2 to R; finally take $g(a) = f(1)$. On the other hand, the property of having a total diagonal function is not necessarily possessed by all u.e.i. ceers, since this property characterizes the e-complete ceers within the class of all u.f.p. as we have already recalled before Definition 3.3.

Proposition 3.13. *Each strongly u.m.c. ceer is u.m.c., but there are u.m.c. ceers that are not strongly u.m.c.*

Proof. It is clear that every strongly u.m.c. ceer is u.m.c., so by Theorem 3.7 it is enough to argue that some u.m.c. R is not u.e.i. The proof proceeds by a finite injury argument, which builds a ceer R satisfying the following requirements: for every ceer R_e and assignment $(a'_0, a'_1) \mapsto (a_0, a_1)$ (where we suppose that in requirement $\mathcal{R}_e$, the assignment is also coded by e), and partial computable function φ_e:

$$\mathcal{R}_e : \text{if } a'_0 \not R_e\, a'_1 \text{ and } a_0 \not R_e\, a_1, \text{ then there is a total extension reducing } R_e \text{ to } R,$$
$$\mathcal{Q}_e : R \text{ is not u.e.i. via the function } \varphi_e.$$

The priority ordering is $\mathcal{R}_0 < \mathcal{Q}_0 < \mathcal{R}_1 < \cdots$. We use the fact that no effectivity is required in satisfying $\mathcal{R}_e$ by allowing the requirements to be injured. That is, we can change our mind about the extension, for each $\mathcal{R}_e$, finitely many times. As in the proof of Lemma 3.10, without loss of generality we assume that $a'_i = i$ for $i \leq 1$. We denote the pair (a_0, a_1) (for requirement $\mathcal{R}_e$) by $\vec{a}_e$. R will be nontrivial since R does not have any uniform productive function.

We denote by $X^{e,s}$ the (modified) i^{th} column of ω, i.e., the set of elements $X^{e,s}(j)$ with $X^{e,s}(j) = \langle i, j\rangle$ where $i = \langle e, s+1, \vec{a}_e\rangle$, if $j > 1$, and $X^{e,s}(j) = a_j$ if $j \leq 1$. (Note that thus the modified columns are not necessarily pairwise disjoint, but any possible overlap between columns are at the first two elements.) This set will be used by $\mathcal{R}_e$ to code R_e: To code R_e into $X^{e,t}$ at some stage s of the construction means to R-collapse $X^{e,t}(i)$ and $X^{e,t}(j)$ if $i\ R_e\ j$ unless $i, j \leq 1$. We write X^e instead of $X^{e,t}$ when the context is clear, and sometimes call it the *R_e-column*. For a tuple $\vec{b}$ we say that $\vec{b}\ R\ x$ if there exists some entry b_i in $\vec{b}$ such that $b_i\ R\ x$.

Construction of R. At each stage s, perform the following two steps.

Step 1: Pick the least e such that $\mathcal{Q}_e$ requires attention. This means either the pair of $\mathcal{Q}_e$-followers y_e^0, y_e^1 is not yet picked, or else $\varphi_e(y_e^0, y_e^1, z_e^0, z_e^1){\downarrow}\notin W_{z_e^0} \cup W_{z_e^1}$. Here, $W_{z_e^i}$ denotes the final R-equivalence class of y_e^i, of which we suppose to know an index by the Recursion Theorem. For convenience, when $\varphi_e(y_e^0, y_e^1, z_e^0, z_e^1)\downarrow$ we denote

$$\varphi_e(y_e^0, y_e^1, z_e^0, z_e^1) = f_e.$$

Clearly some $e < s$ must be found. If the $\mathcal{Q}_e$-followers are not defined, we pick a distinct fresh pair y_e^0, y_e^1 from $X^{0,-1}$ (in particular, larger than any element of $\vec{a}_i$ for a higher-priority $\mathcal{R}_i$-requirement, or follower y_i^j for a higher-priority $\mathcal{Q}_i$-requirement). If the second case holds we R-collapse f_e and y_e^0. In either case we initialize all lower-priority requirements, i.e., for a $\mathcal{Q}_j$-requirement we reset the followers, and for an $\mathcal{R}_j$-requirement we are now ready to code R_j into $X^{j,t}$ for a fresh number t: To prevent R_j from interfering with the other higher priority requirements, we can pick t such that $\langle j, t+1, 2\rangle >$ any element in $\vec{a}_k$, for all $k < j$.

Step 2: For each $j < s$ we code R_j into X^j.

If there exists a least $j < s$ such that the action at Step 1 or 2 collapses the two classes in $\vec{a}_j$, we make $\mathcal{R}_j$ inactive and initialize all lower-priority requirements.

We now verify that the requirements are satisfied. Clearly each requirement is initialized finitely often. The key lemma below says that during the construction, each class targeted for a $\mathcal{Q}$- or $\mathcal{R}$-requirement does not contain elements which are "bad" for the requirement.

Lemma 3.14. *Let s be a stage of the construction. The following hold at s:*

(i) *If $\vec{a}_i \not R\, X^i(n_i)$ for some n_i, then for every $x \; R \; X^i(n_i)$, we have $x \geq X^i(2)$.*
(ii) *If $f_i \not R\, y_i^{n_i}$ then for every $x \; R \; y_i^{n_i}$, we have $x \geq y_i^{n_i}$.*
(iii) *Suppose $X^i(n_i) \; R \; X^j(n_j)$ for some i, j, n_i, n_j. Then*

$$i \neq j \Rightarrow \vec{a}_i \vec{a}_j \; R \; X^i(n_i).$$

(iv) *Suppose $X^i(n_i) \; R \; y_j^{n_j}$ for some i, j, n_i, n_j. Then*

$$f_j \not R\, X^i(n_i) \Rightarrow \vec{a}_i \; R \; X^i(n_i) \text{ and } j < i.$$

(v) *Suppose $y_i^{n_i} \; R \; y_j^{n_j}$ and $y_i^{n_i} \neq y_j^{n_j}$ for some $i \leq j, n_i, n_j$. Then $i \neq j$ and we have*

$$f_j \; R \; y_j^{n_j}.$$

We explain what each of the items (i) through (v) mean. (i) says that for each class $[X^i(n_i)]_R$, with $X^i(n_i)$ in the $\mathcal{R}_i$-column, if it contains an element x not initially in the class, then x must necessarily belong to a column of lower priority, unless $\mathcal{R}_i$ itself has already acted to collapse $X^i(n_i)$ to $\vec{a}_i$. (ii) expresses a similar fact for a $\mathcal{Q}_i$-requirement. It says that if the $\mathcal{Q}_i$-follower $y_i^{n_i}$ is related to a number x which is new, then this number must belong to a column of lower priority, unless $\mathcal{Q}_i$ has already acted to collapse f_i with $y_i^{n_i}$.

(iii), (iv) and (v) give specific details about the kind of elements which may be allowed to be collapsed to a given column. (iii) says that if some $X^i(n_i)$, in the $\mathcal{R}_i$-column, is collapsed with some $X^j(n_j)$ in the $\mathcal{R}_j$-column, where $i \neq j$, then it must be that either $\mathcal{R}_i$ or $\mathcal{R}_j$ has previously acted to collapse $X^i(n_i)$ with $\vec{a}_i$, or $X^i(n_j)$ with $\vec{a}_j$. Hence (iii) says that it is impossible for two $\mathcal{R}$-columns working for different requirements to be collapsed unless one of the two $\mathcal{R}$-requirements has already performed coding into the column. (iv) investigates when elements of an $\mathcal{R}_i$-column can collapse with elements in a column containing a $\mathcal{Q}_j$-follower. It says that the only way this can happen is if either the $\mathcal{Q}_j$-requirement has acted to collapse $y_j^{n_j}$ with f_j (and hence will never act again in the future), or if $\mathcal{R}_i$ has already collapsed $X^i(n_i)$ with $\vec{a}_i$. Lastly (v) asserts that the only way for a $\mathcal{Q}_i$-follower to be collapsed with a $\mathcal{Q}_j$-follower for $i \neq j$, is when the lower-priority one of the two acts to cause the collapse.

This lemma then enables us to later verify that the $\mathcal{R}$-requirements are met. To see this, consider two $\mathcal{R}$-columns which the $\mathcal{R}$-requirements want to keep distinct. To argue that these two columns are never unintentionally collapsed during the construction, note that parts (iii), (iv) and (v) of the lemma say that the foreign elements introduced into these columns during the construction must be targeted for other requirements, say $\mathcal{R}'$ or $\mathcal{Q}'$, that have already acted for these columns. Hence neither $\mathcal{R}'$ nor $\mathcal{Q}'$ will ever again do anything directly with these columns.

Proof of Lemma 3.14. At each stage s of the construction we take finitely many actions. We proceed by induction on this sequence of actions. At stage $s = 0$ before any action is taken, every equivalence class starts off as a singleton, so (i)-(v) are clearly true. Suppose (i)-(v) holds at a certain point at stage s. We consider the next action and argue that (i)-(v) still holds after this action. We consider the different cases.

Suppose we are collapsing f_i and y_i^0 in Step 1 (henceforth, while analyzing Step 1, this will be known as the "action"). Since $f_i \not R\, y_i^0$ holds before this action, this means (by induction hypothesis on (iv)) that for every j, n_j, if $X^j(n_j)\ R\ y_i^0$ then $\vec{a}_j\ R\ y_i^0$, and $i < j$. Let us now verify that each of (i)-(v) holds after this next action.

(iii) Fix $X^j(n_j)\ R\ X^k(n_k)$ where $j \neq k$. If this was true before the action then we apply the induction hypothesis. Let us assume otherwise that $X^j(n_j)$ and $X^k(n_k)$ are collapsed by the action. Hence we must have (without loss of generality) that both $X^j(n_j)\ R\ y_i^0$ and $X^k(n_k)\ R\ f_i$ hold before the action. By induction hypothesis (iv) on $X^j(n_j)\ R\ y_i^0$, we conclude that $\vec{a}_j\ R\ X^j(n_j)$. Hence (iii) holds after the action.

(iv) In a similar way we verify that (iv) holds after this action. Suppose that $X^j(n_j)\ R\ y_k^{n_k}$. We assume that these two elements are collapsed by the action, hence we either have $X^j(n_j)\ R\ f_i$ and $y_k^{n_k}\ R\ y_i^0$, or the symmetric case $X^j(n_j)\ R\ y_i^0$ and $y_k^{n_k}\ R\ f_i$ holds. Observe that $k \leq i$ because otherwise the action will cause $\mathcal{Q}_k$ to be initialized.

In the first case we may assume that $y_k^{n_k} \neq y_i^0$ because otherwise $k = i$ and we are immediately done. Hence we can apply the induction hypothesis (v) on $y_k^{n_k}\ R\ y_i^0$ to conclude that $f_i\ R\ y_i^0$ before the action, but this is impossible. Let us assume now that the latter symmetric case holds, i.e., $X^j(n_j)\ R\ y_i^0$ and $y_k^{n_k}\ R\ f_i$. We apply the induction hypothesis (iv) on $X^j(n_j)\ R\ y_i^0$ to conclude that $\vec{a}_j\ R\ X^j(n_j)$ and $j > i$. We already remarked above that $k \leq i$ must be true. Hence $k < j$ and (iv) is verified.

(v) Suppose that $y_j^{n_j}\ R\ y_k^{n_k}$ and these two numbers are different. Without loss of generality assume that $y_j^{n_j}\ R\ f_i$ and $y_k^{n_k}\ R\ y_i^0$. Since this action causes all lower-priority requirements to be initialized, we must have $j, k \leq i$. Clearly $j \neq i$ because otherwise the construction would not have collapsed f_i and y_i^0. If $k = i$ then we are immediately done for (v). Hence we assume that $k < i$, and applying the inductive hypothesis regarding $y_k^{n_k}\ R\ y_i^0$, we get that $f_i\ R\ y_i^0$ before the action, a contradiction.

(i) We fix j, n_j such that $\vec{a}_j \not R\, X^j(n_j)$. If $X^j(n_j)$ is related to neither f_i nor y_i^0 before the action, then once again we have that (i) holds by applying the induction hypothesis. $X^j(n_j)\ R\ y_i^0$ before the action is not possible by the induction hypothesis (iv). Hence it must be that $X^j(n_j)\ R\ f_i$ before the action. If $\mathcal{R}_j$ is of lower priority than $\mathcal{Q}_i$ then $\mathcal{R}_j$ is initialized after this action and so (i) is trivially true (since each fresh equivalence class is a singleton). Otherwise, $\mathcal{R}_j$ is of higher priority, which means that $X^j(2) \leq y_i^0$, so by induction hypothesis (i)-(ii), we obtain (i).

(ii) We proceed similarly as in (i).

We now consider the next action in Step 2. Fix $i < s$ and consider the action of coding R_i into X^i (henceforth "action" refers to this). There are two cases.

Case 1: We have that $X^i(n)$ and $X^i(n')$ are collapsed, where $\vec{a}_i \not R\, X^i(n)$ and $\vec{a}_i \not R\, X^i(n')$ (before the action). We run through each case.

(iii) Consider $X^j(n_j)\ R\ X^i(n)$ and $X^k(n_k)\ R\ X^i(n')$, where $j \neq k$. Hence i is not equal to one of j or k. Apply the induction hypothesis (iii) on the appropriate pair.

(iv) Apply induction hypothesis (iv).

(v) Consider $y_j^{n_j}\ R\ X^i(n)$ and $y_k^{n_k}\ R\ X^i(n')$. By induction hypothesis (iv), we have $f_j\ R\ y_j^{n_j}$ and $f_k\ R\ y_k^{n_k}$, and clearly $j \neq k$.

(i) Consider j, n_j such that $\vec{a}_j \not R\ X^j(n_j)$, and $X^j(n_j)\ R\ X^i(n)$. The case $j \neq i$ is impossible by induction hypothesis (iii). So assume $j = i$. By induction hypothesis, both classes $[X^i(n)]_R$ and $[X^i(n')]_R$ contain only numbers no smaller than $X^i(2)$, so we are again done.

(ii) Trivially true.

Case 2: We have $\vec{a}_i\ R\ X^i(n)$ and $\vec{a}_i \not R\ X^i(n')$. Again we consider each case separately.

(iii) A straightforward application of the induction hypothesis (iii).

(iv) Consider $y_j^{n_j}\ R\ X^k(n_k)$. If $y_j^{n_j}\ R\ X^i(n')$ then $f_j\ R\ X^i(n')$ by induction hypothesis (iv), which means (iv) must be true. Hence we may assume that $y_j^{n_j}\ R\ X^i(n)$, and that $f_j \not R\ y_j^{n_j}$. By the induction hypothesis (iv) on $y_j^{n_j}\ R\ X^i(n)$, we must have $j < i$. We have $X^k(n_k)\ R\ X^i(n')$. If $k = i$ then we are done, so assume $k \neq i$. Hence by induction hypothesis (iii), we have $\vec{a}_k\ R\ X^i(n')$. If $k < i$ then by construction $X^i(2) >$ each element in $\vec{a}_k$, contradicting the induction hypothesis (i). Hence we must have $k > i > j$, so we have (iv).

(v) Fix $y_j^{n_j}\ R\ X^i(n)$ and $y_k^{n_k}\ R\ X^i(n')$ and $y_j^{n_j} \neq y_k^{n_k}$. We have, by the induction hypothesis (iv), $f_k\ R\ X^i(n')$. If $f_j\ R\ X^i(n)$ then $j \neq k$ and (v) holds. So suppose that $f_j \not R\ X^i(n)$. By the induction hypothesis (i)-(ii), we get that $\mathcal{Q}_j < \mathcal{R}_i < \mathcal{Q}_k$. To wit, by induction hypothesis on (i) and the fact that $\vec{a}_i \not R\ X^i(n')$ and $y_k^{n_k}\ R\ X^i(n')$, we conclude that $y_k^{n_k} \geq X^i(2)$. Therefore $\mathcal{Q}_k$ is of lower priority than $\mathcal{R}_i$ because otherwise, $\mathcal{R}_i$ would pick $X^i(2)$ to be larger than $y_k^{n_k}$. To conclude that $\mathcal{Q}_j$ is of higher priority than $\mathcal{R}_i$, we apply the induction hypothesis (ii) to conclude that some element in $\vec{a}_i$ is $\geq y_j^{n_j}$, whereas, if $\mathcal{Q}_j$ were of lower priority, then it would pick $y_j^{n_j}$ larger than $X^i(2)$. Hence we have (v).

(i) Fix $X^j(n_j)$ where $\vec{a}_j \not R\ X^j(n_j)$. If $j = i$ then the claim is trivial, so assume $j \neq i$. By induction hypothesis (iii), we conclude that $X^j(n_j) \not R\ X^i(n')$, and hence $X^j(n_j)\ R\ X^i(n)$. By the induction hypothesis (i) applied to $X^j(n_j)$, we have $j < i$. Now by the induction hypothesis (i), this time applied to $X^i(n')$, and the fact that $X^i(2) > X^j(2)$, we obtain (i).

(ii) Fix $y_j^{n_j}\ R\ X^i(n)$ where $f_j \not R\ y_j^{n_j}$ (again by induction hypothesis (iv), $y_j^{n_j}\ R\ X^i(n')$ is impossible). Then by the induction hypothesis (ii), $\mathcal{Q}_j$ is of higher priority than $\mathcal{R}_i$, which means that $X^i(2) > y_j^{n_j}$. Thus by the induction hypothesis (i) applied to $X^i(n')$, we have (ii). □

We now argue that each $\mathcal{Q}_e$ is met. Fix a stage after which $\mathcal{Q}_e$ is never initialized, and let y_e^0, y_e^1 be the final $\mathcal{Q}_e$ followers. By Lemma 3.14(v), $y_e^0 \not\mathrel{R}\, y_e^1$. Thus if $\varphi_e(y_e^0, y_e^1, z_e^0, z_e^1) \downarrow f_e$ then $y_e^0 y_e^1\ R\ f_e$, hence φ_e cannot be the function witnessing that the ceer is u.e.i.

Now consider $\mathcal{R}_e$ and a stage after which it is never initialized. Let X^e be the final version of the modified e^{th} column. We claim that for i or $j \geq 2$, $i\ R_e\ j$ if and only if $X^e(i)\ R\ X^e(j)$. The left to right implication is explicitly ensured by the construction. Suppose that $X^e(i)$ is collapsed with $X^e(j)$ at some stage s in the construction, by some action which is not the coding of R_e. There are again two cases.

Case 1: The collapse is due to coding of R_k for $k \neq e$. We may assume that $X^k(l)\ R\ X^e(i)$ and $X^k(l')\ R\ X^e(j)$. We do not worry about the case when $\vec{a}_e\ R\ X^e(i)$ and $\vec{a}_e\ R\ X^e(j)$, since we would make $\mathcal{R}_e$ inactive after this action. Assume that $\vec{a}_e \not\mathrel{R}\, X^e(i)$ and $\vec{a}_e \not\mathrel{R}\, X^e(j)$. By Lemma 3.14(iii) we have $\vec{a}_k\ R\ X^k(l)$ and $\vec{a}_k\ R\ X^k(l')$, but by construction we would not have collapsed $X^k(l)$ and $X^k(l')$. Now assume that $\vec{a}_e \not\mathrel{R}\, X^e(i)$ and $\vec{a}_e\ R\ X^e(j)$. By Lemma 3.14(i) and (iii) we get that $e < k$. Now since $\vec{a}_k \not\mathrel{R}\, X^k(l')$ by Lemma 3.14(i) and (iii) again we get that $k < e$, a contradiction.

Case 2: The collapse is due to action in Step 1. Assume we collapsed $y_k^0\ R\ X^e(i)$ with $f_k\ R\ X^e(j)$. Since $f_k \not\mathrel{R}\, y_k^0$ before this action, by Lemma 3.14(iv) we have $\vec{a}_e\ R\ X^e(i)$ and $k < e$, hence $\mathcal{R}_e$ will get initialized, a contradiction. □

Question 3.15. *Do the u.f.p. ceers coincide with the weakly u.f.p. ceers?*

Corollary 3.16 below subsumes all universality results known in the literature, and is a natural companion of classical results, including: Every creative set is m-complete (Myhill [26]); every pair of effectively inseparable sets is m-complete (Smullyan [31]); all creative sequences are m-complete (Cleave [7]).

Corollary 3.16. *Every u.e.i. ceer is universal.*

Proof. Immediate by Theorem 3.7, as every strongly u.m.c. (or even u.m.c.) ceer is clearly universal: If R is a u.m.c. ceer, and S is any ceer with two distinct equivalence classes, then start off with an assignment $(a_0', a_1') \mapsto (a_0, a_1)$ with $a_0^{'} \not\mathrel{S}\, a_1^{'}$ and $a_0 \not\mathrel{R}\, a_1$, and extend it to a full reduction. □

Corollary 3.17. *A ceer R is universal if and only if there exists a u.e.i. ceer S with $S \leq R$.*

Proof. If R is universal and S is u.e.i., then trivially $S \leq R$. Conversely, if S is u.e.i. and $S \leq R$, then R is universal, since so is S, by Corollary 3.16. □

Remark 3.18. Of course, if R is a universal ceer, then for every ceer S, we have that $R \oplus S$ is also universal. So there are universal ceers that are not u.e.i., in fact not even e.i.

The following theorem shows that uniformity is essential in proving that u.e.i. ceers are universal:

Theorem 3.19. *There exists an e.i. ceer that is not universal.*

Proof. To show the result, we build computable approximations as in Lemma 1.1, to ceers R and S, such that R is e.i., and $S \not\leq R$. At any stage, a number is *new* if it is bigger than any number so far used in the construction.

The construction of R and S will satisfy the following requirements, for all numbers a, b, k:

$P_{a,b}$: $[a]_R \cap [b]_R = \emptyset \Rightarrow f_{a,b}$ is a productive function for the pair $([a]_R, [b]_R)$,

N_k : φ_k does not witness $S \leq R$,

where $f_{a,b}$ is a total computable function we build. In fact, $P_{a,b}$ should be written as $P_{\{a,b\}}$ with $a < b$ (thus $P_{a,b} = P_{b,a}$: Of course $f_{b,a}$ can be easily obtained from $f_{a,b}$), and one should think of the P-requirements as linearly ordered according to the canonical index of $\{a, b\}$. In order to achieve that φ_k does not reduce S to R, strategy N_k will use four witnesses $a_0(k), a_1(k), b_0(k), b_1(k)$.

At any stage we say that we *initialize* $P_{a,b}$ if, at this stage, we set $f_{a,b} = \emptyset$; and we *initialize* N_k if we set the witnesses $a_0(k), a_1(k), b_0(k), b_1(k)$ to be undefined.

We must also make sure that R is not trivial.

Strategy for $P_{a,b}$. We say that $P_{a,b}$ becomes *inactive at stage* $s+1$ (and stays inactive ever after) if either a is not the least element of $[a]_{R^{s+1}}$ or b is not the least element of $[b]_{R^{s+1}}$. If $P_{a,b}$ first becomes inactive at $s+1$, then it initializes all strategies of lower priority.

If $P_{a,b}$ is not inactive at stage $s+1$, then we extend the definition of $f_{a,b}$ to the next (by code) pair (u, v) and correct the already defined values of $f_{a,b}$ as follows:

(1) Define $f_{a,b}(u, v) = m$, where m is new;
(2) If $f_{a,b}(u', v') = m'$ has been already defined at previous stages, then
   (a) R-collapse m' and b, if $m' \in W_{u'}$ at the current stage;
   (b) R-collapse m' and a, if $m' \in W_{v'}$ at the current stage.

Outcomes of strategy $P_{a,b}$. Notice that in (2a) we make $[b]_R \not\subseteq W_{v'}$ if $W_{u'} \cap W_{v'} = \emptyset$; similarly, under the same assumptions, we make $[a]_R \not\subseteq W_{u'}$ in (2b). Thus if $[a]_R \subseteq W_u$, $[b]_R \subseteq W_v$, and $W_u \cap W_v = \emptyset$ then $f_{a,b}(u, v) \notin W_u \cup W_v$. So if $P_{a,b}$ requires attention infinitely often, eventually without being initialized, then $f_{a,b}$ is a total productive function for the pair $([a]_R, [b]_R)$.

Strategy for N_k. The strategy aims at making S not reducible to R via φ_k:

(1) Appoint numbers a_0, a_1, b_0, b_1, which are new (hence, for every $x, y \in \{a_0, a_1, b_0, b_1\}$ such that $x \neq y$, we have $x \not\mathrel{S} y$, at the current stage).
(2) Wait for $\varphi_k(a_0)\downarrow$ and $\varphi_k(b_0)\downarrow$:
   (a) If already $\varphi_k(a_0) \mathrel{R} \varphi_k(b_0)$, then do nothing.
   (b) Otherwise, S-collapse a_0 and b_0, and initialize lower-priority strategies; and
   (c) if later $\varphi_k(a_0) \mathrel{R} \varphi_k(b_0)$ (notice that, by the initialization undertaken in the previous item, this can happen only due to the action of higher-priority strategies), then repeat the previous steps with a_1, b_1 in place of a_0, b_0 respectively; more specifically, go to stage (2) with a_1, b_1 in place of a_0, b_0 respectively.

(3) After completing (2c) for a_1, b_1, if already

$$\varphi_k(a_0) \mathrel{R} \varphi_k(b_0) \mathrel{R} \varphi_k(a_1) \mathrel{R} \varphi_k(b_1),$$

then do nothing;

(4) otherwise, S-collapse a_0, b_0, a_1, b_1, and initialize lower-priority requirements.

We say that N_k *requires attention* at stage $s+1$, if N_k is ready to act according to (1), or (2b) for a_0, b_0, or (2b) for a_1, b_1, or (4).

Outcomes of strategy N_k. The strategy has the following outcomes:

(1) If the strategy stops at (2) before reaching (2a), either for the pair a_0, b_0 or for the pair a_1, b_1, then φ_k is not total, and therefore N_k is satisfied.
(2) If (2a) holds for the pair a_0, b_0 then $a_0 \not\mathrel{S} b_0$ and $\varphi_k(a_0) \mathrel{R} \varphi_k(b_0)$; similarly, if (2a) holds for the pair a_1, b_1 then $a_1 \not\mathrel{S} b_1$ and $\varphi_k(a_1) \mathrel{R} \varphi_k(b_1)$.
(3) If we wait forever at (2c) for the pair a_0, b_0 then $a_0 \mathrel{S} b_0$ and $\varphi_k(a_0) \not\mathrel{R} \varphi_k(b_0)$; similarly, if we wait forever at (2c) for the pair a_1, b_1 then $a_1 \mathrel{S} b_1$ and $\varphi_k(a_1) \not\mathrel{R} \varphi_k(b_1)$.
(4) Otherwise, at some point, the strategy yields

$$a_i \mathrel{S} b_i \text{ and } \varphi_k(a_i) \mathrel{R} \varphi_k(b_i),$$

for both $i = 0, 1$. When this happens,
   (a) if already $\varphi_k(b_0) \mathrel{R} \varphi_k(a_1)$, then we keep $b_0 \not\mathrel{S} a_1$;
   (b) if $\varphi_k(b_0) \not\mathrel{R} \varphi_k(a_1)$, then our action in (4) makes $b_0 \mathrel{S} a_1$, and, by initialization, keeps $\varphi_k(b_0) \not\mathrel{R} \varphi_k(a_1)$.

The outcomes considered so far are all winning outcomes for N_k. We must exclude the possibility that we end up with

$$\varphi_k(a_0) \mathrel{R} \varphi_k(b_0) \mathrel{R} \varphi_k(a_1) \mathrel{R} \varphi_k(b_1)$$

and we have already S-collapsed a_0, b_0, a_1, b_1. Notice that when we defined $a_0 \mathrel{S} b_0$ we had $\varphi_k(a_0) \not\mathrel{R} \varphi_k(b_0)$ by (2a). The R-collapse of $\varphi_k(a_0)$ and $\varphi_k(b_0)$ to, say, a number a (which is the least in its equivalence class) must be the effect of later actions of higher-priority strategies, one of which is of the form $P_{a,b}$: After convergence of $\varphi_k(a_0)$ and $\varphi_k(b_0)$, the lower-priority P-requirements are initialized, and thus they cannot move $\varphi_k(a_0)$ or $\varphi_k(b_0)$ to new equivalence classes, since they can only move their markers m, but these by initialization are chosen to be different from all elements in the equivalence classes of $\varphi_k(a_0)$ and $\varphi_k(b_0)$. Similarly, when we defined $a_1 \mathrel{S} b_1$, we had $\varphi_k(a_1) \not\mathrel{R} \varphi_k(b_1)$. The R-collapse of $\varphi_k(a_1)$ and $\varphi_k(b_1)$ to, say, c (which is the least in its equivalence class) must be the effect of later actions of higher-priority strategies, one being of the form $P_{c,d}$. When we S-collapsed a_0, b_0, a_1, b_1, we had $\varphi_k(b_0) \not\mathrel{R} \varphi_k(a_1)$, hence $a \not\mathrel{R} c$. When later we R-collapse a and c, either a or c stops being the least representative in its equivalence class, and so either $P_{a,b}$ or $P_{c,d}$ becomes inactive, and it initializes N_k. We will argue, however, that eventually N_k is not initialized anymore, so there is a final choice of the witnesses which allows for N_k only winning outcomes.

Construction. The construction at stage s proceeds in substages $t \leq s$. At stage 0, all strategies are initialized. At a substage $t \leq s$ of a stage $s > 0$, if $t = s$ then we end the stage. If $t < s$, then we attack the requirement Q with priority rank t. If Q is a P-requirement, say $Q = P_{a,b}$, that was not inactive at the previous stage, but is now inactive, then we end the stage; otherwise, if Q is not inactive, then we act as described above (in the section "Strategy for $P_{a,b}$"). If Q is an N-requirement, say $Q = N_k$, that requires attention then we act as described above (in the section "Strategy for N_k"), and we end the stage. In all other cases for $t < s$, after completing substage t we move on to substage $t + 1$.

After completing stage s, with t, say, the last substage before completing the stage, then we initialize all requirements having lower priority than Q_t.

Let $S = \bigcup_s S^s$, $R = \bigcup_s R^s$.

Verification. The verification is based on the following

Lemma 3.20. *Each requirement Q initializes lower-priority strategies only finitely often, and if $Q = N_k$, for some k, then Q requires attention finitely often.*

Proof. This follows by a simple inductive argument. Suppose that the claim is true of every requirement Q', with $Q' < Q$. If $Q = N_k$, after all Q', with $Q' < N_k$, stop initializing, we have that N_k cannot be further initialized and requires attention only finitely often, since all outcomes are finitary, and thus it also initializes only finitely often. Similarly, if $Q = P_{a,b}$: After its last initialization, $P_{a,b}$ may initialize lower-priority strategies at most once, upon becoming inactive. □

Lemma 3.21. *Let a, b be such that $P_{a,b}$ is the highest priority P-requirement. Then $[a]_R \cap [b]_R = \emptyset$. Thus R is nontrivial.*

Proof. If a P-requirement $P_{c,d}$ appoints $f_{c,d}(u, v) = m$, then we may assume $m \neq a, b$, and only $P_{c,d}$ can enumerate m at some point in either $[a]_R$ or $[b]_R$, but not in both. □

Lemma 3.22. *Each requirement is satisfied, or eventually inactive.*

Proof. Let $P_{a,b}$ be given, with $a < b$, and $P_{a,b}$ not eventually inactive. By Lemma 3.20, there is a least stage after which $P_{a,b}$ is not initialized any more. Then after this stage, we construct $f_{a,b}$ witnessing that $([a]_R, [b]_R)$ is an e.i. pair.

Let us now consider the case of an N-requirement N_k, and let s_0 be a stage after which N_k is never again initialized, so no higher-priority N-requirement requires attention after s_0, nor does any higher-priority P-requirement become inactive after s_0. After its last initialization, N_k appoints four permanent witnesses $a_0(k)$, $b_0(k)$, $a_1(k)$, $b_1(k)$. For simplicity, for $i = 0, 1$, write $a_i = a_i(k)$ and $b_i = b_i(k)$. We may suppose that for every $i = 0, 1$, $\varphi_k(a_i)$ and $\varphi_k(b_i)$ converge, otherwise N_k is trivially satisfied. Moreover we may suppose that action taken by N_k makes $a_0 \; S \; b_0$ and $a_1 \; S \; b_1$; otherwise, again N_k is satisfied. We must exclude the possibility that the numbers $\varphi_k(a_0), \varphi_k(b_0), \varphi_k(a_1), \varphi_k(b_1)$ all R-collapse, and the numbers a_0, b_0, a_1, b_1 all S-collapse. But, as explained in the informal description of the outcomes of N_k, this possibility would require some $P < N_k$ to become inactive at some stage after s_0, thus providing one more initialization of N_k, which is impossible by the choice of s_0. □

This completes the proof. □

4. Characterizing universal ceers through a jump operator

The following definition is due to Gao and Gerdes [20]; the defined operation is called the *halting jump operation.*

Definition 4.1. Given a ceer R, define

$$x \; R' \; y \Leftrightarrow [x = y \text{ or } \varphi_x(x){\downarrow} R \; \varphi_y(y){\downarrow}] .$$

Lemma 4.2 (Gao and Gerdes [20])**.** *The following properties hold:*

- $R \leq R'$;
- $R \leq S \Leftrightarrow R' \leq S'$;
- *If* R *is not universal then* R' *is not universal.*

One can thus introduce a well-defined operation on $\mathcal{P}$, by

$$(\deg(R))' = \deg(R').$$

Notice that $(\mathrm{Id}_1)' = R_K$, that is the equivalence relation having the halting set K as its unique nontrivial equivalence class, and $(\mathrm{Id})'$ is the ceer yielding the partition

$$\{K_i : i \in \omega\} \cup \{\{x\} : x \notin K\},$$

where $K_i = \{x : \varphi_x(x){\downarrow}= i\}$.

The following theorem answers Problem 10.2 of [20].

Theorem 4.3. *For every ceer* E*, if* $E' \leq E$ *then* E *is universal.*

Proof. Assume that h is a computable function that reduces E' to E. Let R be a ceer, with computable approximations $\{R^s : s \in \omega\}$; similarly, we will work with computable approximations $\{E^s : s \in \omega\}$ to E, as in Lemma 1.1. We aim to show that $R \leq E$.

We first outline the idea of the proof through a particular example. We use an infinite computable sequence of indices $e_0, e_1, \ldots$, which we control by the Recursion Theorem. Eventually we define $g(i) = h(e_i)$, and show

$$i \; R \; j \Leftrightarrow e_i \; E' \; e_j (\Leftrightarrow h(e_i) \; E \; h(e_j))$$

i.e., g reduces R to E. Our choice of these indices will make us able to E'-collapse any pair of them as needed. Suppose for instance that we want to make $e_0 \; E' \; e_1$ because we see at some point that $0 \; R \; 1$. The basic module for this is the following:

(1) Keep $\varphi_{e_0}(e_0)$ and $\varphi_{e_1}(e_1)$ undefined until we see $0 \; R \; 1$.
(2) Define $\varphi_{e_0}(e_0) = \varphi_{e_1}(e_1) = h(e')$ for another suitably chosen fixed point e' (while keeping $\varphi_{e'}(e'){\uparrow}$).

Suppose that even later we want to E'-collapse e_1 and e_2:

(1) Keep $\varphi_{e'}(e')$ and $\varphi_{e_2}(e_2)$ undefined, until $1 \; R \; 2$.
(2) Define $\varphi_{e_2}(e_2) = h(e'')$ and $\varphi_{e'}(e') = \varphi_{e''}(e'') = h(e''')$ (while keeping $\varphi_{e'''}(e'''){\uparrow}$), where e'' and e''' are further suitably chosen fixed points.

Notice that

$$\begin{aligned}\varphi_{e'}(e')\!\downarrow = \varphi_{e''}(e'')\!\downarrow &\Rightarrow e' \; E' \; e'' \\ &\Rightarrow h(e') \; E \; h(e'') \\ &\Rightarrow e_1 \; E' \; e_2.\end{aligned}$$

Care must be taken (by carefully controlling convergence of the various computations $\varphi_e(e)$), to collapse only what we *need* to collapse. In particular, if we see that E is threatening to E-collapse values, say, $h(e_i)$ and $h(e_j)$, without having $i \; R \; j$, then we threaten in our turn to stop the construction leaving certain computations divergent (exploiting the fact that if $u \neq v$ and $\varphi_u(u)$ and $\varphi_v(v)$ do not converge, then $u \not\mathrel{E'} v$, and thus $h(u) \not\mathrel{E} h(v)$), therefore forcing E to remove its threat if it wants to avoid a contradiction.

If D is a finite set, and n is a number, then $\langle D, n\rangle$ denotes the code $\langle u, n\rangle$ where u is the canonical index of D. A pair $\alpha = \langle D, n\rangle$ will be called a *node*: We sometimes denote the components of a node α by D_α and n_α. Our formal implementation of the above idea uses the Case Functional Recursion Theorem as a tool to find infinitely many synchronized fixed points. Thus, we assume that we are working with a computable sequence of indices $\{e_\alpha : \alpha \text{ node}\}$, which we control by the Case Functional Recursion Theorem.

It might be instructive to see how the two-step example above is formally implemented.

(1) Keep $\varphi_{e_{\langle\{0\},0\rangle}}(e_{\langle\{0\},0\rangle})$ and $\varphi_{e_{\langle\{1\},0\rangle}}(e_{\langle\{1\},0\rangle})$ undefined, until we see $0 \; R \; 1$.
(2) Define

$$\varphi_{e_{\langle\{0\},0\rangle}}(e_{\langle\{0\},0\rangle}) = \varphi_{e_{\langle\{1\},0\rangle}}(e_{\langle\{1\},0\rangle}) = h(e_{\langle\{0,1\},1\rangle}),$$

still keeping $\varphi_{e_{\langle\{0,1\},1\rangle}}((e_{\langle\{0,1\},1\rangle})$ undefined (notice that $\{0\}$ and $\{1\}$ merge into $\{0,1\}$) so that, in the two-step example above, we take $e_0 = e_{\langle\{0\},0\rangle}$, $e_1 = e_{\langle\{1\},0\rangle}$, and $e' = e_{\langle\{0,1\},1\rangle}$.

Suppose that even later we want to E'-collapse e_1 and $e_2 = e_{\langle\{2\},0\rangle}$:

(1) Keep $\varphi_{e_{\langle\{0,1\},1\rangle}}(e_{\langle\{0,1\},1\rangle})$ and $\varphi_{e_{\langle\{2\},0\rangle}}(e_{\langle\{2\},0\rangle})$ undefined, until $1 \; R \; 2$.
(2) Define $\varphi_{e_{\langle\{2\},0\rangle}}(e_{\langle\{2\},0\rangle}) = h(e_{\langle\{2\},1\rangle})$, and set

$$\varphi_{e_{\langle\{2\},1\rangle}}(e_{\langle\{2\},1\rangle}) = \varphi_{e_{\langle\{0,1\},1\rangle}}(e_{\langle\{0,1\},1\rangle}) = h(e_{\{0,1,2\},2\rangle}).$$

Thus, taking $e'' = e_{\langle\{2\},1\rangle}$, and $e''' = e_{\langle\{0,1,2\},2\rangle}$ we have that $\varphi_{e_2}(e_2) = h(e'')$ and $\varphi_{e'}(e') = \varphi_{e''}(e'') = h(e''')$ (still keeping $\varphi_{e'''}(e''')\!\uparrow$). (Notice, since we want to merge $\{2\}$ and $\{0,1\}$ into $\{0,1,2\}$, and since the node $\alpha = \langle\{0,1\},1\rangle$ has level 1, i.e., $n_\alpha = 1$, we first transform $\langle\{2\},0\rangle$ into a node $\langle\{2\},1\rangle$ with level 1: This transformation procedure will be called the synchronization procedure in the formal construction given below).

We see that the desired numbers e_i are taken to be $e_i = e_{\langle\{i\},0\rangle}$.

We say that a node β is a *parent* of a node α, if

- $n_\alpha = n_\beta + 1$; and
- $\varphi_{e_\beta}(e_\beta)\!\downarrow = h(e_\alpha)$.

The construction will make sure that every node has at most two parents. A node α has only one parent β if α is the result of a definition due to the synchronization procedure, described below, i.e., $\alpha = \langle D_\beta, n_\beta + 1\rangle$ and $\varphi_{e_\beta}(e_\beta) = h(e_\alpha)$.

Given a node α, let T_α be the finite tree, defined as the smallest set of nodes such that:

- $\alpha \in T_\alpha$;
- if $\beta \in T_\alpha$ and γ is a parent of β, then $\gamma \in T_\alpha$.

Finiteness of T_α follows from the fact that if γ is a parent of β, then $n_\gamma < n_\beta$. We say that a node α is *realized*, if $n_\alpha = 0$ or $T_\alpha \neq \{\alpha\}$.

The above notions (a node β is a parent of a node α; the tree T_α; and α is realized) can be approximated at each stage s in the obvious way, by approximating at stage s the relevant computations $\varphi_e(e)$. In fact, if α is realized at s, then $T_{\alpha,s} = T_\alpha$, as can be easily seen from the construction. The guiding idea is that if α is realized at s, and $\varphi_{e_\alpha}(e_\alpha)$ is still undefined, then D_α is a block of R^s; if at some later stage $t > s$, R collapses D_α with another block D_β, relative to a similarly realized β, with $\varphi_{e_\beta}(e_\beta)$ still undefined, and $n = n_\alpha = n_\beta$, then we will define

$$\varphi_{e_\alpha}(e_\alpha) = \varphi_{e_\beta}(e_\beta) = h(e_{\langle D_\alpha \cup D_\beta, n+1\rangle}).$$

(We say that these convergent computations *code R into E*.)

Lemma 4.4. *Let α be a realized node, with $n_\alpha = n$. For every $i \leq n$, for every $\beta, \gamma \in T_\alpha$, if $n_\beta = n_\gamma = i$ then $h(e_\beta)\ E\ h(e_\gamma)$.*

Proof. We may assume $n > 0$, otherwise the claim is trivial. We will prove the claim by reverse induction. Assume $i = n$: The only node $\beta \in T_\alpha$ with $n_\beta = n$ is α. Thus the claim trivially holds for $i = n$.

Suppose that the claim is true of i, with $0 < i$, and let us show it for $i - 1$: For every node γ with $n_\gamma = i - 1$, there is a node β with $n_\beta = i$ such that $\varphi_{e_\gamma}(e_\gamma) = h(e_\beta)$. But by the inductive assumption, all the nodes β with $n_\beta = i$ are such that the corresponding values $h(e_\beta)$ are all E-equivalent, hence if γ, δ are nodes such that $n_\gamma = n_\delta = i - 1$, we have that $e_\gamma\ E'\ e_\delta$ and thus $h(e_\gamma)\ E\ h(e_\delta)$. □

Lemma 4.5. *If α and β are distinct realized nodes, with $n_\alpha = n_\beta = n$ such that $\varphi_{e_\alpha}(e_\alpha)$ and $\varphi_{e_\beta}(e_\beta)$ are undefined, then, for every $\gamma \in T_\alpha$, and $\delta \in T_\beta$ such that $n_\gamma = n_\delta$, we have that $e_\gamma \not\mathrel{E'} e_\delta$.*

Proof. By hypothesis we have $\varphi_{e_\alpha}(e_\alpha)\uparrow$ and $\varphi_{e_\beta}(e_\beta)\uparrow$. So the claim is true of $i = n$ since $e_\alpha \not\mathrel{E'} e_\beta$. Suppose now that the claim is true of $0 < i \leq n$, and let $\gamma \in T_\alpha$ and $\delta \in T_\beta$ be such that $n_\gamma = n_\delta = i - 1$. Then there are $\gamma' \in T_\alpha$ and $\delta' \in T_\beta$ such that $n_{\gamma'} = n_{\delta'} = i$, and $\varphi_{e_\gamma}(e_\gamma) = h(e_{\gamma'})$, and $\varphi_{e_\delta}(e_\delta) = h(e_{\delta'})$. By induction, $e_{\gamma'} \not\mathrel{E'} e_{\delta'}$, so $h(e_{\gamma'}) \not\mathrel{E} h(e_{\delta'})$, and thus we may conclude that $e_\gamma \not\mathrel{E'} e_\delta$. □

The synchronization procedure for two nodes α, β at stage $s+1$:

(1) If $n_\alpha = n_\beta$ then do nothing.
(2) If $n_\alpha < n_\beta$, then for every i with $n_\alpha \leq i < n_\beta - 1$, define (at stage $s+1$),

$$\varphi_{e_{\langle D_\alpha, i\rangle}}(e_{\langle D_\alpha, i\rangle}) = h(e_{\langle D_\alpha, i+1\rangle}).$$

The purpose of the synchronization procedure can be described as follows: Suppose that we have two nodes α and β and we want to R-collapse D_α and D_β, by defining $\varphi_{e_\alpha}(e_\alpha)\downarrow = \varphi_{e_\beta}(e_\beta)\downarrow$. But, following the construction which is detailed later, this can be done only if $n_\alpha = n_\beta$: If $n_\alpha < n_\beta$, we keep transforming α into nodes γ, with $D_\gamma = D_\alpha$, but with bigger and bigger n_γ, until we catch up with n_β.

Construction. We are now ready to describe the construction, which basically consists of two main actions:

(1) Waiting for R to catch up with E, when we see at any stage that for some a, b, we have that $h(e_{\langle\{a\},0\rangle})\ E\ h(e_{\langle\{b\},0\rangle})$, but $a \not R\, b$: Then, by the End of Stage procedure, we (momentarily) stop the construction, and wait for $a\ R\ b$. We eventually stop waiting, as checked in the verification.
(2) Coding of R into E via Lemma 4.4, through suitable convergent computations that code R into E (as defined in the remark preceding Lemma 4.4); coding is performed while we are not currently waiting as in the previous item.

Without loss of generality, by Lemma 1.1, we assume that the chosen computable approximation of R at each new stage R-collapses exactly one pair of equivalence classes.

Stage 0. Start off with $\varphi_{e_\alpha}(e_\alpha)$ undefined for every node α.

Stage $s+1$. If we are waiting (as defined in the procedure "End of Stage") at $s+1$ for some pair of nodes α, β, then do nothing and go to next stage. Otherwise, let s^- be the last stage, if any, at the end of which we started to wait for some pair of nodes: If there is no such stage, then let $s^- = s$. For the sake of coding R into E, we now consider all possible R-collapses performed by R on pairs of equivalence classes in the time interval between $s^- + 1$ and $s+1$: Proceed by substages $t = 1, \ldots, s+1-s^-$: At substage t, if α and β are nodes such that R collapses D_α and D_β at $s^- + t$, then synchronize α and β to two new nodes α', β', so after synchronization, we may assume (by replacing α, β with α', β', respectively) $n = n_\alpha = n_\beta$, and define

$$\varphi_{e_\alpha}(e_\alpha) = h(e_{\langle G, n+1\rangle})$$
$$\varphi_{e_\beta}(e_\beta) = h(e_{\langle G, n+1\rangle})$$

where $G = D_\alpha \cup D_\beta$. Notice that G is a new block in the approximation to R at stage $s^- + t$. After completing substage $s+1-s^-$ go to the End of Stage procedure.

End of Stage. Suppose there are a, b such that $h(e_{\langle\{a\},0\rangle})\ E^{s+1}\ h(e_{\langle\{b\},0\rangle})$, but $a \not{R}^{s+1} b$. Pick the least such pair of numbers, and pick nodes α, β, realized at s, such that $a \in D_\alpha$, $b \in D_\beta$, and $\varphi_{e_\alpha}(e_\alpha)$ and $\varphi_{e_\beta}(e_\beta)$ are still undefined at stage $s+1$. (These nodes exist and are unique by Lemma 4.6.) Synchronize α and β at stage $s+1$ to get nodes α', β' of the same level. At any future stage we say that we are *waiting for* α, β, until the first stage at which R collapses D_α and D_β, when we say that we *are not waiting for* α, β. Go to next stage.

If there is no pair of numbers as above, then go to next stage.

Verification. For every n, let

$$g(n) = h(e_{\langle\{n\},0\rangle}).$$

We notice:

Lemma 4.6. *For every a and s, there exists exactly one node α, realized at s, such that $a \in D_\alpha$, which is a block of the equivalence relation R^s, and $\varphi_{e_\alpha}(e_\alpha)$ is undefined at stage s.*

Proof. For $s = 0$, the desired unique node α is $\alpha = \langle\{a\}, 0\rangle$. The full claim follows by an easy induction on s. □

Lemma 4.7. *If α is realized at s, then for every $i \leq n_\alpha$, T_α contains also nodes β with $n_\beta = i$, and contains all nodes $\langle\{a\}, 0\rangle$, for all $a \in D_\alpha$.*

Proof. By the synchronization procedure. The claim that T_α contains all nodes $\langle\{a\}, 0\rangle$, for all $a \in D_\alpha$, follows by induction on n_α. □

Finally, we claim that, for every a, b,

$$a\ R\ b \Leftrightarrow g(a)\ E\ g(b).$$

Assume first that there exist a stage s_0 and a pair of realized α, β which makes us wait at all stages $s \geq s_0$. The reason for this was that we saw at a previous stage that

$$h(e_{\langle\{a\},0\rangle})\ E\ h(e_{\langle\{b\},0\rangle}),$$

for some $a \in D_\alpha$, $b \in D_\beta$, but $a \not{R}\ b$, and D_α and D_β never collapse to the same R-equivalence class at any stage $s \geq s_0$. By construction, for these α and β we have $\varphi_{e_\alpha}(e_\alpha)\uparrow$ and $\varphi_{e_\beta}(e_\beta)\uparrow$. So by Lemma 4.5 and Lemma 4.7 we would conclude

$$h(e_{\langle\{a\},0\rangle}) \not{E}\ h(e_{\langle\{b\},0\rangle}),$$

a contradiction. So, there is no permanent wait, and thus there is no pair a, b such that $g(a)\ E\ g(b)$, but $a \not{R}\ b$.

Let us now show the left-to-right implication. Assume that $a\ R\ b$, and let $s+1$ be the least stage at which some pair of equivalence classes containing a and b R-collapse. Then there is a unique pair of realized α, β with $a \in D_\alpha$ and $b \in D_\beta$ such that $\varphi_{e_\alpha}(e_\alpha)$ and $\varphi_{e_\alpha}(e_\alpha)$ are still undefined at s. Since there is no permanent wait in the construction, there will be a later stage at which we process α, β, and thus we define $\varphi_{e_\alpha}(e_\alpha) = \varphi_{e_\beta}(e_\beta) = h(e_\gamma)$, for some γ (or, rather, $\varphi_{e_{\alpha'}} = \varphi_{e_{\beta'}} = h(e_\gamma)$, where α' and β' are the results of synchronizing α and β). Then by Lemma 4.4 applied to γ and to the tree T_γ, and Lemma 4.7, we have for all c, d in $D_\gamma = D_\alpha \cup D_\beta$, thus including a and b, that $h(e_{\langle\{c\},0\rangle}) \mathrel{E} h(e_{\langle\{d\},0\rangle})$, hence $g(a) \mathrel{E} g(b)$. □

5. Index sets

In this section we classify some index sets of collections of ceers which have been considered in the paper.

We use below that for every Σ^0_3-set S there exists a c.e. class $\{X_{\langle i,j\rangle} : i, j \in \omega\}$ (meaning that the set $\{\langle x, y\rangle : x \in X_y\}$ is c.e.) such that

$$i \in S \Rightarrow (\exists j)[X_{\langle i,j\rangle} = \omega],$$
$$i \notin S \Rightarrow (\forall j)[X_{\langle i,j\rangle} \text{ finite}],$$

see [32, Corollary IV.3.7].

The following answers Problem 10.1 of [20]:

Theorem 5.1. *The index set $\{x : R_x$ is universal$\}$ is Σ^0_3-complete.*

Proof. Let $I_{\text{univ}} = \{x : R_x \text{ is universal}\}$. An easy calculation, using the fact that a ceer R is universal if and only if $E \leq R$, for a fixed universal ceer E, shows that $I_{\text{univ}} \in \Sigma^0_3$, namely,

$$x \in I_{\text{univ}} \Leftrightarrow (\exists e)[\varphi_e \text{ is total and } \varphi_e \text{ reduces } E \text{ to } R_x].$$

Next, we show that for every $S \in \Sigma^0_3$, we have $S \leq_m I_{\text{univ}}$. Given S, fix a universal ceer E and a c.e. class $\{X_{\langle i,j\rangle} : i, j \in \omega\}$ as above; uniformly in i, build a ceer R, such that, denoting by $R^{[j]}$ the ceer

$$x \mathrel{R^{[j]}} y \Leftrightarrow \langle j, x\rangle \mathrel{R} \langle j, y\rangle,$$

we have that

$$i \in S \Rightarrow (\exists j)[R^{[j]} = E],$$
$$i \notin S \Rightarrow R \text{ yields a partition into finite sets.}$$

This is enough to prove the claim, since a universal ceer has always (infinitely many) infinite equivalence classes; indeed, if E, T are ceers such that $E \leq T$ via a computable function f, and $[x]_E$ is an undecidable equivalence class, then so is $[h(x)]_T$, as $[x]_E = f^{-1}[[h(x)]_T]$.

Construction. Let $\{E_s\}_{s\in\omega}$ be a computable approximation to E as a c.e. set, with each E_s finite, and consider a computable approximation $\{X_{\langle i,j\rangle,s}\}_{s\in\omega}$ to $\{X_{\langle i,j\rangle}\}_{i,j\in\omega}$ via finite sets: We say that $s+1$ is $\langle i, j\rangle$-*expansionary* if

$$X_{\langle i,j\rangle,s+1} - X_{\langle i,j\rangle,s} \neq \emptyset.$$

Stage by stage we define, uniformly in i, a finite set R^s so that, eventually, $R = \bigcup_s R^s$ is our desired ceer.

Stage 0. Let $R^0 = \emptyset$.

Stage $s+1$. Let j be the least number $\leq s$, if any, such that $s+1$ is $\langle i,j\rangle$-expansionary. Then carry out the following, with the understanding that if there is no such j, then only item (1) applies:

(1) For every $k \neq j$, $k \leq s$, and $x \leq s$, let $\langle\langle k,x\rangle,\langle k,x\rangle\rangle \in R^{s+1}$.
(2) Let $\langle\langle j,x\rangle,\langle j,y\rangle\rangle \in R^{s+1}$ for every $\langle x,y\rangle \in E_s$.

It is straightforward to verify that if $i \notin S$ then every j has only finitely many $\langle i,j\rangle$-expansionary stages, so the equivalence classes of R are finite, hence R is not universal. Otherwise, for the least j such that there are infinitely many $\langle i,j\rangle$-expansionary stages, we have that $R^{[j]} = E$, hence $E \leq R$, i.e., R is universal. □

Theorem 5.2. *The set* $\{x : R_x$ *is u.e.i.*$\}$ *is* Σ^0_3*-complete.*

Proof. Let $I_{\text{u.e.i.}} = \{x : R_x \text{ is u.e.i.}\}$. A simple calculation shows that $I_{\text{u.e.i.}} \in \Sigma^0_3$. We now show that for every $S \in \Sigma^0_3$, we have $S \leq_m I_{\text{u.e.i.}}$. Given S, fix a c.e. class $\{X_{\langle i,j\rangle} : i,j \in \omega\}$ as above; uniformly in i, build a (non-trivial) ceer R, such that

$$i \in S \Rightarrow R \text{ is u.e.i.},$$
$$i \notin S \Rightarrow R \text{ is not u.e.i.}$$

Fixing i, for every j we have the two requirements:

$$\begin{aligned} P_j : &\quad \varphi_j \text{ is not a uniform total productive function for } R,\\ Q_j : &\quad (\exists^\infty \langle i,j\rangle\text{-expansionary stages}) \Rightarrow\\ &\qquad f_j \text{ is a uniform total productive function for } R,\end{aligned}$$

where f_j is a partial computable function (which is total if there exist infinitely many $\langle i,j\rangle$-expansionary stages) we build. We will guarantee also that R is not trivial, as by definition a u.e.i. ceer must be nontrivial.

We give the requirements the following priority ordering:

$$P_0 < Q_0 < \cdots < P_n < Q_n < \cdots$$

Strategy for P_j. The strategy works with two parameters $a = a_j, b = b_j$, and with indices u, v (given by the Recursion Theorem) of $[a]_R$ and $[b]_R$, respectively:

(1) Pick two new parameters $a = a_j, b = b_j$;
(2) wait for $\varphi_j(a,b,u,v)$ to converge;
(3) if $\varphi_j(a,b,u,v)$ converges to m, say, then add m to $[a]_R$, unless already in $[a]_R \cup [b]_R$

The outcomes for the strategy are clear: If we wait forever for $\varphi_j(a,b,u,v)$ to converge, then φ_j is not total and thus cannot be a uniform total productive function for R. Otherwise, $[a]_R \cap [b]_R = \emptyset$, but $\varphi_j(a,b,u,v) \in [a]_R \cup [b]_R \subseteq W_u \cup W_v$. Hence $\varphi_j(a,b,\_,\_)$ does not witness effective inseparability of the pair $[a]_R$ and $[b]_R$.

Strategy for Q_j. Suppose at a stage s, we have defined f_j on a finite set of quadruples, and $s+1$ is $\langle i,j\rangle$-expansionary. Then

(1) we extend f_j to the least quadruple (by code) not yet in the domain of f_j. Suppose that this quadruple is (a,b,u,v): Define $f_j(a,b,u,v)=m$ where m is new (meaning that m is a number that has never appeared so far in the construction);
(2) for every already defined value $f_j(a',b',u',v')=m'$,
   (a) if at the current stage, $m'\in W_{u'}$ then R-collapse m' and b';
   (b) if at the current stage, $m'\in W_{v'}$ then R-collapse m' and a'.

If we have infinitely many stages that are $\langle i,j\rangle$-expansionary, then f_j is a total computable function. Given a,b such that $[a]_R\cap[b]_R=\emptyset$ and $[a]_R\subseteq W_u$, $[b]_R\subseteq W_v$, we have that $f_j(a,b,u,v)\notin W_u\cup W_v$, otherwise the construction makes $W_u\cap W_v\neq\emptyset$.

Construction. We say that P_j *requires attention at stage* $s+1$ if either $a_{j,s}$, $b_{j,s}$ are undefined, or $a=a_{j,s}$ and $b=b_{j,s}$ are defined, and $\varphi_{j,s}(a,b,u,v)$ is defined and equal to m, say, but $m\notin W_{u,s}\cup W_{v,s}$; we *initialize* P_j *at* s by letting $a_{j,s}$ and $b_{j,s}$ be undefined. We say that Q_j *requires attention at stage* $s+1$ if $s+1$ is $\langle i,j\rangle$-expansionary; we *initialize* Q_j *at* s by letting $f_{j,s}=\emptyset$.

We define R by computable approximations, as in Lemma 1.1.

Stage 0. Let $R^0=\mathrm{Id}$; $f_{j,0}=\emptyset$; and let $a_{j,0}$ and $b_{j,0}$ be undefined for all j;

Stage $s+1$. Let N be the least requirement with index $\leq s$ that requires attention at stage $s+1$ (clearly there is always such a requirement), and pick the least such N:

If $N=P_j$: Perform in order the following items:

(1) If $a_{j,s}$ and $b_{j,s}$ are undefined, then choose $a_{j,s+1}$ and $b_{j,s+1}$ to be new numbers.
(2) Otherwise, suppose $a=a_{j,s}$ and $b=b_{j,s}$: Proceed as explained in (3) in the strategy for P_j.

If $N=Q_j$: Proceed as explained in the strategy for Q_j. After acting on N, end the stage, initialize all lower-priority requirements, and go to the next stage.

Finally, let $R=\bigcup_s R^s$.

Verification. The verification is based on the following lemmas.

Lemma 5.3. *If N eventually stops being initialized, then N is satisfied. Moreover R is not trivial.*

Proof. The claim is by induction on the priority rank of N. Suppose that the claim is true of every $N'<N$. If $N=P_j$ then clearly $a_j=\lim_s a_{j,s}$ and $b_j=\lim_s b_{j,s}$ exist, and $[a_j]_R\cap[b_j]_R=\emptyset$, and if $\varphi_j(a,b,u,v)$ is defined and equal to m (where u,v are indices of the R-equivalence classes of a and b, respectively), then $m\in W_u\cup W_v$, thus $\varphi_j(a_j,b_j,\_,\_)$ does not witness effective inseparability of $[a_j]_R$ and $[b_j]_R$. If $N=Q_j$ then Q_j, after its last initialization, is eventually able to build its own function f_j, which is total if there are infinitely many $\langle i,j\rangle$-expansionary stages. The proof shows also that $[a_0]_R\cap[b_0]_R=\emptyset$, thus R is not trivial. □

Lemma 5.4. *If $i \in S$ then R is u.e.i.; otherwise R is not u.e.i.*

Proof. If $i \in S$ and j is the least number for which there exist infinitely many $\langle i, j\rangle$-expansionary stages, then f_j (the function built by Q_j after its last initialization) is the desired uniform productive function.

If there is no $\langle i, j\rangle$ with infinitely many expansionary stages, then all strategies in the construction are finitary, so every requirement is eventually not initialized, and by Lemma 5.3 above, every P_j is satisfied. □

This completes the proof of the theorem. □

We conclude with the following question, for which Theorem 3.19 seems to suggest an affirmative answer:

Question 5.5. *Is $\{x : R_x \text{ is e.i.}\}$ a Π^0_4-complete set?*

References

[1] H. Becker and A. S. Kechris. *The Descriptive Set Theory of Polish Group Actions*, volume 232 of *London Mathematical Society Lecture Notes Series*. Cambridge University Press, 1996.

[2] C. Bernardi. On the relation provable equivalence and on partitions in effectively inseparable sets. *Studia Logica*, 40:29–37, 1981.

[3] C. Bernardi and F. Montagna. Equivalence relations induced by extensional formulae: Classifications by means of a new fixed point property. *Fund. Math.*, 124:221–232, 1984.

[4] C. Bernardi and A. Sorbi. Classifying positive equivalence relations. *J. Symbolic Logic*, 48(3):529–538, 1983.

[5] J. Case. Periodicity in generations of automata. *Math. Syst. Th.*, 8:15'96–32, 1974.

[6] D. Cenzer, V. S. Harizanov, and J. B. Remmel. Σ^0_1 and Π^0_1 equivalence structures. *Ann. Pure Appl. Logic*, 162(7):490–503, 2011.

[7] J. P. Cleave. Creative functions. *Z. Math. Logik Grundlag. Math.*, 7:205–212, 1961.

[8] S. Coskey, J. D. Hamkins, and R. Miller. The hierarchy of equivalence relations on the natural numbers under computable reducibility. *Computability*, 1:15–38, 2012.

[9] Yu. L. Ershov. Positive equivalences. *Algebra and Logic*, 10(6):378–394, 1973.

[10] Yu. L. Ershov. Theorie der Numerierungen I. *Z. Math. Logik Grundlag. Math.*, 19:289–388, 1973.

[11] Yu. L. Ershov. Theorie der Numerierungen II. *Z. Math. Logik Grundlag. Math.*, 19:473–584, 1975.

[12] Yu. L. Ershov. Theorie der Numerierungen III. *Z. Math. Logik Grundlag. Math.*, 23:289–371, 1977.

[13] Yu. L. Ershov. *Theory of Numberings*. Nauka, Moscow, 1977.

[14] E. Fokina and S. D. Friedman. Equivalence relations on computable structures. In *CiE 2009*, volume 5635 of *Lecture Notes in Computer Science*, pages 198–207, Berlin, Heidelberg, 2009. Springer–Verlag.

[15] E. Fokina and S. D. Friedman. On Σ^1_1 equivalence relations over the natural numbers. *MLQ Math. Log. Q.*, 58(1–2):113–124, 2012.

[16] E. Fokina, S. D. Friedman, V. Harizanov, J. F. Knight, C. McCoy, and A. Montalban. Isomorphism relations on computable structures. *J. Symbolic Logic*, 77(1):122–132, 2012.

[17] E. Fokina, S. D. Friedman, and A. Nies. Equivalence relations that are Σ^0_3-complete for computable reducibility. To appear.

[18] E. Fokina, S. D. Friedman, and A. Törnquist. The effective theory of Borel equivalence relations. *Ann. Pure Appl. Logic*, 161:837–850, 2010.

[19] S. Gao. *Invariant descriptive set theory.* Pure and Applied Mathematics (Boca Raton). CRC Press, Boca Raton, FL, 2009.

[20] S. Gao and P. Gerdes. Computably enumerable equivalence realations. *Studia Logica*, 67:27–59, 2001.

[21] A. H. Lachlan. Initial segments of one-one degrees. *Pac. J. Math.*, 29:351–366, 1969.

[22] A. H. Lachlan. A note on positive equivalence relations. *Z. Math. Logik Grundlag. Math.*, 33:43–46, 1987.

[23] A. I. Mal'tsev. Towards a theory of computable families of objects. *Algebra i Logika*, 3(4):5–31, 1963.

[24] L. San Mauro. Forma e complessità. uno studio dei gradi delle relazioni di equivalenza ricorsivamente enumerabili. Master's thesis, University of Siena, July 2011. In Italian.

[25] F. Montagna. Relative precomplete numerations and arithmetic. *J. Philosphical Logic*, 11:419–430, 1982.

[26] J. Myhill. Creative sets. *Z. Math. Logik Grundlag. Math.*, 1:97–108, 1955.

[27] A. Nies. Undecidable fragments of elementary theories. *Algebra Universalis*, 35:8–33, 1996.

[28] P. Odifreddi. *Classical Recursion Theory (Volume II)*, volume 143 of *Studies in Logic and the Foundations of Mathematics*. North-Holland, Amsterdam, 1999.

[29] H. Rogers, Jr. *Theory of Recursive Functions and Effective Computability*. McGraw-Hill, New York, 1967.

[30] V. Yu. Shavrukov. Remarks on uniformly finitely positive equivalences. *Math. Log. Quart.*, 42:67–82, 1996.

[31] R. Smullyan. *Theory of Formal Systems*. Princeton University Press, Princeton, New Jersey, 1961. Annals of Mathematical Studies Vol 47.

[32] R. I. Soare. *Recursively Enumerable Sets and Degrees*. Perspectives in Mathematical Logic, Omega Series. Springer-Verlag, Heidelberg, 1987.

[33] A. Visser. Numerations, λ-calculus & arithmetic. In J. P. Seldin and J. R. Hindley, editors, *To H. B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism*, pages 259–284. Academic Press, London, 1980.

[34] P. R. Young. Notes on the structure of recursively enumerable sets. *Not. Amer. Math. Soc.*, 10:586, 1963.

(Andrews, Lempp, Miller) Department of Mathematics, University of Wisconsin, Madison, WI 53706-1388, USA
E-mail address: andrews@math.wisc.edu
URL: http://www.math.wisc.edu/~andrews/

E-mail address: lempp@math.wisc.edu
URL: http://www.math.wisc.edu/~lempp/

E-mail address: jmiller@math.wisc.edu
URL: http://www.math.wisc.edu/~jmiller/

(Ng) Division of Mathematical Sciences, School of Physical & Mathematical Sciences, College of Science, Nanyang Technological University, SINGAPORE
E-mail address: kmng@ntu.edu.sg
URL: http://www.ntu.edu.sg/home/kmng/

(San Mauro) Scuola Normale Superiore, Perfezionamento in Discipline Filosofiche, I-56126 Pisa, ITALY
E-mail address: luca.sanmauro@sns.it
URL: http://sns-it.academia.edu/LucaSanMauro

(Sorbi) Dipartimento di Ingegneria dell'Informazione e Scienze Matematiche, Universitá degli Studi di Siena, Via Roma, 56, I-53100 Siena, ITALY
E-mail address: sorbi@unisi.it
URL: http://www.mat.unisi.it/personalpages/sorbi/public_html/