A SURVEY ON UNIVERSAL COMPUTABLY ENUMERABLE EQUIVALENCE RELATIONS

URI ANDREWS, SERIKZHAN BADAEV, AND ANDREA SORBI

Abstract. We review the literature on universal computably enumerable equivalence relations, i.e. the computably enumerable equivalence relations (ceers) which are Σ^0_1-complete with respect to computable reducibility on equivalence relations. Special attention will be given to the so-called uniformly effectively inseparable (u.e.i.) ceers, i.e. the nontrivial ceers yielding partitions of the natural numbers in which each pair of distinct equivalence classes is effectively inseparable (uniformly in their representatives). The u.e.i. ceers comprise infinitely many isomorphism types. The relation of provable equivalence in Peano Arithmetic plays an important role in the study and classification of the u.e.i. ceers.

1. Introduction

Recently there has been a growing interest in studying and classifying equivalence relations on the set ω of natural numbers, by mean of the so-called *computable reducibility*, where, given equivalence relations R and S on ω, we say that R is *computably reducible* (or simply, *reducible*) to S (in symbols: $R \leqslant S$), if there exists a computable function f such that

$$(\forall x, y)[x \ R \ y \Leftrightarrow f(x) \ S \ f(y)].$$

The first systematic study of this reducibility goes back perhaps to Ershov [13, 14], as an alternative way of looking at monomorphisms in the category of numbered sets. An obvious related notion is that of completeness: if $\mathcal{A}$ is a class of equivalence relations on ω, one says that R is $\mathcal{A}$ *complete*, if $R \in \mathcal{A}$, and $S \leqslant R$, for every $S \in \mathcal{A}$. This reducibility, and its related notion of completeness, have been successfully applied to measure the complexity of equivalence relations naturally arising in mathematics, and in particular in computable model theory and in computability theory. For instance, the isomorphism relations for various familiar classes of computable structures (identified with numbers via suitable numberings) are Σ^1_1 complete: this includes computable groups, computable torsion abelian groups, computable torsion-free abelian groups, abelian p-groups, see [17]. Other interesting mathematical applications of reducibility $\leqslant$ appear in [16, 11, 20, 21, 15].

This paper is a survey (far from being exhaustive) on Σ^0_1-universal equivalence relations, henceforth called *universal ceers*: we shall use the acronym "ceer" for "computably enumerable equivalence relation"; ceers are called *positive* equivalence relations in the Russian literature. In section 2 we focus our attention on some classes of universal ceers of particular importance in logic and computability

2010 *Mathematics Subject Classification.* 03D25.

Key words and phrases. Computably enumerable equivalence relation; computable reducibility on equivalence relations.

Andrews was partially supported by NSF grant DMS-1201338.

Badaev was partially supported by Grant 3952/GF4 of the Science Committee of the Republic of Kazakhstan.

Sorbi is a member of INDAM-GNSAGA; he was partially supported by Grant 3952/GF4 of the Science Committee of the Republic of Kazakhstan, and by PRIN 2012 "Logica Modelli e Insiemi ".

theory. It is interesting to notice that the first example of a nontrivial and mathematically interesting universal ceer appears in the book [24], where Miller III builds a finitely presented group G for which the ceer provided by equality $=_G$ in G, is universal. If this example was most likely not motivated by any specific interest in ceers and computable reducibility, Ershov [12] on the contrary, in this case clearly motivated by studying ceers under $\leqslant$, pointed out another mathematically interesting universal ceer, see Theorem 2.5. Another universal ceer of special interest, first pointed out in [8], is the relation of provable equivalence in Peano Arithmetic, denoted by $\sim_{PA}$, which relates two numbers if the two sentences coded by these numbers are provably equivalent in PA. The class of nontrivial ceers which are quotients of $\sim_{PA}$ (i.e. computably isomorphic to extensions of $\sim_{PA}$) form the class of the so-called uniformly finitely precomplete (u.f.p.) ceers, which are all universal. Inside this class we find two special isomorphism types: the so-called e-complete ceers (which turn out to be computably isomorphic to $\sim_{PA}$), and the precomplete ceers (which turn out to be computably isomorphic to the restriction of $\sim_{PA}$ to the Σ^0_n-sentences, for any fixed n).

As in the case of universality with respect to m-, or 1-reducibility, for c.e. sets, or pairs of disjoint c.e. sets (where the universal sets coincide with the creative sets, and the universal pairs of disjoint c.e. sets coincide with the effectively inseparable pairs), the notions of creativeness and effective inseparability play an important role in the investigation of universal ceers. Not only can one show that a u.f.p. ceer R yields a partition of ω such that any disjoint pair $([a]_R, [b]_R)$ of equivalence classes are effectively inseparable uniformly in a, b, but it turns out that this latter notion by itself suffices to give universality: every uniformly effectively inseparable (u.e.i.) ceer R (i.e. a nontrivial ceer yielding a uniformly effectively inseparable partition of ω) is universal.

Unlike classical isomorphism theorems (in particular, Myhill's theorem on computable isomorphisms of creative sets, and Smullyan's theorem on computable isomorphisms of e.i. pairs), uniform effective inseparability for ceers does not imply computable isomorphism. Infinitely many distinct computable isomorphism types for u.e.i. ceers appear already at the level of u.f.p. ceers. Moreover, a recent result in [3] shows that there are u.e.i. ceers that are not u.f.p.

The class of u.f.p. ceers is however partitioned into infinitely many computable isomorphism types.

In section 5 we review a characterization (see [1]) of universal ceers in terms of a jump operation on ceers, due to [19]: namely, a ceer is universal if and only if its jump is reducible to it.

In more than one occasion, we give new and simplified proofs of classical results, including for instance universality of u.f.p. ceers, and isomorphism of e-complete ceers.

1.1. **Terminology and notations.** We use standard computability theoretic terminology and notation, which can be found in the textbooks [28, 31, 10]. We often identify finite sets with their canonical indices: so when for a function f we write $f(D)$ where D is a finite set, then we in fact mean $f(u)$, with $F = D_u$.

Given any set X and any equivalence relation R, we write $[X]_R = \{y : (\exists x)[y\ R\ x]\}$; and $[x]_R = [\{x\}]_R$ denotes the R-equivalence class of R.

The following category theoretic terminology is adapted from [13, 14], which study the category of numberings.

Definition 1.1. Given equivalence relations R, S on ω, a *morphism* $\mu : R \longrightarrow S$ is a function from $\omega_{/R}$ to $\omega_{/S}$ (i.e. between the quotient sets), for which there exists a computable function $f : \omega \to \omega$ such that $\mu([x]_R) = [f(x)]_S$, for all x; we say in this case that f *induces* μ; a *monomorphism* is a

1-1 morphism, an *isomorphism* is an onto monomorphism. An *endomorphism* for R is a morphism $\mu : R \longrightarrow R$.

Remark 1.2. We observe that if $\mu : R \longrightarrow S$ is an isomorphism, and all R- and S-equivalence classes are infinite, then by a standard back and forth argument, it is easy to see that there is a computable permutation of ω that induces μ. Since the special classes of ceers in which we will be interested yield, as it turns out, partitions into infinite equivalence classes, henceforth for these ceers we will be justified in taking "isomorphism" as synonym of "computable isomorphism".

Lemma 1.3. *If R, S are ceers then $R \leqslant S$ if and only if there is a monomorphism $\mu : R \longrightarrow S$.*

Proof. Easy. □

In the same vein, we can define a *partial morphism* from R to S to be a partial function μ from $\omega_{/R}$ to $\omega_{/S}$ for which there is a partial computable function φ such that: (1) if $\mu([x]_R)$ is defined, then there is x' such that $x \; R \; x'$ and $\mu([x]_R) = [\varphi(x')]_S$; 2) $[\mathrm{domain}(\varphi)]_R = \{x : [x]_R \in \mathrm{domain}(\mu)\}$.

1.2. **Indexing.** Throughout the paper, we refer to the indexing $\{R_e : e \in \omega\}$ of all ceers, where R_e is the equivalence relation generated by W_e (viewed as a set of pairs).

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 $[x]_{R^s} \cap [y]_{R^s} = \varnothing$, 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 1.4. *There exists a sequence $\{R^s_e : e, s \in \omega\}$ of equivalence relations such that the set $\{\langle e, x, y, s\rangle : x \; R^s_e \; y\}$ is computable (in fact, we may even assume that one can effectively find the canonical index of $[x]_{R^s}$, and we can decide, given e, s whether $R^s_e = R^{s+1}_e$), and the sequence $\{R^s_e : s \in \omega\}$ 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 $R \smallsetminus \{\langle x, x\rangle : x \in \omega\}$ 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. □

One could alternatively consider the following numbering, suggested by Ershov [12]: let

$$x \; S_e \; y \Leftrightarrow (\exists m, n)[\varphi^m_e(x)\!\downarrow = \varphi^n_e(y)\!\downarrow],$$

where, given a partial function ψ, $\psi^n(x)$ denotes the n-th iterate of ψ on x, where $\psi^0(x) = x$, and of course $\psi^n(x)$ converges if and only if both $\psi^{n-1}(x)$ and $\psi(\psi^{n-1}(x))$ converge. We may also write S_{φ_e} for S_e. Indeed, if R is a ceer, then $R = S_\varphi$ where φ is the partial computable function $\varphi(x) = (\mu\langle y, s\rangle).\,[x \; R_s \; y \,\&\, y < x]_0$, where we refer to some computable approximation $\{R_s\}$ to R.

1.3. **Some special classes of ceers.** We now introduce some important classes of ceers, which will be shown to be universal in next section.

Definition 1.5. Let R be an equivalence relation on ω.

(1) [23] R is *precomplete* if there exists a computable function $f(e,x)$ (called a *totalizer of* R) such that, for all e, x,
$$\varphi_e(x)\downarrow \Rightarrow \varphi_e(x)\ R\ f(e,x).$$
Moreover, $f(e,\_)$ is called an *R-totalizer of* φ_e, or alternatively we say that $f(e,\_)$ *makes* φ_e *total modulo* R.

(2) [25] R is *uniformly finitely precomplete* (or *u.f.p.* for short) if there exists a computable function $f(D,e,x)$ such that for every finite set D and every e, x,
$$\varphi_e(x)\downarrow \in [D]_R \ \Rightarrow \varphi_e(x)\ R\ f(D,e,x).$$
Moreover, $f(\_,e,\_)$ is called an *R-totalizer of* φ_e, or alternatively we say that $f(\_,e,\_)$ *makes* φ_e *total modulo* R.

(3) [6, 1] We say that R is *uniformly effectively inseparable* (or *u.e.i.* for short) if there is a *uniform productive function*, i.e., a partial computable function $p(a,b,u,v)$ such that if $[a]_R \cap [b]_R = \varnothing$ then
$$(\forall u,v)[[a]_R \subseteq W_u \,\&\, [b]_R \subseteq W_v \,\&\, W_u \cap W_v = \varnothing \Rightarrow p(a,b,u,v)\downarrow \notin W_u \cup W_v].$$

Remark 1.6. We note that, as in the case of effective inseparability for pairs of c.e. sets, if R is a u.e.i. ceer then we can in fact assume that $p(a,b,u,v)$ be total. Indeed, if p is partial computable, we can always assume that if $a \not R\, b$ then the function $p(a,b,\_,\_)$ is total, as from any partial productive function for a pair of disjoint c.e. sets, one can uniformly find a total productive function for that pair: this is similar to showing that from any productive function for a productive set, one can uniformly find a total productive function for that set, see [28]. Having such a function p, define a total productive function q for R as follows:
$$q(a,b,u,v) = \begin{cases} 0, & \text{if } [a]_R \cap [b]_R \neq \varnothing \preceq p(a,b,u,v)\downarrow \\ p(a,b,u,v), & \text{otherwise,} \end{cases}$$
where given two c.e. relations $U := (\exists x)A(x)$ and $V := (\exists x)B(x)$ in Σ_1-normal form, with A, B decidable, we write as usual $U \preceq V := (\exists x)A(x) \,\&\, (\forall y \leqslant x)\neg B(y)$.

Lemma 1.7. *The classes of Definition 1.5 are closed under isomorphisms, and are upwards $\subseteq$-closed.*

Proof. Straightforward. □

Remark 1.8. Throughout the paper, when we refer to an equivalence relation R as lying in any of the three classes of Definition 1.5, we will also always assume that R is not trivial (i.e. there are two numbers which are non-R-equivalent).

2. Precomplete and uniformly finitely precomplete ceers

As promised, in this section we show that the special ceers introduced in Section 1.3 are all universal.

2.1. **Precomplete ceers.** Let us begin our trip through the land of universal ceers by looking at precomplete ceers. First let us recall some important properties of precomplete equivalence relations. The following theorem is in fact a characterization of all precomplete equivalence relations (including in this case the trivial one), not only the computably enumerable ones.

Theorem 2.1 (Ershov's Fixed Point Theorem)**.** *An equivalence relation R is precomplete if and only if there is a computable function* fix *such that, for every n,*

$$\varphi_n(\mathrm{fix}(n)) \downarrow \Rightarrow \varphi_n(\mathrm{fix}(n))\ R\ \mathrm{fix}(n).$$

Proof. $\Rightarrow$. If R is precomplete then let $\hat{u}(x)$ be a computable function that makes $\varphi_x(x)$ total modulo R. Let $\varphi_{s(n)} = \varphi_n \circ \hat{u}$, and define $\mathrm{fix} = \hat{u} \circ s$. Then if $\varphi_n(\mathrm{fix}(n)) \downarrow$, then $\varphi_n(\mathrm{fix}(n)) = \varphi_n(\hat{u} \circ s(n)) = \varphi_n \circ \hat{u} \circ s(n) = \varphi_{s(n)}(s(n))\ R\ \hat{u} \circ s(n) = \mathrm{fix}(n)$.

$\Leftarrow$. Given fix and a partial computable φ, let $\varphi_{f(x)}(y) = \varphi(x)$. Then we claim that $g = \mathrm{fix} \circ f$ makes φ total modulo R. If $\varphi(x) \downarrow$, then $\varphi(x) = \varphi_{f(x)}(\mathrm{fix} \circ f(x))\ R\ \mathrm{fix} \circ f(x) = g(x)$. □

Another important property of precomplete equivalence relations is the Padding Lemma.

Theorem 2.2 (Padding Lemma)**.** *For every precomplete R there exists a* 1-1 *total computable $p(x, y)$ such that, for all x, m, $p(x, m)\ R\ x$. Hence, all R-equivalence classes contain infinite c.e. sets, and R has an injective totalizer.*

Proof. Let R be a precomplete equivalence relation. We show that there is a computable p with the desired properties which is injective in the second argument; we leave it as an exercise to show that one can get an injective totalizer. We need to show that from any finite set $F = \{n_1, \ldots, n_k\}$ of numbers such that $n_1\ R \cdots R\ n_k$ we can uniformly find $n \notin F$ such that $n\ R\ n_1$. Let $G(e, x)$ be a totalizer for R. Then by the Recursion Theorem, let e be such that

$$\varphi_e(x) = \begin{cases} n_1 & \text{if } G(e, 0) \notin F, \\ \max F + 1 & \text{otherwise.} \end{cases}$$

Then the number

$$n = \begin{cases} G(e, 0) & \text{if } G(e, 0) \notin F, \\ \max F + 1 & \text{otherwise.} \end{cases}$$

is the desired number. Indeed, $n \notin F$ since either $n = \max F + 1$ or $n = G(e, 0)$ if $G(e, 0) \notin F$. In the former case, $n = \max F + 1 = \varphi_e(0)\ R\ G(e, 0) \in F$. So, n is R-equivalent to an element of F, so to n_1. In the latter case, $n = G(e, 0)\ R\ \varphi_e(0) = n_1$. □

Notice that the usual padding lemma for the standard numbering $\{\varphi_e\}$ of the partial computable functions is a corollary of the previous result, as the equivalence relation, in x, y, $\varphi_x = \varphi_y$ is easily seen to be precomplete, see [23].

2.2. **Examples of precomplete ceers.** Recall that a partial computable function u is called *universal*, if there exists a computable function $f(e, x)$ such that $\varphi_e(x) = u(f(e, x))$. By the Padding Lemma for the numbering $\{\varphi_e\}$, we can also assume that f is 1-1.

The following result is attributed in [12] to Lachlan.

Lemma 2.3. *If u is a universal unary partial computable function then S_u is precomplete.*

Proof. If f witnesses that u is universal, and $\varphi_e(x)\downarrow$, then $u^1(\varphi_e(x)) = u(f(e,s))$, hence $\varphi_e(x)\ S_u\ f(e,x)$, which shows that $f(e,\_)$ is a totalizer for φ_e. □

Assume that first order Peano Arithmetic PA is Σ_1-sound, and for every $n \geqslant 1$ let $T_n(v)$, be a Σ_n-truth predicate, i.e., for all Σ_n-sentences σ,

$$PA \vdash \sigma \leftrightarrow T_n(\overline{\ulcorner\sigma\urcorner})$$

where $\ulcorner\_\urcorner$ is a suitable Gödel numbering for all sentences in the language of PA, and $\overline{m}$ denotes the numeral term for the number m.

For every number x there is a Σ_1-formula $F_x(u,v)$ (in fact, $F_x(u,v) := F(\overline{x},u,v)$ for some Σ_1-formula F) *representing* φ_x *in* PA, i.e. such that

$$\varphi_x(n) = m \Leftrightarrow PA \vdash F_x(\overline{n},\overline{m}).$$

We may assume that for every number m, $PA \vdash F_x(\overline{m},v) \wedge F_x(\overline{m},v') \rightarrow v = v'$.)

Define $\sim_n$ on ω by

$$\ulcorner\sigma\urcorner_n \sim_n \ulcorner\tau\urcorner_n \Leftrightarrow T \vdash \sigma \leftrightarrow \tau$$

where $\ulcorner\_\urcorner_n$ is a suitable Gödel numbering identifying Σ_n sentences (which form an infinite c.e. set, and therefore is a set computably isomorphic to ω) with numbers: notice that we use here $\ulcorner\_\urcorner_n$ instead of $\ulcorner\_\urcorner$, as otherwise the domain of $\sim_n$ would be a proper subset of ω. Then $\sim_n$ is a precomplete ceer. Given the relevance of this example, we sketch the proof of why $\sim_n$ is precomplete.

Theorem 2.4. $\sim_n$ *is a precomplete ceer.*

Proof. We limit ourselves to the case $n = 1$. Given a partial computable function φ, let F be a representing Σ_1 formula for the partial computable function ψ, where

$$\psi(\ulcorner\sigma\urcorner_1) = \begin{cases} \ulcorner\tau\urcorner, & \text{if } \varphi(\ulcorner\sigma\urcorner_1)\downarrow = \ulcorner\tau\urcorner_1, \\ \uparrow, & \text{if } \varphi(\ulcorner\sigma\urcorner_1)\uparrow. \end{cases}$$

Define

$$f(m) = \ulcorner(\exists v)[F(\overline{m},v) \wedge T_1(v)]\urcorner_1.$$

(Notice that the formula $(\exists v)[F(\overline{m},v) \wedge T_1(v)]$ is Σ_1.) Assume now that $\varphi(\ulcorner\sigma\urcorner_1)\downarrow = \ulcorner\tau\urcorner_1$, where σ and τ are Σ_1-sentences. Then

$$PA \vdash (\exists v)[F(\overline{\ulcorner\sigma\urcorner_1},v) \wedge T_1(v)] \leftrightarrow F(\overline{\ulcorner\sigma\urcorner_1},\overline{\ulcorner\tau\urcorner}) \wedge T_1(\overline{\ulcorner\tau\urcorner}).$$

But $PA \vdash F(\overline{\ulcorner\sigma\urcorner_1},\overline{\ulcorner\tau\urcorner}) \wedge T_1(\overline{\ulcorner\tau\urcorner}) \leftrightarrow T_1(\overline{\ulcorner\tau\urcorner})$, and $PA \vdash T_1(\overline{\ulcorner\tau\urcorner}) \leftrightarrow \tau$, which implies that $\varphi(\ulcorner\sigma\urcorner_1) \sim_1 f(\ulcorner\sigma\urcorner_1)$. Thus, f is the desired computable function that makes φ total modulo $\sim_1$. □

Other examples of precomplete ceers can be found in [32].

2.3. **The first universality result.** As already remarked in the introduction, one of the earliest nontrivial universality results for ceers was pointed out by Ershov [12].

Theorem 2.5. *If u is a universal unary partial computable function, then S_u is universal.*

Proof. Let u be a universal function and let φ be a partial computable function. As we have observed, we may suppose that there exists a 1-1 computable function g such that $\varphi_e(x) = g(\langle e, x\rangle)$. Thus it is easy to see that there is a computable sequence f_n of computable 1-1 functions such that $\varphi_n = u \circ f_n$. So, by the Recursion Theorem, let e be such that $u \circ f_e = f_e \circ \varphi$ (take a fixed point of a computable h, such that $\varphi_{h(e)} = f_e \circ \varphi$). Let $f = f_e$: then $f \circ \varphi = u \circ f$. Next, by induction on n it is easy to see that for every n, $f \circ \varphi^n = u^n \circ f$. It follows that for every m, n, if $\varphi^m(x) \downarrow = \varphi^n(y) \downarrow$ then $f(\varphi^m(x)) \downarrow = f(\varphi^n(y))) \downarrow$, thus $u^m(f(x) \downarrow = u^n(f(y) \downarrow$. On the other hand, if $u^m(f(x)) \downarrow = u^n(f(y)) \downarrow$ then $f(\varphi^m(x)) \downarrow = f(\varphi^n(y)) \downarrow$, and by injectivity, $\varphi^m(x) \downarrow = \varphi^n(y) \downarrow$. This shows that f reduces S_φ to S_u. Since for every ceer R, there is a partial computable φ such that $R = S_\varphi$, we have proved that S_u is universal. □

2.4. **Precomplete ceers are isomorphic.** The precomplete ceers form a single isomorphism type, as shown by Lachlan [22].

Theorem 2.6 ([22])**.** *If R, S are precomplete ceers then R is computably isomorphic to S, i.e., there exists a permutation h of ω which reduces R to S.*

Proof. We can assume that every ceer R has approximations $\{R_s\}$ and $\{\hat{R}_s\}$ satisfying Lemma 1.4 and in addition:

$$R_{s+1} - R_s \neq \varnothing \Rightarrow s+1 \text{ odd}$$
$$S_{s+1} - S_s \neq \varnothing \Rightarrow s+1 \text{ even}.$$

Let R, S be precomplete ceers, with corresponding computable approximations $\{R_s\}$ and $\{S_s\}$, as above: R may change only at odd stages, and S may change only at even stages. (Although not necessary, these additional properties of the approximations simplify the construction, since they make sure that changes for R (respectively, S) may appear only at stages when we really deal with R (respectively, S). In fact since all R- and S-equivalence classes are infinite, by Lemma 1.4 we could even assume in this case that at each stage exactly one change happens when we deal with the corresponding ceer.) Let F and G be injective totalizers for R and S respectively.

We will define two computable sequences $a_0, a_1, \ldots, a_s, \ldots$ and $b_0, b_1, \ldots, b_s, \ldots$, such that the assignment $a_s \mapsto b_s$ (we say in this case that a_s and b_s *match*) satisfies, for all i, j,

$$a_i \; R \; a_j \Leftrightarrow b_i \; S \; b_j,$$

and $\omega = \{a_s : s \in \omega\} = \{b_s : s \in \omega\}$. We start up with four numbers c_0, c_1, d_0, d_1 such that $c_0 \not R \, c_1$ and $d_0 \not S \, d_1$.

By the Double Recursion Theorem, we will assume that we control indices e, z of partial computable functions φ_e and φ_z. At the beginning of each stage $s+1$, we assume that, for all $i, j < s$,

$$a_i \; R_s \; a_j \Leftrightarrow b_i \; S_s \; b_j.$$

We use in the following the symbols e', z', e'', z'' to represent suitable new indices of φ_e and φ_z, by the Padding Lemma. At stage $s+1$ we say, for $i < s$, that $[a_i]_{R_s}$ is *right available* if there is $a \in [a_i]_{R_s}$ such that $\varphi_{e,s}(a)$ is undefined, and a already matches with a number chosen as $b = G(e', a) \in [b_i]_{S_s}$, with $\varphi_e = \varphi_{e'}$; similarly, we say that $[b_i]_{R_s}$ is *left available* if there is $b \in [b_i]_{R_s}$ such that $\varphi_{z,s}(b)$ is undefined, and b already matches with some number chosen as $a = F(z', b) \in [a_i]_{R_s}$, with $\varphi_z = \varphi_{z'}$. At the end of the stage, we define a new pair (a_s, b_s).

If a_i and b_i match, we assume by induction that either $[a_i]_{R_s}$ is right available or $[b_i]_{R_s}$ is left available.

Step 0. $\varphi_{e,0}(i)$ and $\varphi_{z,0}(i)$ are undefined for all i.

Step $s+1$. Distinguish whether $s+1$ is odd or even:

$s+1$ *odd*. Perform in the order the following actions:

(1) Suppose there are $i<j$ such that a_i and a_j are R-collapsed at $s+1$. There are two subcases:
   (a) at least one among $[a_i]_{R_s}$ and $[a_j]_{R_s}$ is right available, say $a \in [a_i]_{R_s}$ is such that $\varphi_{e,s}(a)$ is undefined, and matches with $b \in [b_i]_{S_s}$, of the form $b = G(e', a)$: then define $\varphi_e(a) = b_j$. This has the effect that
   $$b_i \; S \; b = G(e', a) \; S \; \varphi_{e'}(a) = \varphi_e(a) = b_j;$$
   (b) neither $[a_i]_{R_s}$ nor $[a_j]_{R_s}$ is right available: then $[b_i]_{S_s}$ and $[b_j]_{S_s}$ are both left available. Say $b \in [b_i]_{S_s}$, $b' \in [b_j]_{S_s}$ are such that $\varphi_{z,s}(b)$ and $\varphi_{z,s}(b')$ are still undefined and match with $a = F(z', b) \in [a_i]_{R_s}$ and $a' = F(z'', b') \in [a_j]_{R_s}$, respectively: then define $\varphi_z(b) = c_0$, and $\varphi_z(b') = c_1$. Using the fact that $\varphi_z = \varphi_{z'} = \varphi_{z''}$, this has the effect that
   $$c_0 = \varphi_z(b) \; R \; F(z, b) = a \; R \; a_i$$
   $$c_1 = \varphi_z(b') \; R \; F(z, b') = a' \; R \; a_j,$$
   giving $c_0 \; R \; c_1$: this case *cannot* happen.

(2) Finally we define (a_s, b_s). Let a_s be the least number not in $\{a_i : i < s\}$. Let e' be an index of φ_e chosen by the Padding Lemma and the injectivity of G to be such that
$$G(e', a_s) \notin \bigcup_{i<s} [b_i]_{S_s};$$
and define $b_s = G(e', a_s)$. Now we check that the inductive assumption on availability still holds: suppose we see that a_i and a_j are R-equivalent, and b_i and b_j need to be made S-equivalent, thus we act by making $\varphi_e(a) = b_j$ (where $a \in [a_i]_{R_s}$ which is right available). If the class $[a_i]_{R_{s+1}} \cup [a_i]_{R_{s+1}}$ fails to be right available, then $[a_j]_{R_s}$ was not right available, so $[b_j]_{S_s}$ was left available by the inductive hypothesis. Therefore, $[b_j]_{S_{s+1}}$ is still left available.
   Lastly, we check the inductive assumption for the new pair a_s, b_s. Since we only define φ_e in the operation above, since a_s is not in $\{a_i : i < s\}$, we have $\varphi_{e,s+1}(a_s) \uparrow$. We chose b_s to make a_s right available.

$s+1$ *even*. Perform the same steps, inverting the roles between the a's and the b's, and between F and G.

It is easy to see that for every pair of numbers i, j,
$$a_i \; R \; a_j \Leftrightarrow b_i \; S \; b_j$$
and $\omega = \{a_i : i \in \omega\} = \{b_i : i \in \omega\}$.

Finally, note that we always maintain injectivity when we add a new pair a_s, b_s, and since at odd stages, we enter the least missing number into the domain of the reduction, and at even stages we enter the least missing number into the range of the reduction that this reduction is a permutation of ω. □

Corollary 2.7 ([8])**.** *Every precomplete ceer is universal.*

Proof. By Lemma 2.3, Theorem 2.5, Theorem 2.6 and the fact that for ceers the property of being universal is preserved by isomorphisms. □

The following is an interesting characterization of precomplete ceers.

Corollary 2.8. *Every precomplete ceer R is equal to S_v for some universal function v.*

Proof. Let R be a precomplete ceer and let S_u be the precomplete ceer determined by a universal function u. Then by Theorem 2.6, R and S_u are isomorphic. So, let π be a permutation of ω witnessing the isomorphism of R and S_u. It is straightforward to check that $v = \pi \circ u \circ \pi^{-1}$ is also universal and that $R = S_v$. □

There are interesting extensions of Theorem 2.6, and of Corollary 2.7, due to Shavrukov [29].

Theorem 2.9 ([29])**.** *The following hold:*

(1) Any partial, and not onto monomorphism, induced by some partial computable function, from a ceer R to a precomplete ceer S can be extended to a monomorphism.
(2) Any strictly partial, and not onto monomorphism, induced by some partial computable function, between precomplete ceers R can be extended to an isomorphism.

Proof. We briefly sketch only a proof for item (1), i.e. how to show that every partial, and not onto monomorphism, from a ceer to a precomplete ceer, which is induced by some partial computable function, can be extended to a monomorphism. To prove the second item, combine this extension argument, with a back-and-forth argument in the style of Theorem 2.6, inserting, at odd stages, pairs that guarantee surjectivity.

Let R, S be ceers so that S is precomplete. Let φ be a partial computable function inducing a partial monomorphism from R to S. Suppose we are working with suitable computable approximations $\{R_s\}$ and $\{S_s\}$ (as in Theorem 2.6) to R and S, respectively. Let F be an S-totalizer. We define an assignment $i \mapsto b_i$, such that $i \; R \; j$ if and only if $b_i \; S \; b_j$, and the corresponding monomorphism extends the given partial one. By the Recursion Theorem we also assume that we control the partial computable function φ_e. In the construction, at each stage $s+1$, if i is least in its R_s-equivalence class, then we assume by induction that $\varphi_e(i)$ is still undefined by the end of stage s, unless it has been already defined as $\varphi_e(i) = \varphi(i)$, for the sake of extending φ; to this regard, note that if at some stage we set $\varphi_e(i) = \varphi(i)$ then we regard $\varphi_e(i)$ as already defined, even if $\varphi(i)$ does not as yet converge, as we do so only for numbers i for which eventually $i \in \text{domain}(\varphi)$.

Pick numbers $b \not S \; b'$, with $b, b' \notin [\text{range}(\varphi)]_S$. Such a pair of numbers exists, because we assume that $[\text{range}(\varphi)]_S \neq \omega$ but, on the other hand, the complement of $[\text{range}(\varphi)]_S$ can not be c.e. (see for instance Lemma2.13 below which shows that each pair of distinct equivalence classes of a u.f.p. ceer, and a fortiori of a precomplete ceer, is effectively inseparable).

Take $b_i = F(e, i)$.

Step 0**.** Do nothing; $\varphi_{e,0}(i)$ is undefined for all i.

Step $s+1$**.** We distinguish Cases 1. and 2., depending on whether $s+1$ is odd or even:

(1) ($s+1$ odd.) There are $i < j$ such that i and j R-collapse at stage $s+1$; assume i, j are least in their R_s-equivalence classes:

(a) if $\varphi_e(j)$ is still undefined, then set $\varphi_e(j) = b_i$: since F is an S-totalizer, this will give $b_i \; S \; b_j$, as $b_j = F(e, j) \; S \; \varphi_e(j) = b_i$;
(b) otherwise already $\varphi_e(j) = \varphi(j)$: set $\varphi_e(i) = \varphi(i)$, unless it has been already defined so; since φ induces a partial monomorphism, this fulfils the desired goal (notice that $\varphi(i)$ may be still undefined, but eventually it will converge).

(2) ($s + 1$ even) There are $i < j$ such that i and j are not as yet R-equivalent, but matching b_i, b_j S-collapse:
(a) if $\varphi_e(i)$ and $\varphi_e(j)$ are still undefined, then let $\varphi_e(i) = b$ and $\varphi_e(i) = b'$: this case cannot happen, since F is an S-totalizer, and otherwise we would get $b \; S \; b_i \; S \; b_j \; S \; b'$;
(b) if exactly one of $\varphi_e(i)$ and $\varphi_e(j)$ has been already defined, say $\varphi_e(i) = \varphi(i)$, then take the other one and set it equal to b: in our example, set $\varphi_e(j) = b$; again this case cannot happen, since $b \notin [\mathrm{range}(\varphi)]_S$;
(c) if already $\varphi_e(i) = \varphi(i)$ and $\varphi_e(j) = \varphi(j)$ have been defined, then do nothing, as φ induces a partial monomorphism.

Before leaving stage $s + 1$, we consider i such that $\varphi(i)$ converges for the first time, if any exists: if $\varphi_e(i)$ has not already been defined, (otherwise it has been already stipulated that $\varphi_e(i) = \varphi(i)$), then set $\varphi_e(i) = \varphi(i)$.

Notice that the induction assumption is being preserved. This ends the construction. We skip the remaining details of the verification. □

Remark 2.10. By taking $\varphi = \varnothing$, the first item of Theorem 2.9 gives yet another proof of universality of precomplete ceers.

2.4.1. *Historical remark.* Universality of precomplete ceers was first proved by Bernardi and Sorbi in [8] and appeared before [22]. The proof in [8] used the so-called Anti Diagonal Normalization Theorem by Visser [32].

2.5. **Uniformly finitely precomplete ceers.** The ceer $\sim_{PA}$ is not precomplete because it has a (total) diagonal function, for instance the function induced by the connective $\neg$: we denote this function with the same symbol, namely $\neg\ulcorner\sigma\urcorner = \ulcorner\neg\sigma\urcorner$. Therefore $\sim_{PA}$ does not satisfy the Ershov Fixed Point Theorem, and thus it is not precomplete. However, although not precomplete, $\sim_{PA}$ is "locally" precomplete, i.e., every partial computable function with finite range can be totalized modulo $\sim_{PA}$ since there is some effectively found $n \geqslant 1$ such that all sentences in the range of φ are Σ_n, and thus we can totalize modulo $\sim_n$. This is exactly what led Montagna to introduce the u.f.p. ceers, see Definition 1.5(2).

Corollary 2.11. *Every precomplete ceer is u.f.p. The relation $\sim_{PA}$ is u.f.p., so there are u.f.p. ceers that are not precomplete.*

Proof. The first statement is immediate from the definitions. In order to prove that $\sim_{PA}$ is u.f.p. use the fact that, given a finite D and a sentence x, all sentences in $D \cup \{x\}$ fall into some finite level Σ_n, so that we can use a precompleteness totalizer $F_n(e, x)$ of $\sim_n$, using the fact that a totalizer for $\sim_n$ can be found uniformly in n. Some caution should be taken, since $\sim_{PA}$ and $\sim_n$ refer to different Gödel numbers. □

Lemma 2.12 (Fixed Point Theorem for u.f.p. equivalence relations)**.** *If R is u.f.p. then there exists a computable function* $\mathrm{fix}(D, e)$ *such that, for all D, e,*

$$\varphi_e(\mathrm{fix}(D, e)) \downarrow \in [D]_R \Rightarrow \varphi_e(\mathrm{fix}(D, e)) \; R \; \mathrm{fix}(D, e).$$

Proof. Let $f(D, e, x)$ be a totalizer of R, and let φ_u be so that for all x $\varphi_u(x) = \varphi_x(x)$. Let $s(D, e)$ be a computable function such that

$$\varphi_{s(D,e)}(z) = \varphi_e(f(D, u, z)),$$

and let $\mathrm{fix}(D, e) = f(D, u, s(D, e))$.

Suppose that $\varphi_e(\mathrm{fix}(D, e)) \downarrow \in [D]_R$. Then

$$\varphi_e(\mathrm{fix}(D, e)) = \varphi_e(f(D, u, s(D, e)))) = \varphi_{s(D,e)}(s(D, e)) \downarrow \in [D]_R,$$

and $\varphi_u(s(D, e))\ R\ f(D, u, s(D, e)) = \mathrm{fix}(D, e)$. □

Lemma 2.13. *Every u.f.p. ceer is u.e.i.*

Proof. Let R be a u.f.p. ceer, and let $[a]_R, [b]_R$ be two distinct equivalence classes. Given c.e. sets W_u, W_v, define

$$\psi(x) = \begin{cases} b, & \text{if } (x \in W_u) \preceq (x \in W_v); \\ a, & \text{if } (x \in W_u) \prec (x \in W_v); \\ \uparrow & \text{otherwise} \end{cases}$$

and let $n = \mathrm{fix}(\{a, b\}, e)$ be a fixed point for ψ, given by u.f.p.-ness of R where e is an index of ψ. It is clear that $n \notin W_u \cup W_v$, if $[a]_R \subseteq W_u$, $[b]_R \subseteq W_v$ and $W_u \cap W_v = \varnothing$. Since ψ is defined uniformly in the tuple (a, b, u, v), it is also clear that $n = p(a, b, u, v)$ for some computable function p. □

The following theorem will be superseded by Theorem 2.32 (via Lemma 2.13). However, in order to become more acquainted with a useful proof technique, we include an outline of a direct proof here, different from the original proof given by Montagna [25].

Theorem 2.14 ([25])**.** *Every u.f.p. ceer is universal.*

Proof. Let S be u.f.p. with totalizer f. As usual, we are assuming that S is nontrivial, and thus fix a and b with $a \not S\ b$. In order to show that S is universal, we fix an arbitrary ceer R with $0 \not R\ 1$ and demonstrate that $R \leqslant S$. By the Fixed Point Theorem, we assume that we control the partial computable function φ_e. Define the computable sequence y_i by $y_0 = a$, $y_1 = b$ and $y_i = f(\{y_j \mid j < i\}, e, i)$ for each $i \geqslant 2$. By our choice of whether to make $\varphi_e(i)$ converge, we can control whether y_i and y_j are S-equivalent. We show that $R \leqslant S$ via the function $i \mapsto y_i$. We will ensure in the construction that if a number k is the least number in its R-equivalence class at stage s, then $\varphi_{e,s}(k)\uparrow$.

When we witness at an odd stage $s + 1$ (we assume that R and S are approximated as in the proof of Theorem 2.6) that $i\ R\ j$ for $i \neq j$ with i and j being least in their respective R_s-equivalence classes, and, say $i < j$, then we define $\varphi_{e,s}(j) = y_i$. As $f(\_, e, \_)$ is a totalizer of φ_e, it must occur that y_j becomes S-equivalent to y_i. Notice that i becomes the least number in the combined R_{s+1}-equivalence class and, as promised, that we have not yet caused $\varphi_e(i)$ to converge.

At even stages $s + 1$, we ensure that S does not collapse y_i to y_j unless already $i\ R_s\ j$. We do this by threat of forcing a contradiction via the Fixed Point Theorem. Suppose i and j are the least numbers in their R-equivalence classes at an even stage $s+1$, and the S-classes of y_i and y_j become S-equivalent at $s + 1$. Thus $\varphi_{e,s}(i)\uparrow$ and similarly $\varphi_{e,s}(j)\uparrow$. We then will cause $\varphi_{e,s+1}(i)\downarrow = a$ and $\varphi_{e,s+1}(j)\downarrow = b$, thus forcing that $a\ S\ y_i\ S\ y_j\ R\ b$ contradicting that $a \not S\ b$. Simply the threat of this action ensures that at no stage will it happen that $y_i\ S\ y_j$ but $i \not R\ j$. □

Definition 2.15. An *extended diagonal function* for an equivalence relation R is a computable function d such that for every finite set D, we have that $x \not R\, d(D)$ for every $x \in D$, i.e. $d(D) \notin [D]_R$.

We observe:

Corollary 2.16 ([7])**.** *Every u.f.p. ceer R with a diagonal function has an extended diagonal function.*

Proof. Let R be a u.f.p. ceer, with a diagonal function d, and let $f(D, e, x)$ be a totalizer witnessing that R is u.f.p. By the Recursion Theorem with parameters, let $n(D)$ be a computable function such that

$$\varphi_{n(D)}(x) = d(f(D, n(D), x)) :$$

then $g(D) = d(f(D, n(D), 0))$ is total, and $g(D) \notin [D]_R$: if $d(f(D, n(D), 0)) \in [D]_R$ then $\varphi_{n(D)}(0) \in [D]_R$, hence $f(D, n(D), 0)\ R\ d(f(D, n(D), 0))$, contradiction. □

2.6. **e-complete ceers.** The ceer $\sim_{PA}$ has an interesting additional property which is captured by the following definition, due to Montagna [25], and later independently rediscovered by Lachlan [22]. The equivalence relations described by this definition were called *uniformly finitely m-complete* by Montagna [25], and *extension complete* (or, simply, *e-complete*) by Lachlan [22]. We adopt here Lachlan's terminology.

Definition 2.17. ([25],[22]) An equivalence relation S is *e-complete* if for every ceer R and every pair of m-tuples $(a_1, \ldots, a_m)$, $(b_1, \ldots, b_m)$ such that the assignment $a_i \mapsto b_i$ induces a partial monomorphism from R to S, one can extend the assignment (uniformly from the two tuples and an index for R) to a computable function inducing a monomorphism. (Notice that uniformity extends also to the case in which the assignment does not provide a partial monomorphism.)

Corollary 2.18. *Every e-complete ceer is universal.*

Proof. Obvious. □

2.7. **e-complete ceers are isomorphic.** Finally we show that alle e-complete ceers are isomorphic.

Theorem 2.19 ([25],[22])**.** *The e-complete ceers are all isomorphic with each other.*

Proof. Let R, S be e-complete ceers. To show isomorphism, one uses a straightforward back-and-forth argument. We define an assignment $a_s \mapsto b_s$ at stages as follows.

Step 0. Do nothing.

Step $2s + 1$. Assume that we have already defined (a_i, b_i) for all $i \leqslant 2s - 1$, so that $a_i\ R\ a_j$ if and only if $b_i\ S\ b_j$. Let a_{2s} be the least such that $a_{2s} \notin \{a_i : i \leqslant 2s - 1\}$. By the uniform extension property due to the fact that S is e-complete, we can uniformly extend the finite assignment which has been defined so far, to a monomorphism, induced, say, by the computable function f. Then, let $b_{2s} = f(a_{2s})$.

Step $2s+2$. Assume that we have already defined (a_i, b_i) for all $i \leqslant 2s$, so that $a_i \ R \ a_j$ if and only if $b_i \ S \ b_j$. Let b_{2s+1} be the least such that $b_{2s+1} \notin \{b_i : i \leqslant 2s\}$. By the uniform extension property due to the fact that R is e-complete, we can uniformly extend the finite assignment which has been defined so far, to a monomorphism, induced say, by the computable function g. Then, let $a_{2s+1} = g(b_{2s+1})$. □

Theorem 2.20 ([25], [7])**.** *A ceer R is e-complete if and only if R is u.f.p. and R has a diagonal function.*

Proof. Given the fact that all e-complete ceers are isomorphic, and that there exists a ceer that is u.f.p. and with a diagonal function (namely, $\sim_{PA}$), it is enough to show that every u.f.p. ceer R with a total diagonal function, is e-complete as the property of being u.f.p. and having a total diagonal function is invariant under computable isomorphisms. Now, by Corollary 2.16 this amounts to show that every u.f.p. ceer R with an extended diagonal function, is e-complete.

To see this, let us see that if S is any ceer, and $a_i \mapsto y_i$, for $i < m$ induces a monomorphism from S to R, then this assignment can be extended to a monomorphism. We can assume that $a_i = i$. We argue almost as in the proof of universality of u.f.p. ceers. We suppose to control, by the Recursion Theorem, a partial computable function φ_e, and define (for $i \geqslant m$),

$$y_i = f(\{y_j : j < i\} \cup \{d(\{y_j : j < i\})\}, e, i)$$

where f is an R-totalizer, and by Corollary 2.16, d is an extended diagonal function. A distinguishing difference with the proof of Theorem 2.14 is how we prevent that $y_i \ R \ y_j$ before we see that $i \ S \ j$. If we see this happen at some stage, we simply define (assume $i < j$, and j is least in its S-equivalence class at the stage, so that we assume by induction that $\varphi_e(j)$ is undefined at the given stage) $\varphi_e(j) = d(\{y_k : k < j\})$. Thus, as $\varphi_e(j) \downarrow \in \{y_k : k < i\} \cup \{d(\{y_k : k < j\})\}$,

$$d(\{y_k : k < j\}) = \varphi_e(j) \ R \ f(\{y_k : k < j\} \cup \{d(\{y_k : k < j\})\}, e, j) = y_j$$

giving a contradiction as now $d(\{y_k : k < j\}) \ R \ y_i$. □

Lemma 2.21. *The ceer $\sim_{PA}$ is e-complete.*

Proof. By Corollary 2.11 and the presence of a diagonal function. □

Notice that Peano Arithmetic provides examples of each one of the fundamental isomorphism types we have seen so far: in fact $\sim_{PA}$ is e-complete, whereas for instance $\sim_1$ is precomplete.

In contrast with the extension property for precomplete ceers pointed out in Theorem 2.9, and with the purpose of better distinguishing precomplete ceers from e-complete ceers, Shavrukov [29] shows

Theorem 2.22 ([29])**.** *For every e-complete E, there is a partial non-onto monomorphism that can not be extended to an endomorphism of E.*

Proof. Let E be e-complete, and P precomplete. We use Greek letters to denote morphisms. Let $\kappa : Id \longrightarrow P$, $\lambda : P \longrightarrow E$ be monomorphisms, and let $\eta = \lambda \circ \kappa$. Let $\theta : Id \longrightarrow E$ be given, induced by

$$t(x) = d(\{t(0), t(1), \ldots, t(x-1), x\})$$

where d is an extended diagonal function for E.

We claim that there is no endomorphism μ of E extending $\theta \circ \eta^{-1}$. Otherwise, if μ is such, let h be a computable function inducing $\mu \circ \lambda$. Then

$$\delta(x) = \text{first } y.[h(y)\, E\, t(h(x))]$$

is total and diagonal for P. For totality, notice that since μ extends $\theta \circ \eta^{-1}$, we have

$$\theta = \mu \circ \eta = \mu \circ \lambda \circ \kappa,$$

thus $\text{range}(t) \subseteq [\text{range}(h)]_E$. The remaining claim, i.e., $\delta(x) \not P x$ follows easily. Indeed, given x, first notice that $t(x) \not E x$ by definition of t; on the other hand, $h(\delta(x))\ E\ t(h(x))$, by definition of δ; but if $x\ E\ \delta(x)$ then also $h(x)\ E\ h(\delta(x))$, as h induces a morphism: contradiction. □

Corollary 2.23. *If R is a u.f.p. ceer with a diagonal function then R has an automorphism without fixed points.*

Proof. Trivial since in this case R isomorphic to $\sim_{PA}$, for which $\neg$ induces an automorphism without fixed points. □

About fixed points of endomorphisms, Shavrukov [29] has shown that every u.f.p. ceer possesses endomorphisms with as many fixed points as we wish:

Theorem 2.24 ([29])**.** *Let E be a u.f.p. ceer, and A a nonempty E-closed c.e. set. Then there is a computable function h, inducing and endomorphism of E such that $A = \{x : x\ E\ h(x)\}$.*

Proof. We may suppose without loss of generality that $0 \in A$. We define a computable function $h(i) = y_i$ that induces an endomorphism whose fixed points are exactly the equivalence classes of elements of A. In the rest of the proof, we say that a number is a fixed point if its equivalence class is a fixed point for the endomorphism induced by h.

The number y_i will be of the form

$$y_i = f(\{y_j : j < i\} \cup \{0, i\}, e, i),$$

where f is an E-totalizer, and e is an index such that by the Recursion Theorem we control φ_e. Since (by Lemma 2.13) the equivalence classes of E are infinite, we may suppose $f(D, z, i) \notin \{0, i\}$ for every D, z, i, and thus $y_i \neq 0, i$ for every i. At each stage, if i is least in its equivalence class and we have not previously defined $\varphi_e(i)$ to be 0 or i, then assume by induction that $\varphi_e(i)$ is undefined.

We use approximations $\{E_s\}$ to E as in Lemma 1.4, with the additional feature that if $E_{s+1} \smallsetminus E_s \neq \varnothing$ then $s + 1 = 3t + 1$ for some t; and we use a computable approximation $\{A_s\}$ to A such that if $A_{s+1} \smallsetminus A_s \neq \varnothing$ then $s + 1 = 3t$ for some t, and $A_{s+1} \smallsetminus A_s \neq \varnothing$ is at most a singleton, and the approximation starts from the empty set.

The construction is by stages: at stages of the form $3t$ we make sure that all numbers in A are fixed points; at stages $3t + 1$ we make sure that h eventually induces an endomorphism; at stages $3t + 2$ we make sure that all fixed points are in A. At stage $s > 0$ we act as follows:

Stage $s = 3t$. If $i \in A_s \smallsetminus A_{s-1}$, and $\varphi_e(i)$ is still undefined, define $\varphi_e(i) = i$.

Stage $s = 3t + 1$. If $i < j$ were least in their equivalence classes at stage $s - 1$ and they E-collapse at stage s, then we act as follows: if $\varphi_e(j)$ is still undefined, define $\varphi_e(j) = y_i$; if $\varphi_e(j)$ has been already defined (with $\varphi_e(j) \in \{0, j\}$), and if $\varphi_e(i)$ is still undefined, define $\varphi_e(i) = i$.

Stage $s = 3t + 2$. If i and y_i have become E-equivalent at the previous stage, and $\varphi_e(i)$ is still undefined, then define $\varphi_e(i) = 0$.

Notice that our action at each stage preserves the inductive assumption that $\varphi_e(i)$ is still undefined if i is least in its equivalence class, unless we define $\varphi_e(i) \in \{0, i\}$. When we define $\varphi_e(i)$ we make $\varphi_e(i) \in \{y_j : j < i\} \cup \{0, i\}$ so that $\varphi_e(i)\ E\ y_i$ as f is a totalizer for E. We further observe that if $\varphi_e(i)$ is defined and $\varphi_e(i) \in \{0, i\}$, then $i \in A$ and i is a fixed point: this is trivial if $\varphi_e(i) = 0$; if $\varphi_e(i) = i$ then either $\varphi_e(i)$ has been defined at a stage $3t$, in which case the claim is trivial; or it has been defined through the second clause of a stage $3t + 1$. In this latter case, as f is a totalizer, our definition $\varphi_e(i) = i$ makes $i\ E\ y_i$; but $\varphi_e(j) \in \{0, j\}$ (where $j\ E\ i$ is the other number of the pair on which we act at the stage) and thus by induction on the stage we may assume that $j \in A$ which implies $i \in A$ as A is E-closed.

Let us now show that h induces a morphism. Assume that $i\ E\ j$, with $i < j$. Using that f is a totalizer, we get $y_i\ E\ y_j$ if we act on i, j at the stage s at which they are E-collapsed (we may again assume that they were least in their E-equivalence classes immediately before E-collapse); if we do not act on i, j , then both $\varphi_e(j)$ and $\varphi_e(i)$ have been already defined, and $\varphi_e(j) \in \{0, j\}, \varphi_e(i) \in \{0, i\}$, which, as argued above, gives $y_j\ E\ j\ E\ i\ E\ y_i$.

Finally we show that $j \in A$ if and only if j is a fixed point. If we ever define $\varphi_e(j) \in \{0, j\}$, then we have already seen that $j \in A$ and j is a fixed point. Suppose towards a contradiction that j is least so that $j \in A$ but j is not a fixed point, or vice versa. So suppose that $j \in A$ ($j\ E\ y_i$, respectively) but we never get to define $\varphi_e(j) = j$ ($\varphi_e(j) = 0$, respectively). This happens only if at the appropriate stage $3t$ ($3t + 2$, respectively), when we would like to act correspondingly, we see that $\varphi_e(j)$ has already been defined through the first clause of some step $3t + 1$, say $\varphi_e(j) = y_i$ for some $i < j$ with $i\ E\ j$. Since $i\ E\ j$, we have that $i \in A$ if and only if $j \in A$ and i is a fixed point if and only if j is a fixed point. So, $i < j$ contradicts the minimality of j. □

2.8. **Uniformly effectively inseparable ceers.** The main result of this section shows that every u.e.i. ceer is universal. To this end, we introduce a class of ceers, the *strongly uniformly m-complete (strongly u.m.c.) ceers*, and show, for any ceer R,

$$R \text{ u.e.i} \Rightarrow R \text{ strongly u.m.c.} \Rightarrow R \text{ universal.}$$

Here is the definition of a strongly u.m.c. ceer. It is a strengthening of the definition of a uniformly m-complete ceer given by Bernardi and Sorbi [8]. 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 $a_0 \not S\ a_1$ and $b_0 \not R\ b_1$, there exists a computable function extending the assignment and reducing S to R. It is shown in [1, Proposition 3.13] that not every u.m.c. is strongly u.m.c.

Definition 2.25. 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 $a_0 \not S\ a_1$ and $b_0 \not R\ b_1$. (Note that the uniformity extends also to the cases $a_0\ S\ a_1$ or $b_0\ R\ b_1$; however, then no claim is made as to f reducing S to R.)

It immediately follows:

Corollary 2.26. *Every strongly u.m.c. ceer is universal.*

Proof. Straightforward. □

Now we aim to prove that

$$R \text{ u.e.i} \Rightarrow R \text{ strongly u.m.c..}$$

For this we introduce yet another class of ceers, the weakly u.f.p. ceers, and show

$$R \text{ u.e.i} \Rightarrow R \text{ is weakly u.f.p.} \Rightarrow R \text{ strongly u.m.c.}.$$

Definition 2.27. We say that a nontrivial ceer R is *weakly u.f.p.* if there exists a total computable function $f(D, e, x)$ such that for every finite set D, where $i \not\mathrel{R} j$ for every $i, j \in D$, and every e, x,

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

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 2.28. *Every u.f.p. ceer is weakly u.f.p.*

Proof. Immediate. □

A restriction of the definition is the following:

Definition 2.29. We call a nontrivial ceer *weakly n-u.f.p.* if in the definition for weakly u.f.p., we replace "finite set D" with "finite set D where $|D| \leqslant n$".

Lemma 2.30. *Each u.e.i. ceer is weakly u.f.p.*

Proof. Let R be a u.e.i. ceer. We first prove that R is weakly 2-u.f.p. To this end, assume that R is u.e.i. via the uniform productive function $p(a, b, u, v)$ as in Definition 1.5(3). We argue that R is weakly 2-u.f.p. Given any $a \neq b$, and e, we uniformly build a function $f(x) = f(\{a, b\}, e, x)$ 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. 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

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

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

Now assume that $a \not\mathrel{R} b$, 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 $f(x) \not\mathrel{R} a$ then $W_{a_x} \cap W_{b_x} = \varnothing$ and $p(a_x, b_x) \in W_{a_x} \cup W_{b_x}$, which contradicts p being a productive function.

Next, we show that if R is weakly 2-u.f.p. then R is weakly u.f.p. To this end, let f_i be a computable function witnessing that R is weakly i-u.f.p., for $2 \leqslant i \leqslant n$. 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 $i \leqslant 0$ then $f_{n+1}(D, e, x)$ outputs 0 for every x; if $1 \leqslant i \leqslant n$ then $f_{n+1}(D, e, x) = f_i(D, e, x)$ for every x. We assume now $D = \{d_0, \ldots, d_n\}$. By the Double Recursion Theorem, assume that we build φ_a and φ_b for some a, b. Let $E_x = \{f_n(D \smallsetminus \{d_n\}, a, x), d_n\}$, and $f_{n+1}(D, e, x) = f_2(E_x, b, x)$.

Here is how we compute $\varphi_a(x)$ and $\varphi_b(x)$. Initially both values are undefined. Step by step, we see which of the following cases happens first:

- $\varphi_e(x)\downarrow R\ d_n$: define $\varphi_b(x) = d_n$.

- $\varphi_e(x)\downarrow R\ d_i$ for some $i < n$: define $\varphi_b(x) = f_n(D \smallsetminus \{d_n\}, a, x)$ and $\varphi_a(x) = \varphi_e(x)$.
- $f_n(D \smallsetminus \{d_n\}, a, x)\ R\ d_n$: 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.

In order to see 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 $i \leqslant n$. First we claim that $f_n(D \smallsetminus \{d_n\}, a, x) \not R\ d_n$: otherwise, 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 $d_n\ R\ f_n(D \smallsetminus \{d_n\}, a, x)\ R\ d_i$, 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 $f_{n+1}(D, e, x)\ R\ \varphi_e(x)$. □

In the proof of Lemma 2.31 below we will use a computable infinite sequence of fixed points. This means that we wish to have an infinite sequence $\{e_i\}_{i\in\omega}$ so that we control each φ_{e_i} simultaneously. This can be done by the usual fixed point theorem, which gives us a single φ_e which we control. We simply let e_i be an index so that $\varphi_{e_i}(x) = \varphi_e(\langle i, x\rangle)$. Then by constructing the single function φ_e which we control, we simultaneously construct the infinite sequence of functions $\{\varphi_{e_i}\}_{i\in\omega}$. Of course, given the single index e we can computably list the infinite sequence $\{e_i\}_{i\in\omega}$.

Lemma 2.31. *Each weakly u.f.p. ceer is strongly u.m.c.*

Proof. We only sketch the proof, which is rather difficult. For a full and rigorous proof see [1].

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 $(0, 1) \mapsto (a_0, a_1)$ can be extended, uniformly in a_0, a_1, to a total computable function inducing a reduction from S to R, provided that $0 \not S\ 1$ and $a_0 \not R\ a_1$. (Uniformity extends also to the cases in which $0\ S\ 1$, or $a_0\ R\ a_1$.)

Notice that it is no loss of generality considering an assignment $(0, 1) \mapsto (a_0, a_1)$, instead of $(a'_0, a'_1) \mapsto (a_0, a_1)$: indeed, given S and $(a'_0, a'_1) \mapsto (a_0, a_1)$, one can consider the ceer S' and the new assignment $(0, 1) \mapsto (a_0, a_1)$, where we have picked a computable permutation g of ω with $g(i) = a'_i$, for all $i \leqslant 1$, and we have defined $x\ S'\ y$ if and only if $g(x)\ S\ g(y)$. 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.

Our goal (under the assumption that $0 \not S\ 1$, and $a_0 \not R\ a_1$) is to extend this assignment to a total computable function yielding a reduction, by specifying a computable sequence of points $(y_2, y_3, \dots)$ (we let $y_0 = a_0$, $y_1 = a_1$) where for every pair $i < k$ such that $k > 1$, we can force y_k to R-collapse to y_i, i.e., to have $y_k\ R\ y_i$. The idea would be of course to mimic the proof that every u.f.p. is universal, and just define $y_i = f(\{y_j : j < i\}, e, i)$, where e is some index that we control by the Recursion Theorem. But, totalizers for R work only if the elements of D are pairwise R-inequivalent. Thus, if we defined $y_i = f(\{y_j : j < i\}, e, i)$, and then we see $0\ S\ 2$ so we force $y_2\ R\ y_0$ by making $\varphi_e(2)\downarrow = 0$, then we would no longer be able to cause y_k to collapse to y_i $(i < k)$ for any $k > 2$, because the set $\{y_j : j < k\}$ is no longer comprised of pairwise R-inequivalent elements. So the proof and the definition of y_i become more complicated.

By the Recursion Theorem we assume that we control φ_{e_i} for a computable sequence $\{e_i\}_{i\in\omega}$ of indices.

We define computable arrays $\{x_i^k, y_n\}_{i,k,n\in\omega}$, in the following way:

- $x_0^k = f(\{a_0, a_1\}, e_1, k)$;
- $y_k = f(\{x_1^k, \ldots, x_{k-1}^k\} \cup \{x_0^{2k}\}, e_{2k}, 0)$;
- $x_i^k = f(\{y_i, x_0^{2i+1}\}, e_{2i+1}, k)$ (for $i > 0$).

There are three possible problems which could occur, and make it impossible to use the properties of the totalizer: we call them *Problem a1*, *Problem a2*, *Problem b*, abbreviated as P_{a1}, P_{a2}, P_b respectively:

P_{a1}: we will get $x_0^{2k}\ R\ x_i^k$, for some $1 \leqslant i \leqslant k-1$;
P_{a2}: we will get $x_r^k\ R\ x_j^k$, for some distinct r, j, $1 \leqslant r, j \leqslant k-1$;
P_b: we will get $y_i\ R\ x_0^{2k+1}$.

Even if we define $\varphi_{e_{2k}}(0) \in \{x_1^k, \ldots, x_{k-1}^k\} \cup \{x_0^{2k}\}$, P_{a1} and P_{a2} make it impossible to guarantee that $\varphi_{e_{2k}}(0)\ R\ y_k$, as the finite set in the totalizer contains R-equivalent elements. Likewise, P_b makes it impossible to guarantee that $\varphi_{e_{2i+1}}(k)\ R\ x_i^k$ even if we define $\varphi_{e_{2i+1}}(k) \in \{y_i, x_0^{2i+1}\}$.

Identifications. We now explain how to get $y_i\ R\ y_k$, when we see $i\ S\ k$. Suppose we want to R-collapse y_k to y_i, with $i < k$, because we see $i\ S\ k$; we may also assume that i and k are least in their current S-equivalence classes, and for all $j', j < k$, we currently have $j'\ S\ j$ if and only if $y_{j'}\ R\ y_j$:

(1) if $i \leqslant 1$, then *identify* y_k with x_0^{2k} by defining $\varphi_{e_{2k}}(0) = x_0^{2k}$: if all the elements in the set $\{x_1^k, \ldots, x_{k-1}^k\} \cup \{x_0^{2k}\}$ are pairwise R-non-equivalent, then from the totalizer we get

$$x_0^{2k} = \varphi_{e_{2k}}(0)\ R\ f(\{x_1^k, \ldots, x_{k-1}^k\} \cup \{x_0^{2k}\}, e_{2k}, 0) = y_k$$

(possible problems for this are of type P_{a1} or P_{a2}: we will see how to prevent them); and then *identify* x_0^{2k} with a_i, by defining $\varphi_{e_1}(2k) = a_i$; as f is a totalizer and problems do not occur, this guarantees $x_0^{2k}\ R\ a_i$;
(2) if $i > 1$, then *identify* y_k with x_i^k by defining $\varphi_{e_{2k}}(0) = x_i^k$ (possible problems are again of type P_{a1} or P_{a2}: we will see how to prevent them) and then *identify* x_i^k with y_i, by defining $\varphi_{e_{2i+1}}(k) = y_i$, (with possible problem P_b: we will see how to prevent it): if problems do not occur, then we achieve the desired R-collapses, using the fact that f is a totalizer.

We summarize the above actions with the following schemes: $a \longrightarrow b$ denotes that we identify a with b; $a \underset{P}{\longrightarrow} b$ denotes that in the identification we encounter a problem of type P. So (1) and (2) above are schematized as

$$(i \leqslant 1) \quad y_k \underset{P\in\{P_{a1},P_{a2}\}}{\longrightarrow} x_0^{2k} \longrightarrow a_i$$
$$(i > 1) \quad y_k \underset{P\in\{P_{a1},P_{a2}\}}{\longrightarrow} x_i^k \underset{P_b}{\longrightarrow} y_i.$$

Problems and their solutions. We indicate how to prevent problems from occurring. For this, we need more dientifications:

P_{a1}: $x_0^{2k}\ R\ x_i^k$. *Solution*:

$$x_i^k \underset{P_b}{\longrightarrow} x_0^{2i+1} \longrightarrow a_0$$
$$x_0^{2k} \longrightarrow a_1;$$

The actions for the *identifications* are the following:

$x_i^k \longrightarrow x_0^{2i+1}$: define $\varphi_{e_{2i+1}}(k) = x_0^{2i+1}$;
$x_0^{2i+1} \longrightarrow a_0$: define $\varphi_{e_1}(2i+1) = a_0$;
$x_0^{2k} \longrightarrow a_1$: define $\varphi_{e_1}(2k) = a_1$;

P_{a2}: $x_r^k\ R\ x_j^k$. *Solution*:

$$x_r^k \underset{P_b}{\longrightarrow} x_0^{2r+1} \longrightarrow a_0$$
$$x_j^k \underset{P_b}{\longrightarrow} x_0^{2j+1} \longrightarrow a_1;$$

The actions for the *identifications* are the following

$x_r^k \longrightarrow x_0^{2r+1}$: define $\varphi_{e_{2r+1}}(k) = x_0^{2r+1}$;
$x_0^{2r+1} \longrightarrow a_0$: define $\varphi_{e_1}(2r+1) = a_0$;
$x_j^k \longrightarrow x_0^{2j+1}$: define $\varphi_{e_{2j+1}}(k) = x_0^{2j+1}$;
$x_0^{2j+1} \longrightarrow a_1$: define $\varphi_{e_1}(2j+1) = a_1$.

P_b: $y_i\ R\ x_0^{2i+1}$. *Solution*:

$$y_i \underset{P\in\{P_{a1},P_{a2}\}}{\longrightarrow} x_0^{2i} \longrightarrow a_0$$
$$x_0^{2i+1} \longrightarrow a_1.$$

The actions for the *identifications* are the following

$y_i \longrightarrow x_0^{2i+1}$: define $\varphi_{e_{2i}}(0) = x_0^{2i}$;
$x_0^{2i} \longrightarrow a_0$: define $\varphi_{e_1}(2i) = a_0$;
$x_0^{2i+1} \longrightarrow a_1$: define $\varphi_{e_1}(2i+1) = a_1$.

What really happens? When we see some collapse which prevents us from using the properties of the totalizer, we *threaten* to start two parallel lines of successive identifications which propagate R and end respectively with a_0 and a_1: since the two starting elements of these two lines are (against our wishes) R-equivalent, we could conclude that $a_0\ R\ a_1$, a contradiction. So, in fact, the unwanted collapse does not happen. We call *threatening action*, any identification in these virtual lines of identifications. Therefore we can conclude that we are able to R-collapse y_k to y_i, when we see that $i\ S\ k$. The difficult part of the verification consists in showing that we are always able to identify when we want to do so, i.e. the relevant values $\varphi_e(j)$ of the involved partial computable functions are still undefined. To show this, one can use the following facts:

i. we are always able to make new definitions when we work with a k least in its current S-equivalence class: we assume by induction at the beginning of the stage, that the values $\varphi_{e_1}(k)$, $\varphi_{e_{2k}}(0)$, $\varphi_{e_{2i+1}}(k)$ are still undefined;
ii. problems alternate, i.e. the sequence of problems is such that P_{a_1} or P_{a_2} is either the last problem to occur, or it is followed by P_b; and P_b is either the last problem to occur, or it is followed by P_{a_1} or P_{a_2};

iii. when we face problem P_b, we introduce a y_j with smaller j; notice that j need not be the least in its current equivalence class: in this case we continue with $y_{j'}$ with j' least in its current S-equivalence class (as, $j', j < k$, we currently have that $j'\ S\ j$ if and only if $y_{j'}\ R\ y_j$);
iv. no threatening action does in fact take place, so no problem does in fact take place, so no new definitions of values $\varphi_e(j)$ involved in threatening actions do in fact take place; hence the inductive assumptions relative to values of various φ_e being undefined at the beginning of the current stage is preserved;
v. no new definitions are in fact ever made for the elements corresponding to y_1 in the array, so we are always virtually able, once we bump into y_1 in a threatening action, to end the threatening line with a_0 or a_1 as we wish.

With the same trick, i.e. of threatening to force a contradiction via suitable identifications, we argue that there is never any unwanted R-collapse between some y_i and y_k, in fact we never see y_k to R-collapse to y_i, before we see k and i S-collapse. □

It is now possible to close the circle, and show:

Theorem 2.32. *The following properties are equivalent for ceers:*

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

Proof. For the proof, we just need the following lemma. □

Lemma 2.33. *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 in which U and V are the only two nontrivial equivalence classes. 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., one can uniformly extend this assignment to a computable function $f_{a,b}$. If $[a]_R \cap [b]_R = \varnothing$, 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., [28]). The fact that R is u.e.i. follows from the uniformity in this argument. □

Remark 2.34. Uniformity plays a crucial role in the proof of universality for the u.e.i. ceers. Recent work has in fact shown ([1]) that there exist ceers yielding a partition of ω into effectively inseparable equivalence classes but they are not u.e.i. In fact the index set of the u.e.i. ceers is Σ^0_3-complete ([1]), but the index set of the effectively inseparable ceers is Π^0_4-complete ([2]).

2.9. **Summarizing.** Corollary 2.35 below subsumes all universality results known in the literature, including: every creative set is m-complete (Myhill [26]); every pair of effectively inseparable sets is m-complete (Smullyan [30]); all creative sequences are m-complete (Cleave [9]).

Corollary 2.35. *Every u.e.i. ceer is universal.*

Proof. Immediate by Theorem 2.32, 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 $a'_0\ \not S\ a'_1$ and $a_0\ \not R\ a_1$, and extend it to a full reduction. □

Corollary 2.36. *A ceer R is universal if and only if there exists a u.e.i. ceer S with $S \leqslant R$.*

Proof. If R is universal and S is u.e.i., then trivially $S \leqslant R$. Conversely, if S is u.e.i. and $S \leqslant R$, then R is universal, since so is S, by Corollary 2.35. □

Corollary 2.37. *A ceer R is universal if and only if there is a c.e. set $X \subseteq \omega$ which is R-closed (i.e. so that $x\ R\ y$ and $x \in X$ implies $y \in X$) and X^R is u.e.i. where $X^R = \{(i,j) : x_i\ R\ x_j\}$ for a computable enumeration $X = \{x_i : i \in \omega\}$.*

Proof. If R is universal, then let S be u.e.i. with $S \leqslant R$ via a reduction f. Then let $X = \{x : (\exists y, c)[x\ R\ y \,\&\, y = f(c)]\}$. Then X is chosen to have the property that $x\ R\ y$ and $x \in X$ implies $y \in X$. We now show that X^R is u.e.i. Given any two numbers i, j, let c_i, c_j be so that $f(c_i)\ R\ x_i$ and $f(c_j)\ R\ x_j$. For any r.e. set U, let U_0 be the set $\{x : (\exists y)[f(x)\ R\ y \,\&\, y \in U]\}$. If p is a uniform productive function for S, then the function $P(i,j,U,V) = i$, where i is so that $x_i = f(p(c_i, c_j, U_0, V_0))$, is a uniform productive function for X^R.

Conversely, it is clear that $X^R \leqslant R$ via the function $f(i) = x_i$. Thus if X^R is u.e.i., it is universal, and thus R is universal. □

3. u.f.p. ceers which are neither precomplete nor e-complete

Precomplete ceers and e-complete ceers are not however the only ceers in the class of u.f.p. ceers.

Definition 3.1. [4] An equivalence relation E is *weakly precomplete* if there exists a partial computable function fix such that, for all e,

$$\varphi_e \text{ total} \Rightarrow [\text{fix}(e)\!\downarrow\ \,\&\, \varphi_e(\text{fix}(e))\ E\ \text{fix}(e)].$$

Clearly, a ceer E is weakly precomplete if and only if E has no computable diagonal function.

The following theorem and its corollary showing that there are infinitely many non-isomorphic u.f.p. ceers are taken from [5].

Theorem 3.2. *If E is a ceer, such that E has an extended diagonal function, then there exist infinitely many ceers $\{E_i : i \in \omega\}$ such that, for every i, j,*

$$E \subseteq E_i \,\&\, [i \neq j \Rightarrow E_i \not\simeq E_j],$$

where $\simeq$ denotes isomorphism.

Proof. Let E be a given ceer, such that E has an extended diagonal function d.

We want to construct a countable set $\{E_i : i \in \omega\}$ of ceers such that for every i, $E \subseteq E_i$, satisfying the following requirement for each i, j, k, with $i \neq j$,

$$P_{i,j,k} : \varphi_k \text{ is total} \Rightarrow \varphi_k \text{ does not induce an isomorphism from } E_i \text{ onto } E_j.$$

Satisfaction of all requirements implies our claim, as for every isomorphism there is a total computable function inducing it.

We outline the strategy to meet $P_{i,j,k}$ in isolation, which is of course implemented at certain stages s: hence E_i and E_j have to be understood as their approximations E_i^s and E_j^s, respectively, and in particular at each such stage, $[a_0]_{E_i}$ is a finite set:

(1) choose a witness b_0 using the extended diagonal function to be E-inequivalent to every number mentioned so far;
(2) wait for a number a_0 such that $\varphi_k(a_0)\downarrow E_j\ b_0$;
(3) let $a_1 = d([a_0]_{E_i})$, and wait for $\varphi_k(a_1)\downarrow$;
(4) if, say, $\varphi_k(a_1) = b_1$ then E_j-collapse b_0 and b_1, and restrain $a_0 \not\!\!E_i\ a_1$.

Outcomes for the strategy to meet $P_{i,j,k}$. Here are the outcomes of the strategy:

(i) if we wait forever at (2), then we meet $P_{i,j,k}$ since φ_k, even if total, does not induce an onto morphism;
(ii) if we wait forever at (3), then we win $P_{i,j,k}$ since φ_k is not total;
(iii) if we act in (4), then we win $P_{i,j,k}$ since φ_k, even if total, does not induce a monomorphism.

The strategies can be combined by a finite priority argument. The critical part of the verification is that since b_0 is always chosen to be E-inequivalent to any number mentioned so far, and since each requirement is re-initialized if a higher-priority requirement acts, any collapse caused by the requirement $R_{i,j,k}$ cannot collapse together the elements a_0 and a_1 of a higher priority requirement. □

Corollary 3.3. *There exist infinitely many weakly precomplete non-isomorphic u.f.p. ceers.*

Proof. Take $E = \sim_{PA}$ in the previous theorem, and use the fact that the u.f.p. ceers coincide with the nontrivial quotients of any e-complete ceer. □

4. Separating u.e.i. ceers from u.f.p. ceers

The u.f.p. ceers are properly contained in the class of u.e.i. ceers, as shown by Andrews and Sorbi [3]:

Theorem 4.1. [3] *There is a u.e.i. ceer which is not u.f.p.*

Proof. See [3]. □

In the same paper they show that in a sense, little is missing for a u.e.i. ceer to be u.f.p.

Theorem 4.2. [3] *If a u.e.i. ceer has an extended diagonal function then it is u.f.p.*

Proof. See [3]. □

The following picture summarizes the inclusion relationships between the classes of universal ceers, which we have introduced so far. The u.m.c. ceers have been defined at the beginning of Subsection 2.8.

All the inclusions shown by the picture are proper, by the above results. Not all universal ceers of course appear in one of these classes. For instance if R is a universal ceer then clearly $R \oplus \mathrm{Id}_1$ is universal but not u.m.c., where $R \oplus \mathrm{Id}_1$ is the ceer which collapses all odd numbers, and $2x\ R \oplus \mathrm{Id}_1\ 2y$ if and only if $x\ R\ y$.

The following result by Nies and Sorbi [27] shows that the class of u.e.i. contains interesting mathematical objects.

Theorem 4.3 ([27])**.** *There is a finitely presented group D such that $=_D$ is a u.e.i. ceer.*

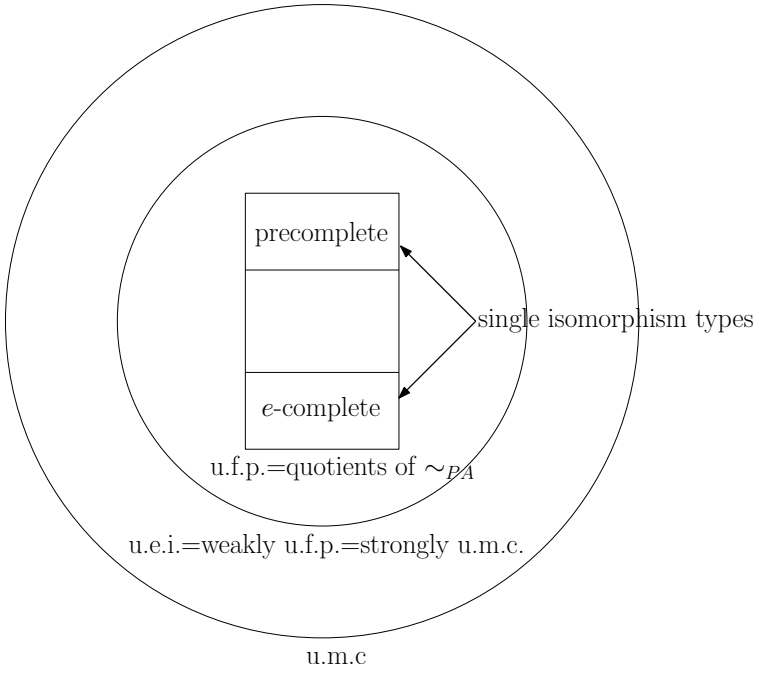

FIGURE 1. Some classes of universal ceers

Proof. See [27]. □

5. A characterization of the universal ceers through a jump operation

In this section, we look at a jump operation on ceers (due to [19]), and show that the universal ceers are exactly the ceers which are fixed points (modulo the equivalence) for this operation.

Definition 5.1. [19] *For any ceer R, we define the jump of R to be the ceer R' so that $x\ R'\ y$ if and only if $x = y$ or $\varphi_x(x)\downarrow$, $\varphi_y(y)\downarrow$, and $\varphi_x(x)\ R\ \varphi_y(y)$.*

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

Lemma 5.2. *The following properties hold:*

(1) $R \leqslant R'$;
(2) $R \leqslant S \Leftrightarrow R' \leqslant S'$;
(3) If R is not universal then R' is not universal.

Proof. (1) For every i, we can effectively find a number x_i so that $\varphi_{x_i}(x_i) = i$. By the Padding Lemma we may assume that the sequence (x_i) is injective. Then the map $i \mapsto x_i$ is a reduction of R to R'.

(2) Suppose $R \leqslant S$ via the function f. Given an index i, we can effectively find an index x_i so that if $\varphi_i(i)\downarrow$, then $\varphi_{x_i}(x_i)\downarrow = f(\varphi_i(i))$: as before we may assume that the sequence (x_i) is injective. Then the map $i \mapsto x_i$ gives a reduction of R' to S'.

Suppose $R' \leqslant S'$ via g. We first claim that for each x, if $\varphi_x(x) \downarrow$ then $\varphi_{g(x)}(g(x)) \downarrow$. Otherwise, we would have that the S'-class of $g(x)$ consists of a single element. But then the R'-class of x would be computable. But this is the set K_r, for $\varphi_x(x) = r$. It is a standard result that the set K_r is a complete c.e. set for any r. Thus we conclude that if $\varphi_x(x) \downarrow$ then $\varphi_{g(x)}(g(x)) \downarrow$. Now, consider the map $i \mapsto y_i$ given by taking x_i so that $\varphi_{x_i}(x_i) = i$ and letting $y_i = \varphi_{g(x_i)}(g(x_i))$. This is well-defined and gives a reduction of R to S.

(3) Suppose R' is universal. Then for any X, we have that $X' \leqslant R'$. Thus, we have that $X \leqslant R$. Thus R is universal as well. □

Note that (2) shows that the jump is an operation on degrees of ceers (where the degree of an equivalence relation is the equivalence class of the relation under the equivalence relation $\equiv$ given by $R \equiv S$ if and only if $R \leqslant S$ and $S \leqslant R$). Also, unlike most things called a jump, we can have $R' \equiv R$, for example if R is universal. Gao and Gerdes posed the question (Problem 10.2 of [19]) of whether there are non-universal ceers R so that $R' \equiv R$.

Theorem 5.3. *For every ceer E, if $E' \leqslant E$ then E is universal.*

Proof. We sketch the proof. The reader interested in details is invited to read the paper [1].

Assume that h is a computable function that reduces E' to E. Let R be any ceer, with computable approximations $\{R_s : s \in \omega\}$ to R and $\{E^s : s \in \omega\}$ to E, as in Theorem 2.6: $\{R_s\}$ may collapse equivalence classes only at odd stages, and $\{E_s\}$ may collapse equivalence classes only at even stages, and they both collapse at most two equivalence class at any stage when they do so. We aim to show that $R \leqslant E$.

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. 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.

But this implies $e_1 \; E' \; e_2$, as follows from the sequence of implications

$$\varphi_{e'}(e') \downarrow = \varphi_{e''}(e'') \downarrow \Rightarrow e' \; E' \; e'' \Rightarrow h(e') \; E \; h(e'') \Rightarrow e_1 \; E' \; e_2.$$

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 were to see an E-collapse of say, $h(e_i)$ and $h(e_j)$, without having $i \; R \; j$, then we would stop the construction leaving certain computations divergent, thus forcing a contradiction (exploiting the fact that if $u \neq v$ and $\varphi_u(u)$ and $\varphi_v(v)$ do

not converge, then $u \not E' v$, and thus $h(u) \not E h(v)$): therefore E cannot cause this collapse before seeing $i \, R \, j$.

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 $\alpha = \langle D, n\rangle$, with $D \neq \varnothing$, will be called a *node*: We sometimes denote the components of a node α by D_α and n_α. As in the proof of Lemma 2.31, we will use an infinite computable sequence of indices $\{e_\alpha : \alpha \text{ node}\}$ which we control.

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
$$\varphi_{e_{\langle\{2\},1\rangle}}(e_{\langle\{2\},1\rangle}) = \varphi_{e_{\langle\{0,1\},1\rangle}}(e_{\langle\{0,1\},1\rangle}) = h(e_{\langle\{0,1,2\},2\rangle}).$$
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 is an instance of what is later called synchronization procedure).

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

We now try to be more formal. We say that a node β is a *parent* at stage s of a node α, if $n_\alpha = n_\beta + 1$, and at s $\varphi_{e_\beta}(e_\beta)$ is defined, 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 in more detail below, in which case we have $\alpha = \langle D_\beta, n_\beta + 1\rangle$ and $\varphi_{e_\beta}(e_\beta) = h(e_\alpha)$.

Given a node α, let T_α be the finite tree (depending on the stage), defined as the smallest set of nodes such that: (1) $\alpha \in T_\alpha$; (2) if $\beta \in T_\alpha$ and γ is a parent of β, then $\gamma \in T_\alpha$.

A node α is *realized at stage s* if D_α is a singleton, or α has a parent: the idea is that if α is realized, but $\varphi_{e_\alpha}(e_\alpha)$ is still undefined, then D_α is an R_s-equivalence class: if at some later stage, R collapses D_α with another block D_β, relative to a similarly already realized β, with $\varphi_{e_\beta}(e_\beta)$ still undefined, then we will define

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

so that $\langle D_\alpha \cup D_\beta, n+1\rangle$ becomes realized. To facilitate the construction and the verification of the two following lemmas, we do this only if $n_\alpha = n_\beta$: for this, we may employ at the stage the *synchronization procedure*: if, say, $n_\alpha < n_\beta$, then for every i with $n_\alpha \leqslant i < n_\beta - 1$, define,

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

The two basic lemmas (which refer to the current stage s, at which computations and approximations are taken), are:

Lemma 5.4. *Let α be a realized node, with $n_\alpha = n$. For every $i \leqslant n$, 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 $\gamma \in T_\alpha$ with $n_\gamma = i - 1$, there is a node $\beta \in T_\alpha$ 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 5.5. *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 $e_\gamma \not\equiv' e_\delta$.*

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 $e_\alpha \not\equiv' e_\beta$. Suppose now that the claim is true of $0 < i \leqslant n$, 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, $e_{\gamma'} \not\equiv' e_{\delta'}$, so $h(e_{\gamma'}) \not\equiv h(e_{\delta'})$, and thus we may conclude that $e_\gamma \not\equiv' e_\delta$. □

Construction. The construction can be sketched as follows:

(1) code R into E through suitable convergent computations, as described in the equation (†): if we see for the first time $i\ R\ j$, then pick α, β so that $\langle\{i\}, 0\rangle \in T_\alpha$, $\langle\{j\}, 0\rangle \in T_\beta$, α, β are realized, and $\varphi_{e_\alpha}(e_\alpha)$ and $\varphi_{e_\beta}(e_\beta)$ are still undefined (via synchronization, we may assume that $n_\alpha = n_\beta$), and define $\varphi_{e_\alpha}(e_\alpha) = \varphi_{e_\beta}(e_\beta) = h(e_{\langle D_\alpha \cup D_\beta, n+1\rangle})$. This is implemented at odd stages, the only ones at which R can collapse. Use Lemma 5.4 to show that the values $g(i) = h(\langle e_{\{i\},0}\rangle)$ and $g(i) = h(\langle e_{\{i\},0}\rangle)$ satisfy that $g(i)\ E\ g(j)$;

(2) prevent bad situations, at any stage, of the form $h(e_{\langle\{a\},0\rangle})\ E\ h(e_{\langle\{b\},0\rangle})$ but $a \not\mathrel{R} b$, by threatening to leave certain computations undefined: this is implemented at even stages, the only ones at which E can collapse. In particular, if we see $h(e_{\langle\{a\},0\rangle})\ E\ h(e_{\langle\{b\},0\rangle})$, but $a \not\mathrel{R} b$, then we pick α, β so that $\langle\{a\}, 0\rangle \in T_\alpha$, $\langle\{b\}, 0\rangle \in T_\beta$, α, β are realized, and $\varphi_{e_\alpha}(e_\alpha)$ and $\varphi_{e_\beta}(e_\beta)$ are still undefined (via synchronization, we may assume that $n_\alpha = n_\beta$). We then stop the construction and ensure that $\varphi_{e_\alpha}(e_\alpha)$ and $\varphi_{e_\beta}(e_\beta)$ remain undefined forever. By Lemma 5.5, this would cause a contradiction. Thus we never see this bad situation.

□

6. Characterizations of some classes of universal ceers

Bernardi and Montagna ([7]) use the notion of a quotient object to characterize u.f.p. ceers and precomplete ceers. Given equivalence relations R, S, we say that R is a *quotient of* S, if there is an onto morphism from S to R.

Lemma 6.1. *Let R, S be ceers with no finite classes. Then R is a quotient of S if and only if there is a ceer $S' \simeq R$ such that $S' \supseteq S'$.*

Proof. Easy: if f induces an onto morphism from S to R, then define $x \ S' \ y$ if and only if $f(x) \ R \ f(y)$. □

Theorem 6.2. [7] *The following hold:*

(1) A ceer R is u.f.p. if and only if R is a nontrivial quotient of $\sim_{PA}$.
(2) A ceer R is precomplete if and only if R is a nontrivial quotient of every universal ceer.

Proof. The two implications from right to left follow from Lemma 1.7.

We now show the implications from left to right. We begin with the first item. Let R be a u.f.p. ceer. Construct an onto morphism $\mu :\sim_{PA} \longrightarrow R$, by defining a computable h by stages. Suppose that $f(D, e, x)$ is a totalizer for R. We assume that by the Recursion Theorem we control the index e. Also, assume that we work with computable approximations $\{R_s\}$ to R, and $\sim_{PA,s}$ to $\sim_{PA}$, as in Lemma 1.4: without loss of generality we may assume that $\sim_{PA,s}$ changes only at odd stages.

At the end of stage s, suppose that we have defined a finite set of pairs $(a_0, b_0), \ldots, (a_{s-1}, b_{s-1})$ approximating a computable function h that we build and that will induce the desired onto morphism. At each stage $s + 1$ we assume by induction that if i is least such that $a_i \in [a_i]_{\sim_{PA,s}}$ then $\varphi_{e,s}(i)$ is still undefined.

Stage 0. Let $\varphi_{e,0}(x)$ be undefined for all x.

Stage $s + 1$ odd. See if there are $i < j$ such that a_i, a_j become $\sim_{PA}$-equivalent. If, so, pick such a pair i, j: we may assume that j is least such that $a_j \in [a_j]_{\sim_{PA,s}}$. Define $\varphi_e(j) = b_i$.

Let now $a_s = \mu\, x.[x \notin \{a_i : i < s\}]$, and let $b_s = f(\{b_i : i < s\}, e, s)$.

Stage $s+1$ even. Let $b_s = \mu\, x.[x \notin \{b_i : i < s\}]$, and let a_s be a number which is not $\sim_{PA}$-equivalent to any number which is already in $\{a_i : i < s\}$. We use here that $\sim_{PA}$ has an extended diagonal function.

At each step the inductive assumption is preserved. It is not difficult to see that the assignment $a_s \mapsto b_s$, defines a computable function h with the desired properties.

We now turn to the second item of the statement. Let R be a precomplete ceer, and S a universal ceer: so there is a computable function f which induces a monomorphism from R to S. We want to show that there is a computable function h that induces an onto morphism from S to R. Suppose that we have already defined $h(i)$ for all $i < n$, and let e_n be a uniformly found index such that

$$\varphi_{e_n}(x) = \begin{cases} h(i) & \text{if } ((\exists i < n)[n \in [i]_S]) \preceq ((\exists y)[n \ S \ f(y)]) \text{ and } i \text{ is first,} \\ y & \text{if } ((\exists y)[n \ S \ f(y)]) \prec ((\exists i < n)[n \in [i]_S]) \text{ and } y \text{ is first,} \\ \uparrow & \text{otherwise;} \end{cases}$$

let $f(e, z)$ be a totalizer for R, and define $h(n) = f(e_n, 0)$. For the verification, let us inductively assume that if $i, j < n$ and $i \ S \ j$ then $h(i) \ R \ h(j)$, and let $i < n$ be such that $i \ S \ n$. We want to show that $h(i) \ R \ h(n)$: by the inductive assumption, we may assume that i is least with this property. Then $\varphi_{e_n}(0)$ is defined: if it is defined through the first clause, then by the totalizer f, $h(n) \ R \ h(i)$; otherwise, let $h(n) = y$ where $f(y) \ S \ n$; but then $\varphi_{e_i}(0)$ is defined, and by minimality

of i, it is defined through the second clause, so that $h(i) = z$ for some z such that $f(z)\ S\ i$. It follows that $f(y)\ S\ f(z)$, and thus $y\ R\ z$, as f induces a monomorphism. Therefore h induces a morphism: it is easy to see that this morphism is also onto. □

6.1. **Extensional formulae of Peano Arithmetic.** In this section we consider ceers defined by extensional formulae of Peano Arithmetic.

Definition 6.3. *Given a formula $F(v)$ in the language of PA, let $\sim_F$ be the ceer*

$$x \sim_F y \Leftrightarrow PA \vdash F(\overline{x}) \leftrightarrow F(\overline{y}).$$

A formula $F(v)$ of PA is *extensional* if for every x, y,

$$x \sim_{PA} y \Rightarrow PA \vdash F(\overline{x}) \leftrightarrow F(\overline{y})$$

Theorem 6.4. *The u.f.p. ceers coincide with the ceers that are computably isomorphic to the ones induced by extensional formulas of PA.*

Proof. If R is given by an extensional formula, then $R \supseteq \sim_{PA}$, thus it is u.f.p. by Theorem 6.2.

Conversely, if R is u.f.p., then $R \simeq S$ for some ceer $S \supseteq \sim_{PA}$, by Theorem 6.2. Then, by Lemma 6.5 below, there is a formula $F(v)$ such that $S = \sim_F$. Since $S \supseteq \sim_{PA}$, F is extensional. □

Lemma 6.5. *For every ceer S there exists a Σ_1 formula $F(v)$, such that*

$$x\ S\ y \Leftrightarrow PA \vdash F(\overline{x}) \leftrightarrow F(\overline{y}).$$

Proof. Let S be a ceer. Since $\sim_1$ is precomplete, there exists a computable function f such that

$$x\ S\ y \Leftrightarrow PA \vdash \rho_{f(x)} \leftrightarrow \rho_{f(y)},$$

and $f(x)$ is the $\ulcorner\urcorner_1$-code of a Σ_1 sentence $\rho_{f(x)}$, for all x. Define $g(x) = \ulcorner\alpha\urcorner$ where $f(x) = \ulcorner\alpha\urcorner_1$, and let $G(u, v)$ a Σ_1 formula representing g. By an argument similar to the one in the proof of Theorem 2.4, it is easy to see that can take $F(u)$ to be $(\exists v)(G(u, v) \wedge T_1(v))$. □

An important example of an extensional formula is the provability predicate $\mathrm{Pr}_{PA}(v)$, a Σ_1 formula representing the set of theorems, and satisfying the Hilbert-Bernays Derivability Conditions.

Lemma 6.6. *Let $F(v)$ be a Σ_n extensional formula such that there exists $q \in \Sigma_n$, for which*

$$\left[\{\ulcorner F(\overline{n})\urcorner : n \in \omega\}\right]_{\sim_{PA}} = \left[\{\ulcorner p\urcorner : \ PA \vdash q \to p\}\right]_{\sim_{PA}}.$$

Then $\sim_F$ is precomplete.

Proof. Let ψ be a partial computable function and let

$$\varphi(x) = \begin{cases} \text{the first } y \text{ such that } \ulcorner F(\overline{y})\urcorner \sim_{PA} x, & \text{if there is any such } y; \\ \uparrow, & \text{otherwise.} \end{cases}$$

Let $\Psi(u, v)$ be a formula that represents ψ in PA, and define

$$\hat{\psi}(x) = \ulcorner \exists v(\Psi(\overline{x}, v) \wedge F(v)) \vee q\urcorner.$$

Clearly, $\hat{\psi}$ is total. Let now $h = \varphi \circ \hat{\psi}$. We claim that also h is total. Indeed, let x be given, and observe that

$$PA \vdash q \to (\exists v(\Psi(\overline{x}, v) \wedge F(v)) \vee q)$$

hence by the hypothesis there exists some z such that

$$\ulcorner(\exists v(\Psi(\overline{x}, v) \wedge F(v)) \vee q)\urcorner \sim_{PA} \ulcorner F(\overline{z})\urcorner.$$

This shows that $\varphi(\hat{\psi}(x))$ is defined, hence h is total.

Notice, that by the hypothesis, for every z, since $F(\overline{z})$ is provably equivalent to some sentence which is implied by q, we have $PA \vdash q \to F(\overline{z})$, and thus

$$\ulcorner F(\overline{z}) \vee q\urcorner \sim_{PA} \ulcorner F(\overline{z})\urcorner. \tag{1}$$

We now claim that h makes ψ total modulo $\sim_F$. Suppose that $\psi(x)\downarrow = y$. Then

$$\hat{\psi}(x) \sim_{PA} \ulcorner F(\overline{y}) \vee q\urcorner \sim_{PA} \ulcorner F(\overline{y})\urcorner$$

(where the last equivalence is justified by (1)). Hence if $\varphi(\hat{\psi}(x)) = z$ with $\ulcorner F(\overline{z})\urcorner \sim_{PA} \hat{\psi}(x)$, then we see that $\ulcorner F(\overline{z})\urcorner \sim_{PA} \ulcorner F(\overline{y})\urcorner$, and thus

$$h(x) = \varphi(\hat{\psi}(x)) \sim_F y.$$

□

Theorem 6.7. $\mathrm{Pr}_{PA}(v)$ *is precomplete.*

Proof. We verify that $\mathrm{Pr}_{PA}(v)$ satisfies the hypotheses of Lemma 6.6 with the sentence $q = \neg\mathrm{Con}_{PA}$ and $n = 1$. As independently proved by Goldfarb and Friedman, see [18], for every Σ_1 sentence p such that $PA \vdash \neg\mathrm{Con}_{PA} \to p$ there is a Σ_1 sentence p' such that $PA \vdash p \leftrightarrow \mathrm{Pr}_{PA}(\overline{\ulcorner p'\urcorner})$. The other inclusion follows from the fact that for every n, $PA \vdash \neg\mathrm{Con}_{PA} \to \mathrm{Pr}_{PA}(\overline{n})$. □

References

[1] U. Andrews, S. Lempp, J. S. Miller, K. M. Ng, L. San Mauro, and A. Sorbi. Universal computably enumerable equivalence relations. *J. Symbolic Logic*, 79(1):60–88, March 2014.

[2] U. Andrews and A. Sorbi. The complexity of index sets of classes of computably enumerable equivalence relations. To appear in J. Symbolic Logic.

[3] U. Andrews and A. Sorbi. Jumps of computably enumerable equivalence relations. in preparation, 2015.

[4] S. Badaev. On weakly precomplete positive equivalences. *Siberian Math. Journal*, 32:321–323, 1991.

[5] S. Badaev and A. Sorbi. Weakly precomplete computably enumerable equivalence relations. *Mat. Log. Quart.*, 62(1-2):111–127, 2016.

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

[7] 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.

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

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

[10] S. B. Cooper. *Computability Theory*. Chapman & Hall/CRC Mathematics, Boca Raton, London, New York, Washington, DC, 2003.

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

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

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

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

[15] E. Fokina, B. Khoussainov, P. Semukhin, and D. Turetsky. Linear orders realized by c.e. equivalence relations. *J. Symbolic Logic*, to appear.

[16] E. B. Fokina, S. D. Friedman, and A. Nies. Equivalence relations rhat are Σ^0_3 complete for computable reducibility - (Extended Abstract). In *WoLLIC*, pages 26–33, 2012.

[17] E.B. Fokina, S.D. Friedman, V. Harizanov, J.F. Knight, C. Mccoy, and A. Montalbán. Isomorphism relations on computable structures. *J. Symbolic Logic*, 77(1):122–132, 2012.
[18] H. Friedman. Proof-theoretic degrees. Unpublished.
[19] S. Gao and P. Gerdes. Computably enumerable equivalence realations. *Studia Logica*, 67:27–59, 2001.
[20] A. Gravuskin, S. Jain, A. Khoussainov, and F. Stephan. Graphs realized by r.e. equivalence relations. To appear in *Ann. Math. Logic*.
[21] A. Gravuskin, A. Khoussainov, and F. Stephan. Reducibilities among equivalence relations induced by recursively enumerable structures. *Theoret. Comput. Sci.*, 612(25):137–152, 2016.
[22] A. H. Lachlan. A note on positive equivalence relations. *Z. Math. Logik Grundlag. Math.*, 33:43–46, 1987.
[23] A. I. Mal'tsev. Sets with complete numberings. *Algebra i Logika*, 2(2):4–29, 1963.
[24] C. F. Miller III. *On Group-theoretic Decision Problems and their Classification*. Annals of Mathematics Studies. Princeton University Press, New Jersey, 1971.
[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 and A. Sorbi. Calibrating word problems of groups via the complexity of equivalence relations. To apper in Math. Structures Comput. Sci., 2016.
[28] H. Rogers, Jr. *Theory of Recursive Functions and Effective Computability*. McGraw-Hill, New York, 1967.
[29] V. Yu. Shavrukov. Remarks on uniformly finitely positive equivalences. *Math. Log. Quart.*, 42:67–82, 1996.
[30] R. Smullyan. *Theory of Formal Systems*. Princeton University Press, Princeton, New Jersey, 1961. Annals of Mathematical Studies Vol 47.
[31] R. I. Soare. *Recursively Enumerable Sets and Degrees*. Perspectives in Mathematical Logic, Omega Series. Springer-Verlag, Heidelberg, 1987.
[32] 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.

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/

Department of Fundamental Mathematics, Al-Farabi Kazakh National University, Almaty, 050040, Kazakhstan

E-mail address: serikzhan.badaev@kaznu.kz

Dipartimento di Ingegneria Informatica e Scienze Matematiche, Università Degli Studi di Siena, I-53100 Siena, Italy

E-mail address: andrea.sorbi@unisi.it

URL: http://www3.diism.unisi.it/~sorbi/