The complexity of index sets of classes of computably enumerable equivalence relations

THE COMPLEXITY OF INDEX SETS OF CLASSES OF COMPUTABLY ENUMERABLE EQUIVALENCE RELATIONS

URI ANDREWS AND ANDREA SORBI

Abstract. Let $\leqslant_c$ be computable reducibility on computably enumerable equivalence relations (or ceers). We show that for every ceer R with infinitely many equivalence classes, the index sets $\{i : R_i \leqslant_c R\}$ (with R non-universal), $\{i : R_i \geqslant_c R\}$, and $\{i : R_i \equiv_c R\}$ are Σ^0_3 complete, whereas in case R has only finitely many equivalence classes, we have that $\{i : R_i \leqslant_c R\}$ is Π^0_2 complete, and $\{i : R_i \geqslant_c R\}$ (with R having at least two distinct equivalence classes) is Σ^0_2 complete. Next, solving an open problem from [1], we prove that the index set of the effectively inseparable ceers is Π^0_4 complete. Finally, we prove that the 1-reducibility pre-ordering on c.e. sets is a Σ^0_3 complete pre-ordering relation, a fact that is used to show that the pre-ordering relation $\leqslant_c$ on ceers is a Σ^0_3 complete pre-ordering relation.

1. Introduction

Given equivalence relations R and S on the set ω of natural numbers, we say that R is *reducible* to S (in symbols: $R \leqslant_c S$), if there exists a computable function f such that

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

Given a class $\mathcal{A}$ of equivalence relations on ω, one says that R is $\mathcal{A}$ *complete*, if $R \in \mathcal{A}$, and $S \leqslant_c R$, for every $S \in \mathcal{A}$. This reducibility, and this notion of completeness, have turned out to be very useful tools for measuring the complexity of equivalence relations naturally arising in mathematics, and, in particular, in computable model theory and in computability theory (where equivalence relations on structures can be viewed as relations on numbers via identification of structures with numbers, thanks to suitable indexings). For instance, Fokina, S. Friedman, Harizanov, Knight, McCoy, and Montalbán [8] show Σ^1_1 completeness of the isomorphism relations for various familiar classes of computable structures, including computable groups, computable torsion abelian groups, computable torsion-free abelian groups, abelian p-groups. On the other hand, Fokina, S. Friedman, and Nies [7] show that other familiar equivalence relations arising from computability are Σ^0_3 complete, including computable isomorphism of c.e. sets. (In Corollary 4.11 we give another proof of this result.) Other interesting mathematical applications of reducibility $\leqslant_c$ appear in [4], [11], [12], [9].

2010 *Mathematics Subject Classification.* 03D25.

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

This work was partially supported by Grant 3952/GF4 of the Science Committee of the Republic of Kazakhstan. Andrews was partially supported by NSF grant DMS-1201338.

Sorbi is a member of INDAM-GNSAGA; he was partially supported by PRIN 2012 "Logica Modelli e Insiemi ".

The reducibility $\leqslant_c$, as well as the notion of $\mathcal{A}$-completeness, can obviously be extended to pre-ordering relations on ω. Ianovski, R. Miller, Ng, and Nies [13] characterize the arithmetical complexity of several pre-orders of interest to computability theory, for instance showing that almost inclusion $\subseteq^*$, and $\leqslant_T$ on c.e. sets, are Σ^0_3 complete.

There is already a non-trivial literature concerning the restriction of $\leqslant_c$ to computably enumerable equivalence relations (abbreviated as ceers): pioneering papers in this regard include (in chronological order) Ershov [5], Bernardi and Sorbi [2], Montagna [15], Lachlan [14], Gao and Gerdes [10], Andrews, Lempp, Miller, Ng, San Mauro, and Sorbi [1]. These papers study Σ^0_1 complete (also called *universal*) ceers, and the degree structure of ceers under $\leqslant_c$. We investigate and classify the arithmetical complexity of some index sets of ceers. Throughout the paper, we refer to some fixed universal computable numbering $\{R_i : i \in \omega\}$ of all ceers (see [1]), where "computable" means that the set $\{\langle i, x, y\rangle : x \mathrel{R_i} y\}$ is c.e., and "universal" means that for every such computable numbering $\{S_i : i \in \omega\}$ of all ceers, there exists a computable function f such that $S_i = R_{f(i)}$, for all i. Extending results in [10], we give complete characterizations in the arithmetical hierarchy of the complexity of the index sets $\{i : R_i \leqslant_c R\}$ (with R non-universal), $\{i : R_i \geqslant_c R\}$, and $\{i : R_i \equiv_c R\}$: if R has infinitely many equivalence class then all these sets are Σ^0_3 complete, whereas if R has only finitely many equivalence classes, we have that $\{i : R_i \leqslant_c R\}$ is Π^0_2 complete, and $\{i : R_i \geqslant_c R\}$ (with R having at least two distinct equivalence classes) is Σ^0_2 complete. Solving a problem in [1], we prove that the index set of the effectively inseparable ceers is Π^0_4 complete.

In the last section of the paper we consider $\leqslant_c$ on pre-ordering relations on ω. The literature regarding the restriction of $\leqslant_c$ (as a reducibility on pre-orders) to computably enumerable pre-orders, includes, among others, the papers Pour El and Kripke [18], Montagna and Sorbi [16], and Ianovski, R. Miller, Ng, and Nies [13]: these papers are mainly dedicated to the investigation of Σ^0_1 complete pre-orders naturally arising in logic. We prove that the pre-ordering relation $\leqslant_c$ on ceers (viewed as a pre-order on their indices) is a Σ^0_3 complete pre-ordering relation. The proof goes by first showing that the 1-reducibility pre-ordering on c.e. sets is a Σ^0_3 complete pre-ordering relation.

1.1. **Background.** The reader is referred to [20] for all computability theoretic notions that are used, but not explicitly introduced, in this paper. For more information on ceers, their structure under $\leqslant_c$, bibliography, and even history, our basic reference is [1]. Given a ceer E, we say that a sequence $\{E_s : s \in \omega\}$ of equivalence relations on ω is a *computable approximation to* E, if the following conditions hold: the set $\{\langle x, y, s\rangle : x \mathrel{E_s} y\}$ is computable; E_0 is the identity equivalence relation; for all s, $E_s \subseteq E_{s+1}$; the equivalence classes of E_s are finite; there exists at most one pair $[x]_{E_s}, [y]_{E_s}$ of equivalence classes, such that $[x]_{E_s} \cap [y]_{E_s} = \varnothing$, but $[x]_{E_{s+1}} = [y]_{E_{s+1}}$ (we say in this case that the equivalence relation E *collapses* x *and* y at stage $s+1$); and finally $E = \bigcup_t E_t$. Every ceer has computable approximations; in fact we can show (see [1]) that there exists a uniform sequence $\{R_{i,s} : i, s \in \omega\}$ of equivalence relations such that the set $\{\langle i, x, y, s\rangle : x \mathrel{R_{i,s}} y\}$ is computable, and for every i, the sequence $\{R_{i,s} : s \in \omega\}$ is a computable approximation to R_i.

2. Computing the complexity of index sets of ceers above, below, or equivalent to a given ceer

Index sets of classes of ceers of natural computability theoretic interest have been investigated for the first time by Gao and Gerdes [10]. Index sets of the form $\{i : R_i \leqslant_c R\}$, $\{i : R \leqslant_c R_i\}$, $\{i : R_i \equiv_c R\}$, for particular choices of R, are classified in [10] to be Σ^0_3 complete: for instance, this is the case when R is the identity relation on the natural numbers. In this section we completely classify all index sets of this type, thus showing for instance (see Corollary 2.6) that if R is a ceer with infinitely many equivalence classes then $\{i : R \leqslant_c R_i\}$, $\{i : R_i \equiv_c R\}$ are always Σ^0_3 complete, and if R has infinitely many classes and is not universal then $\{i : R_i \leqslant_c R\}$ is always Σ^0_3 complete.

Theorem 2.1. *Let R be a non-universal ceer with infinitely many classes. Then $(\Sigma^0_3, \Pi^0_3) \leqslant_1 (\{i \mid R_i \equiv_c R\}, \{i \mid R_i \nleqslant_c R \,\&\, R_i \ngeqslant_c R\})$ (where $(\Sigma^0_3, \Pi^0_3) \leqslant_1 (A, B)$ means that for every Σ^0_3 set C, there is a computable function which reduces C to A, and the complement of C to B: see* [20, p. 66] *for this notation).*

Proof. Fix a Σ^0_3 complete set $S := \{i \mid (\exists l)[W_{g(i,l)} = \omega]\}$, where g is a computable function (the fact that every Σ^0_3 set can be expressed in this way is an easy consequence of the proof of [20, Corollary IV.3.7]). We construct a function which, on input i, outputs an index of a ceer E so that if $i \in S$ then $E \equiv_c R$, and if $i \notin S$ then E and R are $\leqslant_c$-incomparable.

Given i, we describe the enumeration of the ceer E based on the enumeration of the sets $W_{g(i,l)}$ for various l. It will be clear from the construction that an index for E can be uniformly found in i.

Requirements and their strategies. Given i, we have three kinds of requirements:

$Q_l : W_{g(i,l)} = \omega \Rightarrow E \equiv_c R$.

$N_j : (\forall l < j)[W_{g(i,l)} \neq \omega] \Rightarrow \varphi_j$ does not give a reduction witnessing $E \leqslant_c R$.

$P_k : (\forall l < k)[W_{g(i,l)} \neq \omega] \Rightarrow \varphi_k$ does not give a reduction witnessing $R \leqslant_c E$.

Let us fix some computable priority ordering on the requirements. We first describe the action taken by each requirement individually.

Q-requirements. A Q_l-requirement acts as follows: When initialized, Q_l is given a finite set of distinct E-equivalence classes $[b_1]_E, \dots, [b_n]_E$ of numbers created due to higher priority requirements (we will formally define a number being created by a requirement below). Q_l is also given a finite set of elements $c_1, \dots c_n$. Q_l works under the assumption that the classes $[b_i]_E$ are pairwise distinct, and the $[c_i]_R$ are pairwise distinct. If either of these assumptions becomes incorrect, Q_l will be re-initialized. Q_l collapses all remaining elements (those created for lower priority requirements) into one class $[d]_E$, and, beginning with that one class, copies R, using a computable coding function $x \mapsto a(x)$. At every stage wherein $\min(\omega \smallsetminus W_{g(i,l)})$ increases, Q_l again E-collapses every new element which is created due to a lower priority requirement, to d, and continues building its copy of R, E-collapsing codes of elements exactly as the corresponding elements are collapsed by R. If no higher priority requirement acts ever again and in fact $\{c_1, \dots, c_n\}$ are non-equivalent in R, and Q_l acts infinitely often (as $W_{g(i,l)} = \omega$), then we will argue that $E \equiv_c R$. Whenever Q_l acts, it restrains all elements created so far.

N-requirements. An N_j requirement acts as follows: We fix a universal ceer T. When initialized, N_j selects new elements $a(0), a(1) \in \omega$, and E-collapses these elements if and only if 0 and 1 collapse in T. If at some stage, $\varphi_j(a(0))$ and $\varphi_j(a(1))$ converge, and

$$0 \; T \; 1 \Leftrightarrow \varphi_j(a(0)) \; R \; \varphi_j(a(1)),$$

then N_j selects a new element $a(2)$, E-collapsing (for $m, n \leqslant 2$) $a(m)$ to $a(n)$ if and only if $m \; T \; n$. If at a later stage, $\varphi_j(a(2))$ converges and

$$(\forall n, m \leqslant 2)[n \; T \; m \Leftrightarrow \varphi_j(a(n)) \; R \; \varphi_j(a(m)))],$$

then N_j selects a new element $a(3)$. The construction proceeds as such. We will argue that if no higher priority requirement re-initializes N_j, then N_j can choose only finitely many elements $\{a(i) : i \leqslant k\}$, otherwise, we would have $T \leqslant_c R$ via the map $i \mapsto \varphi_j(a(i))$, which contradicts non-universality of R. Thus, φ_j can not be a reduction of E to R. Whenever N_j chooses an element $a(k)$, by initialization it restrains all elements $\leqslant a(k)$.

P-requirements. A P_k-requirement acts as follows: P_k searches for elements $x < y \in \omega$ so that φ_k converges on all inputs $\leqslant y$, $\varphi_k(x)$ and $\varphi_k(y)$ are not restrained by higher priority requirements (so that it is allowed to E-collapse $\varphi_k(x)$ and $\varphi_k(y)$), and $x \not R\, y$. If such are found, then P_k collapses $\varphi_k(x)$ and $\varphi_k(y)$ in E. If, at a later stage, $x\ R\ y$, then P_k is injured and begins again. If, in fact, there are only finitely many elements restrained by higher priority requirements, then some pair of elements x, y will eventually be found so that $x \not R\, y$, and either already $\varphi_k(x)\ E\ \varphi_k(y)$, or $\varphi_k(x)$ and $\varphi_k(y)$ are not restrained by higher priority requirements (since R has infinitely many classes). But then we cause $\varphi_k(x)\ E\ \varphi_k(y)$. This contradicts φ_k being a reduction of R to E after all. As P_k never minds things collapsing, it places no restraints.

Environments for the requirements. A Q_l-requirement uses a parameter $\gamma_l(s) = \langle c_1, \dots, c_n \rangle$, and values of a finite function, $a_l^Q(x, s)$, which approximates the function $x \mapsto a(x)$ described in the above informal discussion for Q-requirements. An N_j-requirement uses a parameter $a_j^N(x, s)$, which approximates the numbers $a(x)$ described in the above informal discussion for N-requirements. In the following, we will often omit the superscripts Q, or N, when the exact choice will be clear from the context. A P_k-requirement uses parameters $x_k(s), y_k(s)$, which approximate the numbers x, y, described in the above informal discussion for P-requirements. If $\mathcal{R}$ is either a Q-requirement or an N-requirement, the construction also uses a parameter $\rho^{\mathcal{R}}(s)$ to record the elements that $\mathcal{R}$ wants to restrain.

Construction. To tackle N-requirements, we fix a universal ceer T, with computable approximations $\{T_s\}_{s\in\omega}$. At stage s, to *initialize* a requirement $\mathcal{R}$ means one of the following:

- if $\mathcal{R} = Q_l$, then we set $\gamma_l(s)$, $a_l(x, s)$ to be undefined for all x; and we set $\rho^{Q_l}(s) = \varnothing$;
- if $\mathcal{R} = N_j$, then we set $a_j(x, s)$ to be undefined, all x; and we set $\rho^{N_j}(s) = \varnothing$;
- if $\mathcal{R} = P_k$, then we set $x_k(s), y_k(s)$ to be undefined.

At stage $s > 0$ we say that a requirement $\mathcal{R}$ *requires attention* if either $\mathcal{R}$ is initialized, or

- $\mathcal{R} = Q_l$ and s is $\langle i, l\rangle$-*expansionary*, i.e., $\min(\omega \smallsetminus W_{g(i,l),s}) > \min(\omega \smallsetminus W_{g(i,l),s-1})$. Or
- $\mathcal{R} = N_j$ and $\varphi_{j,s}(a_j(x, s))$ converges, where x is the greatest number in the domain of $a_j(\_, s)$, and
$$(\forall n, m \leqslant x)[n\ T_s\ m \Leftrightarrow \varphi_{j,s}(a_j(n, s))\ R_s\ \varphi_{j,s}(a_j(m, s)))].$$
Or
- $\mathcal{R} = P_k$, and $\varphi_{k,s}$ converges on all $z \leqslant y_k(s)$, and either $x_k(s) \not R_s\, y_k(s)$ and $\varphi_{k,s}(x_k(s)) \not E_s\, \varphi_{k,s}(y_k(s))$, and both $\varphi_{k,s}(x_k(s))$ and $\varphi_{k,s}(y_k(s))$ have been E-collapsed to elements restrained by higher priority strategies; or at least one of the values $\varphi_{k,s}(x_k(s))$ and $\varphi_{k,s}(y_k(s))$ has not as yet been E-collapsed to any element restrained by higher priority strategies, and $x_k(s) \not R_s\, y_k(s)$ and $\varphi_{k,s}(x_k(s)) \not E_s\, \varphi_{k,s}(y_k(s))$; or $x_k(s)\ R_s\ y_k(s)$.

At stage $s > 0$ a number z is said to have been *created* by a requirement $\mathcal{R}$, if

- $\mathcal{R} = Q_l$, and z is E_s-equivalent to some c_i, where $\gamma_l(t) = \langle c_1, \dots, c_i, \dots c_n \rangle$, or to some $a_l(x,t)$, for some $t \leqslant s$;
- $\mathcal{R} = N_j$, and z is E_s-equivalent to some $a_j(x,t)$, for some $t \leqslant s$;

a number is *new* at s, if it is bigger than all numbers (that are E_s-equivalent to numbers) so far mentioned in the construction.

We are now ready to give the construction.

Step 0. Initialize all requirements.

Step $s+1$. Let $\mathcal{R}$ be the least requirement that requires attention at stage $s+1$. We say that $\mathcal{R}$ *acts* at $s+1$. Notice that there always exists such a requirement, as at each stage infinitely many requirements are initialized. We distinguish the following cases. (For simplicity, when describing the various parameters, or the various approximations to the equivalence relations, or to partial computable functions, we omit to mention the stage s: thus for instance, $x \not\mathrel{R} y$ has to be read as $x \not\mathrel{R_s} y$, and so on.)

(1) If $\mathcal{R} = Q_l$, then we take action as follows:
   (a) if Q_l is initialized, then let n be the number of the distinct equivalence classes created by E, up to s, as the result of the actions taken by the higher priority requirements, and let $\{b_1, \dots, b_n\}$ be representatives of these equivalence classes. Choose γ_l to be the least (by code) n-tuple of numbers that are currently pairwise non-equivalent in R;
   (b) if there exist $1 \leqslant i, j \leqslant n$, $i \neq j$, such that $c_i \mathrel{R} c_j$ (where c_i and c_j are the i-th and j-th components, respectively, of γ_l), then initialize Q_l;
   (c) if neither of the previous two cases holds then:
      (i) take the least number x for which $a_l(x)$ is not defined, and define $a_l(x)$ to be a new number;
      (ii) for every $z \leqslant a_l(x)$ such that $z \not\mathrel{E} b_i$ for each b_i, and $z \not\mathrel{E} a_l(y)$ for each existing $a_l(y)$, then E-collapse z and $a_l(0)$;
      (iii) E-collapse existing $a_l(y)$ and $a_l(z)$ if $y \mathrel{R} z$;

   put into $\rho^{Q_l}(s+1)$ all numbers b_i, $1 \leqslant i \leqslant n$, and $a_l(y)$, $y \leqslant x$, where x is the greatest number for which $a_l(\_)$ is defined.
(2) If $\mathcal{R} = N_j$ then we act as follows:
   (a) if N_j is initialized then we appoint new elements $a_j(0)$ and $a_j(1)$;
   (b) otherwise, let x be the greatest number such that $a_l(\_)$ is defined: for every $y, z \leqslant x$, E-collapse all $a_j(y), a_j(z)$ if $y \mathrel{T} z$; finally, appoint a new $a_j(x+1)$;

   put in $\rho^{N_j}(s+1)$ all numbers $a_j(i)$, $i \leqslant x$, where x is the greatest number for which $a_j(\_)$ is defined.
(3) If $\mathcal{R} = P_k$ then we act as follows:
   (a) if P_k is initialized then appoint x_k and y_k so that $\langle x_k, y_k \rangle$ is the least pair $\langle x, y \rangle$ for which $x < y$, $x \not\mathrel{R} y$, and $\langle x, y \rangle > \langle x_k(t), y_k(t) \rangle$, for every $t \leqslant s$;
   (b) if $x_k \not\mathrel{R} y_k$ and $\varphi_k(x_k) \not\mathrel{E} \varphi_k(y_k)$, but both $\varphi_k(x_k)$ and $\varphi_k(y_k)$ are already E-equivalent to restrained elements, i.e., elements belonging to the set

$$S(s) := \{z : (\exists \mathcal{R})[\mathcal{R} \text{ has higher priority than } P_k \ \&\ z \in \rho^{\mathcal{R}}(s)]\}$$

   then initialize P_k;
   (c) if $x_k \not\mathrel{R} y_k$ and $\varphi_k(x_k) \not\mathrel{E} \varphi_k(y_k)$, and at least one of them has not as yet been E-collapsed to a restrained element, then E-collapse $\varphi_k(x_k)$ and $\varphi_k(y_k)$;

(d) if $x_k \mathrel{R} y_k$, then initialize P_k.

After acting, end the stage, and initialize all lower priority requirements.

Verification. We now check that the construction works.

Lemma 2.2. *If every higher priority requirement acts only finitely often, then P_k acts only finitely often.*

Proof. Assume that every higher-priority requirement acts only finitely often, and suppose, towards a contradiction, that P_k acts infinitely often. Let s be the last stage at which a higher-priority requirement acts. Let S be the set of all elements E-equivalent to some element in $S(s)$ which is the finite set restrained by higher priority actions by stage s, as in (3b) of step $s+1$. Thus S is the union of finitely many E-equivalence classes. For P_k to act infinitely often, we must have φ_k total, and by (3a) and (3d), we test all possible choices of x_k, y_k, with $x_k < y_k$ and $x_k \not\mathrel{R} y_k$: for each one of these pairs (by definition of P_k requiring attention) we have that $\varphi_k(x_k), \varphi_k(y_k) \in S$ and $\varphi_k(x_k) \not\mathrel{E} \varphi_k(y_k)$. But this would imply that there exists a 1-1 function from the infinitely many distinct equivalence classes of R to the finitely many equivalence classes in S. Therefore, we must have that either P_k eventually does not require attention because φ_k is not total; or we find x_k, y_k such that $x_k \not\mathrel{R} y_k$, $\varphi_k(x_k), \varphi_k(y_k) \in S$, and $\varphi_k(x_k) \mathrel{E} \varphi_k(x_k)$; or (3c) applies. ☐

Lemma 2.3. *If every higher priority requirement acts only finitely often, then N_j acts only finitely often.*

Proof. Suppose, towards a contradiction, that N_j acts infinitely often. Let s be the last stage at which a higher priority requirement acts, i.e. N_j is initialized for the last time at stage s. We consider the assignments of $a_j(k)$ after stage s. Then for each n, m,

$$n \mathrel{T} m \Leftrightarrow \varphi_j(a_j(n)) \mathrel{R} \varphi_j(a_j(m)).$$

Thus the function $i \mapsto \varphi_j(a_j(i))$ gives a reduction of T to R. This yields a contradiction since R is non-universal, showing that N_j acts only finitely often. ☐

Lemma 2.4. *Suppose that $W_{g(i,l)}$ is finite for each l. Then E is $\leqslant_c$-incomparable to R.*

Proof. By assumption, each Q_l acts only finitely often, so by Lemmas 2.2 and 2.3, every requirement acts only finitely often.

We now argue that since each requirement acts only finitely often, each succeeds. Since every requirement acts only finitely often, we can consider the final assignments of $a_j(k)$ for the requirement N_j. Either φ_j is not total or for some $a_j(n), a_j(m)$,

$$a_j(n) \mathrel{E} a_j(m) \Leftrightarrow n \mathrel{T} m,$$

but not

$$n \mathrel{T} m \Leftrightarrow \varphi_j(a_j(n)) \mathrel{R} \varphi_j(a_j(m)).$$

Thus φ_j is not a reduction of E to R, and N_j is satisfied. Since P_k acts only finitely often and R has infinitely many classes, either φ_k is not total or there are x_k, y_k so that $x_k \not\mathrel{R} y_k$ but $\varphi_k(x_k) \mathrel{E} \varphi_k(y_k)$. Thus φ_k is not a reduction of R to E, and P_k is satisfied. ☐

Lemma 2.5. *Suppose that for some l, $W_{g(i,l)} = \omega$. Then $E \equiv_c R$.*

Proof. Let Q_l be of highest priority so that $W_{g(i,l)} = \omega$. By Lemmas 2.2 and 2.3, every higher priority requirement acts only finitely often. Consider the least stage t at which every higher priority action stops acting, giving n distinct equivalence classes. Further, consider a stage $s > t$ where Q_l has found (through (1a) and (1d)) the appropriate choice of n R-non-equivalent elements, thus choosing the final γ_l. After this stage s, every time Q_l picks a number $a_l(x)$, then this is the final value of $a_l(x,s)$, and Q_l creates a class $[d]$, with $d = a_l(0)$, which contains all elements previously created for all lower priority requirements, and it will also contain all elements later created for lower priority requirements (when Q_l acts again, it will E-collapse them to d). We now provide reductions witnessing that $E \equiv_c R$.

To see $E \leqslant_c R$, consider the function f constructed as follows: We begin with the finitely many elements created for higher priority requirements, which are grouped into finitely many finite E-equivalence classes, as created at stage s: $[b_1]_E = [b_1]_{E_s}, \ldots, [b_n]_E = [b_n]_{E_s}$. We have found $c_1, \ldots, c_n$ so that $b_i \; E \; b_j$ if and only if $c_i \; R \; c_j$, so that the assignment $b_k \mapsto c_k$ satisfies

$$(\forall 1 \leqslant h, k \leqslant n)[b_h \; E \; b_k \Leftrightarrow c_h \; R \; c_k].$$

Let T be the set of elements $a \in \omega$ created on a Q_l-stage after s (i.e., a stage $\geqslant s$ where $\min(\omega \smallsetminus W_{g(i,l)})$ increases). Note that $d \in T$. For any $a \in T$, a is created to copy R on some number, i.e. $a = a_l(x)$ for some x. So, consider the function f,

$$f(x) = \begin{cases} c_i, & \text{if } x \in [b_i]_{E_s}, \text{ some } 1 \leqslant i \leqslant n, \\ y, & \text{if } x \in T, \text{ say } x = a_l(y), \\ 0, & \text{otherwise.} \end{cases}$$

(Notice that $0 = f(a_l(0)) = f(d)$.) The numbers x not created on Q_l-stages, are either in some $[b_i]_{E_s}$, or are created for lower priority requirements: in this latter case, $x \; E \; d$, for which we have defined $f(x) = f(d)$. This function f is computable and witnesses that $E \leqslant_c R$.

For the converse, the mapping $x \mapsto a_l(x)$ provides a reduction from R to E. ☐

This concludes the proof of the theorem.

☐

Corollary 2.6. *The following hold:*

(1) If R is any ceer with infinitely many classes, then $\{i \mid R_i \equiv_c R\}$ is Σ^0_3 complete.
(2) If R is any ceer with infinitely many classes, then $\{i \mid R_i \geqslant_c R\}$ is Σ^0_3 complete.
(3) If R is any non-universal ceer with infinitely many classes, then $\{i \mid R_i \leqslant_c R\}$ is Σ^0_3 complete.
(4) If R is universal, then $\{i \mid R_i \leqslant_c R\} = \omega$, thus is decidable.
(5) If R has only finitely many classes, then $\{i \mid R_i \leqslant_c R\}$ is Π^0_2 complete.
(6) If R has finitely many, but at least 2, classes, then $\{i \mid R_i \geqslant_c R\}$ is Σ^0_2 complete.
(7) If R has only one class, then $\{i \mid R_i \geqslant_c R\} = \omega$, thus is decidable.
(8) If R has finitely many, but at least 2, classes, then $\{i \mid R_i \equiv_c R\}$ is d-Σ^0_2 complete (i.e., $\{i \mid R_i \equiv_c R\}$ is the intersection of a Σ^0_2 and a Π^0_2 set, and if X is any set which is the intersection of a Σ^0_2 and a Π^0_2 set, then $X \leqslant_m \{i \mid R_i \equiv_c R\}$).

Proof. It is straightforward to check that the proposed sets lie in the appropriate level of the arithmetical hierarchy. To how hardness, we prove the items one by one.

(1) There are two cases. If R is universal, this is exactly Theorem 5.1 in [1]. If R is non-universal, this follows directly from the previous theorem.
(2) If R is universal, then the claim follows from (1). If not, then it follows from the previous theorem.
(3) This follows from the previous theorem.
(4) Trivial.
(5) Note that if R has k classes, then $E \leqslant R$ if and only if E has $\leqslant k$ classes. It is easy to show that having $\leqslant k$ classes is a Π^0_2 complete property: E has $\leqslant k$ classes if and only if

$$(\forall x_0, \dots, x_k)(\exists i, j \leqslant k)[i \neq j \,\&\, x_i \; E \; x_j].$$

Let us now show that this property is Π^0_2 hard. It is known that $\mathrm{Inf} = \{i \mid W_i \text{ infinite}\}$ is Π^0_2 complete: it is easy to see that there is a computable function f such that, for every i, $E_{f(i)}$ is a ceer satisfying:

$$i \in \mathrm{Inf} \Rightarrow [0]_{E_{f(i)}} = \omega,$$
$$i \notin \mathrm{Inf} \Rightarrow (\exists x)(\forall y \geqslant x)\left[[y]_{E_{f(i)}} = \{y\}\right].$$

(6) If R has $k \geqslant 2$ classes, then $E \geqslant_c R$ holds if and only if $E \nleqslant_c S$, where S has $k-1$ classes. Thus, by (5), this is Σ^0_2 complete.
(7) Trivial.
(8) By combining the arguments in (5) and (6). Note that if R has exactly one class, then $\{i \mid R_i \equiv_c R\} = \{i \mid R_i \leqslant_c R\}$ is Π^0_2 complete by (5).

□

3. The index set of the effectively inseparable ceers

A pair of disjoint sets A, B is *effectively inseparable* (shortly, *e.i.*) if there exists a partial computable function ψ (called a *productive function* for the pair) such that, for every pair of c.e. indices u, v,

$$A \subseteq W_u \,\&\, B \subseteq W_v \,\&\, W_u \cap W_v = \emptyset \Rightarrow \psi(u,v)\downarrow \;\&\, \psi(u,v) \notin W_u \cup W_v.$$

It is not difficult to see:

Lemma 3.1. *Every e.i. pair of c.e. sets, has a total productive function.*

Proof. The proof is similar to the one showing that every productive set has a total productive function, see e.g. [20, p. 41]. □

A ceer R is called *effectively inseparable* (shortly, *e.i.*), see [1], if every pair of distinct equivalent classes $[a]_R, [b]_R$ is e.i.. If indices for productive functions for the various pairs of equivalence classes can be found uniformly (i.e., there exists a computable function g such that, for every pair a, b, if $a \not\mathrel{R} b$ then $\varphi_{g(a,b)}$ is a productive function for the pair $[a]_R, [b]_R$), then R is said to be *uniformly effectively inseparable* (or, shortly, *u.e.i.*), [1]. It is proved in [1] that the index set of the u.e.i. ceers is Σ^0_3 complete, and is posed as an open question whether the index set of the e.i. ceers is Π^0_4 complete. In the following theorem we answer this question.

Theorem 3.2. *The index set of the e.i. ceers is Π^0_4 complete.*

Proof. It is straightforward to check that the index set of the e.i. ceers is Π^0_4. Now, every Π^0_4 set S can be described as $S = \{i : (\forall j)[W_{g(i,j)}$ is cofinite$]\}$: this is an easy consequence of the fact that the index set $\{i : W_i$ is cofinite$\}$ is Σ^0_3 complete (see e.g., [20, p. 66]). Therefore, we can fix a recursive function $g(i,j)$ so that $S := \{i \mid (\forall j)[W_{g(i,j)}$ is cofinite$]\}$ is a Π^0_4 complete set. We now produce a function which, on input i, uniformly produces a ceer E so that E is e.i. if and only if $i \in S$. In what follows, we describe the enumeration of E for a given i.

Given a set X let $X^{[2]}$ denote the collections of all subsets of X consisting of exactly two elements. We fix a pair of recursive bijections $m : \omega^{[2]} \to \omega$ and $n_0 : (2\omega)^{[2]} \to \omega$, where 2ω is the set of even elements of ω. We then define $n : \omega^{[2]} \to \omega$ so that $n(x) = n_0(x)$ if $x \in (2\omega)^{[2]}$, and $n(x) = m(x)$ otherwise.

Requirements and strategies. We have the following requirements, where $a < b$, i.e., $\{a,b\} \in \omega^{[2]}$.

$$\begin{aligned} P^{a,b}_j : & [j,\infty) \subseteq W_{g(i,m(a,b))} \cap W_{g(i,n(a,b))} \ \&\ a \not\mathrel{E} b \Rightarrow\ f^{a,b}_j \text{ is a productive function for } [a]_E, [b]_E, \\ & \text{(where } f^{a,b}_j \text{ is a computable function being constructed by this requirement)} \\ N^{a,b}_j : & [j,\infty) \nsubseteq W_{g(i,m(a,b))} \cap W_{g(i,n(a,b))} \ \&\ a,b \in (2\omega)^{[2]} \Rightarrow\ \varphi_j \text{ is not productive for } [a]_E, [b]_E. \end{aligned}$$

The requirements are partitioned, in the obvious way, into P*-requirements* and N*-requirements*.

Remark 3.3. If $i \in S$ then for every r, $W_{g(i,r)}$ is cofinite, and thus for every $a \neq b$ there is $j_{a,b}$ such that $[j_{a,b},\infty) \subseteq W_{g(i,m(a,b))} \cap W_{g(i,n(a,b))}$: hence if $a \not\mathrel{E} b$, and we satisfy $P^{a',b'}_{j_{a',b'}}$, where a' and b' are the least elements in the E-equivalence classes of a and b, respectively, then we guarantee that $f^{a',b'}_{j_{a',b'}}$ is a productive function for the pair $[a]_E, [b]_E$.

Vice versa, if $i \notin S$, then there is r such that $W_{g(i,r)}$ is not cofinite, nor is any $W_{g(i,r')} \cap W_{g(i,r)}$, and thus if $a, b \in 2\omega$, $a \neq b$, are such that $n(a,b) = r$, then for every j, one has $[j,\infty) \nsubseteq W_{g(i,m(a,b))} \cap W_{g(i,n(a,b))}$. In this case, if we satisfy all $N^{a,b}_j$-requirements, then we guarantee that the pair $[a]_E, [b]_E$ is not effectively inseparable.

We will never cause non-equal even elements a, b to become E-equivalent, and in fact each even number will be the least element in its equivalence class. $N^{a,b}_j$-requirements will only pose restraints asking that two elements not become equivalent, but will never cause E-collapse.

We fix any priority ordering of order type ω in which if $j < j'$ then $P^{a,b}_j < P^{a,b}_{j'}$.

We first describe the actions of each requirement separately. The reader should think of a, b as the least numbers in their respective equivalence classes, and $a \neq b$.

P-Requirements. A $P^{a,b}_j$-requirement performs the standard effective inseparability strategy: it builds a computable function $f = f^j_{a,b}$ as follows. For the least (by code) pair u, v, on which $f(u,v)$ is still undefined, define $f(u,v)$ to be an odd number y larger than any number considered so far: if y is observed to be enumerated into W_u, cause $y \mathrel{E} b$; if y is observed to be enumerated into W_v, then cause $y \mathrel{E} a$. The strategy for $P^{a,b}_j$ acts every time the least element of $[j,\infty) \smallsetminus$

$W_{g(i,m(a,b))} \cap W_{g(i,n(a,b))}$ enters $W_{g(i,m(a,b))} \cap W_{g(i,n(a,b))}$, i.e. when there is evidence that eventually $[j,\infty) \subseteq W_{g(i,m(a,b))} \cap W_{g(i,n(a,b))}$: we say that in this case the strategy $P_j^{a,b}$ *takes the infinite outcome*; otherwise $P_j^{a,b}$ *takes the finite outcome.* It is clear that either $P_j^{a,b}$ takes the infinite outcome infinitely many times (we say that in this case that $P_j^{a,b}$ *has outcome* ∞), or from some point on, $P_j^{a,b}$ always takes the finite outcome (we say that in this case that $P_j^{a,b}$ *has outcome f.*)

We summarize as follows:

$$\begin{aligned} &\text{outcome } \infty : [j,\infty) \subseteq W_{g(i,m(a,b))} \cap W_{g(i,n(a,b))};\\ &\text{outcome f} : [j,\infty) \nsubseteq W_{g(i,m(a,b))} \cap W_{g(i,n(a,b))}. \end{aligned}$$

N-Requirements. An $N_j^{a,b}$-requirement acts only if, and immediately after, $P_j^{a,b}$ has taken the finite outcome, and its action is as follows: Choose a pair u, v so that we (via the Recursion Theorem) control the enumeration of W_u and W_v. Let W_u enumerate $[a]_E$ and W_v enumerate $[b]_E$, and wait for a stage when $\varphi_j(u,v)$ converges to a value, say y. If $y \in [a]_E \cup [b]_E$, then the requirement does nothing further. Otherwise, we distinguish:

Case 1: y is an odd number chosen as $f_{j'}^{a',b'}(u',v')$ for some j', a', b' with a', b' least numbers in their equivalence classes, and $a' \neq a$ (if $a' = a$ and $b' \neq b$, the requirement acts symmetrically). In this case, we enumerate y into W_v. We place a restraint for y to never enter $[a]$.

Case 2: Not case 1. In this case, we enumerate y into W_u and place a restraint for y to never enter $[b]$.

Every time the least element of $[j,\infty) \cap W_{g(i,m(a,b))} \cap W_{g(i,n(a,b))}$ enters $W_{g(i,m(a,b))} \cap W_{g(i,n(a,b))}$, $N_j^{a,b}$ will be injured, so if $[j,\infty) \subseteq W_{g(i,m(a,b))} \cap W_{g(i,n(a,b))}$, then N will not prevent effective inseparability of the pair $[a],[b]$.

The Recursion Theorem. In carrying on the strategies for the N-requirements, we use indices that we control by the Recursion Theorem, or, more precisely, we make use of a computable sequence of fixed points. Equivalently, we fix a single index e so that we control φ_e by the Recursion Theorem, and we then take a countable sequence of indices $(e_i)_{i\in\omega}$ for the columns $\varphi_{e_i}(j) = \varphi_e(\langle i,j\rangle)$. We can then make choices about convergence and values of each of the φ_{e_i} in any order we wish, as we are simply controlling the single function φ_e.

Alternatively, since a computable sequence of indices can be viewed as the range of a computable function f, a formal justification to this argument is also provided by the Case Functional Recursion Theorem, see [3]: see also [17] for useful comments about this theorem.

Lemma 3.4 (Case Functional Recursion Theorem)**.** *Given a partial computable functional F, there is a total computable function f such that, for every e, x,*

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

The tree of strategies. We organize the construction on a tree T, which is a set of strings on the alphabet $\{\mathrm{g}, \infty, \mathrm{f}\}$. With respect to the above discussion of requirements and their outcomes, it is convenient to use also an additional outcome g, which for a requirement $P_j^{a,b}$ or $N_j^{a,b}$, will record the fact that at least one among a, b is not the least number in its equivalence class.

The tree T, and the function

$$\mathcal{R} : T \longrightarrow \text{ Requirements},$$

assigning requirements to the nodes of T, are defined as follows, where λ denotes the empty string.

Definition 3.5. $\lambda \in T$, and $\mathcal{R}(\lambda)$ is the highest priority P-requirement.

- If $\sigma \in T$, and $\mathcal{R}(\sigma) = P^{a,b}_j$ is a P-requirement, then $\sigma^\frown\langle o\rangle \in T$, for $o \in \{\mathrm{g}, \infty, \mathrm{f}\}$:
  - all requirements $P^{a,b}_{j'}$ for $j' > j$ are declared to be *cancelled* by $\sigma^\frown\langle\infty\rangle$. (Since if $[j, \infty) \subseteq W_{g(i,m(a,b))} \cap W_{g(i,n(a,b))}$, then $[j', \infty) \subseteq W_{g(i,m(a,b))} \cap W_{g(i,n(a,b))}$ for all $j' \geqslant j$, thus the requirement $P^{a,b}_{j'}$ need not be considered again below $\sigma^\frown\langle\infty\rangle$.) $\mathcal{R}(\sigma^\frown\langle\infty\rangle)$ is the highest priority P-requirement not assigned to any $\tau \subseteq \sigma$ and not cancelled by any $\tau \subseteq \sigma^\frown\langle\infty\rangle$.
  - If a, b are both even, then $\mathcal{R}(\sigma^\frown\langle\mathrm{f}\rangle) = N^{a,b}_j$; otherwise $\mathcal{R}(\sigma^\frown\langle\mathrm{f}\rangle)$ is the highest priority P-requirement not assigned to any $\tau \subseteq \sigma$ and not cancelled by any $\tau \subseteq \sigma$.
  - $\mathcal{R}(\sigma^\frown\langle\mathrm{g}\rangle)$ is the highest priority P-requirement not assigned to any $\tau \subseteq \sigma$ and not cancelled by any $\tau \subseteq \sigma$.
- If $\sigma \in T$, and $\mathcal{R}(\sigma)$ is an N-requirement, then $\sigma^\frown\langle\mathrm{f}\rangle \in T$ (by construction, a, b will be the least elements in their respective equivalence classes, so we do not consider the g outcome); $\mathcal{R}(\sigma^\frown\langle\mathrm{f}\rangle)$ is the highest priority P-requirement not assigned to any $\tau \subseteq \sigma$ and not cancelled by any $\tau \subseteq \sigma$.
- No other string on $\{\infty, \mathrm{f}, \mathrm{g}\}$ lies in T.

The elements of T are ordered by the *lexicographical order* $\leqslant$, generated by the ordering on the alphabet, for which $\mathrm{g} < \infty < \mathrm{f}$: thus $\sigma \leqslant \tau$ if $\sigma \subseteq \tau$ or, for the least i such that σ, τ are both defined on i, and $\sigma(i) \neq \tau(i)$, we have that $\sigma(i) < \tau(i)$: in this latter case we also write $\sigma <_L \tau$.

The environments of the strategies. Notice that the function $\mathcal{R}$, assigning requirements to nodes, is computable. For every σ, we also call $\mathcal{R}(\sigma)$ a *strategy*. Each strategy has several parameters: if $\mathcal{R}(\sigma) = P^{a,b}_j$ then it uses the parameter $f_{\sigma,s}$ (approximating the function $f^{a,b}_j$ of the above informal description), whereas if $\mathcal{R}(\sigma)$ is an N-requirement, then it uses the parameters $u_\sigma(s), v_\sigma(s)$, and $y_\sigma(s)$ (approximating u, v, y of the above informal description).

The construction. At stage s we define a finite string δ_s of length $|\delta_s| \leqslant s$, which approximates the true path at stage s. The string δ_s is defined by substages: at substage n, we define $\sigma_n = \delta_s{\upharpoonright}n$. A number is *new* at any substage of stage $s > 0$ if it is bigger than all numbers already E-collapsed to numbers so far mentioned in the construction. If $\sigma = \sigma_n$ and $\mathcal{R}(\sigma) = P^j_{a,b}$ is a P-strategy, then a stage s is *σ-expansionary* if for no $t < s$ did we have $\sigma \subseteq \delta_t$, or $\min\big([j, \infty) \smallsetminus (W_{g(i,m(a,b)),s} \cap W_{g(i,n(a,b),s)})\big)$ has increased since the last stage $t < s$ which was *σ-true*, i.e., at which $\sigma \subseteq \delta_t$. A number z is *created by* $\mathcal{R}(\sigma)$ at s, if z is in the range of $f_{\sigma,s}$; or, z is appointed as $u_\sigma(s)$ or $v_\sigma(s)$, or $y_\sigma(s)$. At stage s, we *initialize a strategy* $\mathcal{R}(\sigma)$ if we set $f_{\sigma,s} = \varnothing$ and we set $u_\sigma(s), v_\sigma(s)$, and $y_\sigma(s)$ to be undefined. If y has been created by $\mathcal{R}(\sigma) = P^{a,b}_j$, by stage s, then y is *active* at s if $\mathcal{R}(\sigma)$ has not been initialized after y has been created, and $\mathcal{R}(\sigma)$ has not as yet E-collapsed y to either a or b.

Stage 0. Initialize all strategies $\mathcal{R}(\sigma)$.

Stage $s+1$. Proceed according to the following substages (as in the proof of Theorem 2.1, when describing the various parameters, or the various approximations to c.e. sets, partial computable functions, or E, we omit mentioning the stage s):

Substage 0. Let $\delta_{s+1}\upharpoonright 0 = \lambda$.

Substage $n+1$. If $n = s$ then go to next stage. Otherwise, take the first relevant case that applies below:

(1) Suppose that $\mathcal{R}(\sigma_n) = P_j^{a,b}$.
   (a) If one among a, b is not the least element of its E-equivalence class, then let $\sigma_{n+1} = \sigma_n{}^\frown\langle \mathrm{g}\rangle$.
   (b) If s is a σ_n-expansionary stage, let $\sigma_{n+1} = \sigma_n{}^\frown\langle\infty\rangle$. Then extend f_{σ_n} by considering the least (by code) pair (u, v) on which f_{σ_n} is not defined, and define $f_{\sigma_n}(u, v) = y$, for some new odd $y > a, b$. Also, if $f_{\sigma_n}(u', v') = y'$ has been already defined, and up to now y' has been active, but currently $y' \in W_{u'} \cup W_{v'}$, then
      (i) if $y' \in W_{u'}$ then E-collapse y' and b;
      (ii) if $y' \in W_{v'}$ then E-collapse y' and a.
   (c) Otherwise, let $\sigma_{n+1} = \sigma_n{}^\frown\langle \mathrm{f}\rangle$.

(2) If $\mathcal{R}(\sigma_n) = N_j^{a,b}$, then let $\sigma_{n+1} = \sigma_n{}^\frown\langle \mathrm{f}\rangle$. We act according to the first applicable case among the following:
   (a) $\mathcal{R}(\sigma_n)$ is initialized: assume by the Recursion Theorem that u and v are indices that we control, such that u and v are new numbers; let $u_\sigma(s+1) = u$, $v_\sigma(s+1) = v$;
   (b) $\varphi_j(u, v)$ converges to some number y (where $u = u_\sigma(s)$, $v = v_\sigma(s)$, and we define $y_\sigma(s+1) = y$);
      (i) if $s+1$ is the first σ_n-true stage at which $\varphi_j(u, v)$ converges, then end the stage (thus initializing all strategies of lower priority);
      (ii) if y is E-equivalent to some active $f_\tau(u', v')$ created by $\mathcal{R}(\tau) = P_{j'}^{a',b'}$, with $\tau^\frown\langle\infty\rangle \subseteq \sigma_n$, and $\{a', b'\} \neq \{a, b\}$, then if $a' \neq a$, enumerate y into W_v; otherwise (i.e., $a' = a$, but $b' \neq b$), enumerate y into W_u; (notice that by the way requirements are assigned to strings in T, there is no $\tau^\frown\langle\infty\rangle \subseteq \sigma_n$ with $\mathcal{R}(\tau) = P_{j'}^{a,b}$, any j'). Also, enumerate $[a]_E$ into W_u and $[b]_E$ into W_v.
      (iii) if $y \in [a]_E \cup [b]_E$ then enumerate $[a]_E$ into W_u, and enumerate $[b]_E$ into W_v;
      (iv) otherwise, enumerate $[a]_E \cup \{y\}$ in W_u, and $[b]_E$ in W_v.

At the end of the stage, initialize all strategies $\mathcal{R}(\tau)$, with $\tau \geqslant \delta_{s+1}$. Define E_{s+1} to be the least equivalence relation generated by E_s plus the pairs E-collapsed at stage $s+1$. This ends Stage $s+1$.

Finally, let

$$E = E_i = \bigcup_s E_s.$$

The verification. The following holds:

Lemma 3.6. *There exists an infinite path* tp *through the tree* T *such that, for every* n,

$$\mathrm{tp}\upharpoonright n = \liminf_s \delta_s\upharpoonright n,$$

(where the $\liminf$ *is taken with respect to the lexicographical order of strings of* T*), and* $\mathrm{tp}\restriction n$ *eventually does not end the stage.*

Proof. The proof is by induction on n. Suppose that the claim is true of n, and let s_0 be the least stage such that there is no σ-true stage $s \geqslant s_0$ for any $\sigma <_L \mathrm{tp}\restriction n$, and $\mathrm{tp}\restriction n$ does not end the stage at s: thus $s_0 > n$. If there is a stage $s_1 \geqslant s_0$ such that $\mathrm{tp}\restriction n^\frown\langle \mathrm{g}\rangle \subseteq \delta_{s_1}$, then for every $\mathrm{tp}\restriction n$-true $s \geqslant s_1$ we have $\mathrm{tp}\restriction n^\frown\langle \mathrm{g}\rangle \subseteq \delta_s$, and if $s > n+1$ then $\mathrm{tp}\restriction n^\frown\langle \mathrm{g}\rangle$ does not end the stage, and clearly $\mathrm{tp}\restriction n+1 = \mathrm{tp}\restriction n^\frown\langle \mathrm{g}\rangle$. If for almost all true $\mathrm{tp}\restriction n$-true stages $s \geqslant s_0$ we have $\mathrm{tp}\restriction n^\frown\langle \mathrm{f}\rangle \subseteq \delta_s$, then $\mathrm{tp}\restriction n+1 = \mathrm{tp}\restriction n^\frown\langle \mathrm{f}\rangle$, and $\mathrm{tp}\restriction n+1$ ends at most twice, at any such s: namely, if $s = n+1$, and when we act through (2bi) of the construction. Otherwise there exist infinitely many true $\mathrm{tp}\restriction n$-true stages $s \geqslant s_0$ at which $\mathrm{tp}\restriction n^\frown\langle \infty\rangle \subseteq \delta_s$: thus $\mathrm{tp}\restriction n+1 = \mathrm{tp}\restriction n^\frown\langle \infty\rangle$ and $\mathrm{tp}\restriction n+1$ does not end the stage at any such $s > n+1$. ☐

Lemma 3.7. *Let* σ *be so that* $\sigma \subset \mathrm{tp}$ *and* $\sigma^\frown\langle g\rangle \not\subseteq \mathrm{tp}$*. If* σ *is an* $N_j^{a,b}$ *or* $P_j^{a,b}$ *strategy, then* a *and* b *are the least numbers in their respective equivalence classes.*

Proof. Immediate. ☐

Lemma 3.8. *At every stage* s*, in any equivalence class* $[c]_{E_s}$ *there is at most one element which is even or active. If, at some stage* s *where* c *is not new, the class* $[c]_{E_s}$ *contains no even or active element, then for all* $t > s$*,* $[c]_{E_t}$ *contains no even or active element. Similarly, if at some stage* s *where* c *is not new,* $[c]_{E_s}$ *contains no element active for requirement* $P_j^{a,b}$*, then at no stage does* $[c]_E$ *contain an element active for requirement* $P_j^{a,b}$*.*

Proof. We prove the first claim by induction. This is clearly true at stage 0 where every equivalence class has size 1. When we activate a new number, we choose it to be a new odd element, thus is inequivalent to any even or active number. When we collapse classes $[a]$ and $[y]$, it is because some element y' in $[y]$ is active and equals $f_\sigma(a', b')$ for some $a' \in [a]$ and some b' (or symmetrically, it equals $f_\sigma(c', a')$ for some $a' \in [a]$ and some c'). We then make y' inactive and collapse $[y]$ to $[a]$. Thus there is still at most one even or active element in the class $[a]$. The second statement is proved analogously: Any element which becomes active is new, thus is not E-equivalent to c, and the property of not containing an even or active element is preserved when a second class collapses with $[c]$. The last statement is similar. ☐

Lemma 3.9. *Every even number is the least number in its* E*-equivalence class.*

Proof. By the previous lemma, no two even numbers are ever equivalent.

We now show that if a is even, then a is the least number in its equivalence class. By the previous conclusion, it is enough to show that for every s, and odd number y, if y is E-collapsed to a at s, then $y > a$. Assume that the claim is true of all odd numbers y' already E-collapsed to a at stages $s' < s$. An odd number y can be moved to $[a]_{E_s}$ at s, either because (1bi) or (1bii) for some $P_j^{a,b}$, but then $y > a$, by choice of $y > a, b$ in (2); or y is E-collapsed, through (1bi) or (1bii) for some $P_{j'}^{y',b'}$, to some some odd number y' previously E-collapsed to a, but then by induction and, again, choice of y by (1b) of the construction, we have $y > y' > a$. ☐

Lemma 3.10. *If* $i \in S$ *then for every* a, b*, if* $a \not\mathrel{E} b$*, the pair* $[a]_E, [b]_E$ *is e.i.. On the other hand, if* $i \notin S$ *then there are* a, b *even numbers such that* $W_{g(i,n(a,b))}$ *is co-infinite, and the pair* $[a]_E, [b]_E$ *is not e.i.*

Proof. If $i \in S$, then (see Remark 3.3) for every a, b there exists a minimal j, such that $[j, \infty) \subseteq W_{g(i,m(a,b))} \cap W_{g(i,n(a,b))}$. Now, if $a \not\!E\, b$, and a, b are the least numbers in their respective equivalence classes, then there exists n such that $\mathcal{R}(\mathrm{tp}\restriction n) = P^{a,b}_j$ and $\mathrm{tp}\restriction(n+1) = \mathrm{tp}\restriction n^\frown\langle\infty\rangle$. (Notice that, under these assumptions, for every $j' < j$ there is a node $\tau_{j'}$ such that $\mathcal{R}(\tau_{j'}) = P^{a,b}_{j'}$, and $\tau_{j'}^\frown\langle \mathrm{f}\rangle \subset \mathrm{tp}$, and for every $j' > j$ there is no node $\tau \subset \mathrm{tp}$ such that $\mathcal{R}(\tau) = P^{a,b}_{j'}$.) It is clear by the construction that $f_{\mathrm{tp}\restriction n}$ is a computable function witnessing that the pair $[a]_E, [b]_E$ is e.i.. Thus every pair of distinct E-equivalence classes is e.i., as on the true path the corresponding requirement relative to the least numbers in the classes, is satisfied.

Assume now that $i \notin S$. Then, by surjectivity of the function n_0, there exists a pair a, b of distinct even numbers such that, for every j, $[j, \infty) \nsubseteq W_{g(i,m(a,b))} \cap W_{g(i,n(a,b))}$. By Lemma 3.9, for every j there is a (unique) node $\tau_j \subset \mathrm{tp}$ such that $\mathcal{R}(\tau_j) = P^{a,b}_j$ and $\tau_j^\frown\langle \mathrm{f}\rangle \subset \mathrm{tp}$. We show that, for every j, φ_j can not be a total productive function for the disjoint pair $[a]_E, [b]_E$. Let s_0 be the least stage such that there is no τ-true stage $s \geqslant s_0$ for any $\tau <_L \tau_j^\frown\langle \mathrm{f}\rangle$, and no $\tau \subseteq \tau_j$ ends the stage after s_0. At the least $\tau_j^\frown\langle \mathrm{f}\rangle$-stage following s_0 we appoint the last choice of $u = u_{\tau_j^\frown\langle \mathrm{f}\rangle}(s)$, and $v = v_{\tau_j^\frown\langle \mathrm{f}\rangle}(s)$. If we do not find y as in (2b) of the construction, then φ_j is not total. So assume that $\varphi_j(u, v)$ converges to y, which is the final value of $y_{\tau_j^\frown\langle \mathrm{f}\rangle}(s)$. We claim that $[a]_E \subseteq W_u$, $[b]_E \subseteq W_v$, $W_u \cap W_v = \varnothing$, but $y \in W_u \cup W_v$, which implies that φ_j is not a productive function. Now, it is clear that $[a]_E \subseteq W_u$, $[b]_E \subseteq W_v$, since there are infinitely many stages s at which we enumerate $[a]_{E_s}$ into W_u and $[b]_{E_s}$ into W_v. It is also clear that $y \in W_u \cup W_v$. It remains to see that $W_u \cap W_v = \varnothing$. Assume that $\mathcal{R}(\tau_j^\frown\langle \mathrm{f}\rangle)$ enumerates y into W_u: the case in which $\mathcal{R}(\tau_j^\frown\langle \mathrm{f}\rangle)$ enumerates y into W_v is similar.

By initialization in (2a) and Lemma 3.8, the number y will never be equivalent to an element active for a $\tau > \tau_j^\frown\langle \mathrm{f}\rangle$.

For y to eventually become E-equivalent to a or b, it must be equivalent at stage s_0 to some active element d for some $\mathcal{R}(\tau) = P^{a',b'}_{j'}$ with $\tau^\frown\langle\infty\rangle \subseteq \tau_j$. By our use of the outcome g, a', b' are the least numbers in their equivalence classes (and so are a and b), and since there is no such τ with $\tau^\frown\langle\infty\rangle \subseteq \tau_j$ and $\mathcal{R}(\tau) = P^{a,b}_{j'}$, any j', we may conclude that $\{a', b'\} \neq \{a, b\}$. If $a' \neq a$, then $\mathcal{R}(\tau_j^\frown\langle \mathrm{f}\rangle)$ enumerates $y \in W_v$, contrary to assumption. Therefore $a = a'$: we can exclude the subcase $b = b'$, because otherwise $P^j_{a,b}$ would be cancelled along the true path, by the way requirements are assigned to nodes of the tree, and the fact that in this case we would have $j' < j$. Thus we are left to consider the case $a = a'$ and $b \neq b'$. If d remains active at all future stages, then y cannot be equivalent to any even number by Lemma 3.8. Otherwise, y collapses with a' or b'. In either case, it cannot in the future collapse with b, since all three of a', b', b are the least elements of their equivalence classes and $b \notin \{a', b'\}$. □

This concludes the proof of the theorem. □

It is proved in [1] that the class of u.e.i. ceers is properly contained in the class of e.i. ceers (by showing that there is an e.i. ceer that is not universal, whereas all u.e.i. ceers are universal). This conclusion is also a consequence of the previous theorem:

Corollary 3.11. *The u.e.i. ceers form a proper subclass of the e.i. ceers.*

Proof. The claim follows immediately by the fact that the index set of the u.e.i. ceers is Σ^0_3, whereas the index set of the e.i. ceers is Π^0_4 complete. □

4. The complexity of $\leqslant_c$ itself

An obvious generalization of computable reducibility from equivalence relations to pre-orders is the following: Given pre-orders R, S on the natural numbers, we say that R is *computably reducible* (or, simply, *reducible*) to S (notation: $R \leqslant_c S$) if there is a computable function f such that, for all x, y, $x\ R\ y$ if and only if $f(x)\ S\ f(y)$. Recently Ianovski, Miller, Nies and Ng [13] have used this reducibility to classify the complexity of several pre-orders which appear in mathematics and computability theory. For instance they show that the pre-order $\leqslant$, where $i \leqslant j$ if $W_i \leqslant_T W_j$, is Σ^0_4 complete.

In this section we prove that the reducibility $\leqslant_c$ on ceers induces a Σ^0_3 complete pre-order on numbers, where we write $i \leqslant_c j$ if $R_i \leqslant_c R_j$. This will follow from the next result, which in turn shows that the pre-ordering relation $\leqslant_1$ on numbers induced by 1-reducibility on c.e. sets (for which we write $i \leqslant_1 j$ if $W_i \leqslant_1 W_j$) is Σ^0_3 complete.

Theorem 4.1. *$\leqslant_1$ is a Σ^0_3 complete pre-order: in fact, for any given Σ^0_3 pre-order $\preceq$, there is a computable function f so that $W_{f(i)}$ is infinite for all i and*

$$(\forall i, j)[i \preceq j \Leftrightarrow W_{f(i)} \leqslant_1 W_{f(j)}].$$

Proof. It is straightforward to check that $\leqslant_1$ is Σ^0_3. Let $\preceq$ be a Σ^0_3 complete pre-order. We construct a uniform enumeration of V_a for each a as follows. Since $\preceq$ is Σ^0_3, as in the proof of Theorem 2.1, we can fix a recursive g so that

$$a \preceq b \Leftrightarrow (\exists k)[W_{g(a,b,k)} = \omega].$$

Requirements and their strategies. We have requirements:

$$Q^k_{ij} : W_{g(i,j,k)} = \omega \Rightarrow V_i \leqslant_1 V_j;$$
$$P^k_{ij} : (\forall l \leqslant k)[W_{g(i,j,l)} \neq \omega] \Rightarrow [\varphi_k \text{ does not } m\text{-reduce } V_i \text{ to } V_j];$$
$$I^k_i : \text{the set } V_i \text{ contains at least } k \text{ elements.}$$

Let us fix a priority ordering on the requirements. We now outline the strategies to meet the requirements.

Q-requirements. A Q^k_{ij}-requirement builds a computable set $A^k_{i,j}$ as follows: whenever $\min(\omega \smallsetminus W_{g(i,j,k)})$ increases, it adds a new element a to $A^k_{i,j}$. At such stages, if this is the m^{th} element (i.e., $a = a^k_{i,j}(m)$, where we write $a^k_{i,j}(n)$ for the n^{th} element of $A^k_{i,j}$), and some $n < m$ is enumerated into V_i, then the strategy enumerates $a^k_{i,j}(n)$ into V_j. As such, if there are infinitely many stages where $\min(\omega \smallsetminus W_{g(i,j,k)})$ increases (and no higher priority requirement ruins the coding), then $n \mapsto a^k_{i,j}(n)$ is a 1-reduction of V_i into V_j.

P-requirements. A P^k_{ij}-requirement acts as follows: to diagonalize and ensure that φ_k is not an m-reduction, we pick x larger than any element mentioned before. We wait for $\varphi_k(x)$ to converge. If it converges to an element which lies already in V_j, then we restrain x out of V_i. If it converges to an element not restrained out of V_j by any higher priority requirement, we enumerate $\varphi_k(x)$ into V_j and do not enumerate x into V_i (again, we place a restraint against this). We now suppose that $\varphi_k(x)$ is restrained out of V_j for a higher-priority requirement: suppose it is restrained due to being a witness chosen for a higher priority P-requirement. Then $P^k_{i,j}$ simply enumerates x into V_i. If, later, $\varphi_k(x)$ is enumerated into V_j, then that higher-priority P-requirement will have injured P^k_{ij}, which we allow. Now suppose $\varphi_k(x)$ is restrained due to being in the set $A^{k'}_{i',j}$ for a higher-priority $Q^{k'}_{i',j}$-requirement. Suppose it is the n^{th} element of the set $A^{k'}_{i',j}$, i.e., $\varphi_k(x) = a^{k'}_{i'j}(n)$, and $n \neq x$ or $i' \neq i$. We then put $\varphi_k(x)$ into V_j and n into $V_{i'}$, and we restrain x out of V_i. In a subsequent paragraph we will analyze in more detail how P^k_{ij} interacts with several higher priority requirements, and how to deal with the case $n = x$ and $i' = i$.

I-requirements. An I^k_i requirement simply selects new unrestrained elements and enumerates them into V_i to ensure V_i has size at least k.

The environments. At stage s of the construction, we use several parameters. A Q-requirement $Q^k_{i,j}$ uses the parameters $A^k_{i,j}(s)$, $a^k_{i,j}(n,s)$, approximating respectively the set $A^k_{i,j}$ and the witness, coding whether or not n is in V_i, as in the informal description of the strategy for $Q^k_{i,j}$; in other words, the mapping $n \mapsto a^k_{i,j}(n,s)$ approximates a computable function that 1-reduces V_i to V_j. After the last initialization of $Q^k_{i,j}$ (if eventually it stops being re-initialized), whenever we define $a^k_{i,j}(m,s)$, for some m, then this will be also the last value $a^k_{i,j}(m) = a^k_{i,j}(m,s)$. Notice that without loss of generality we may assume

$$n < a^k_{i,j}(n,s).$$

A P-requirement $P^k_{i,j}$ uses the parameter $x^k_{i,j}(s)$, which approximates the witness x, as described in the above description of the strategy for $P^k_{i,j}$. For each i,j,k, $P^k_{i,j}$ also uses a parameter $S^k_{i,j}(s)$, which is a finite set of numbers representing the restraint that these numbers not enter V_i. For every i, in the construction below we build V_i in stages, so that $\{V_{i,s} \mid s \in \omega\}$ is a computable approximation to V_i.

Interaction of $P^k_{i,j}$ with more than one requirement. We now need to analyze in detail what happens when we want to act for $P^k_{i,j}$ at a stage when $\varphi_k(x)$, with $x = x^k_{i,j}$, has not as yet been enumerated into V_j, and in fact is restrained out of V_j for a higher-priority requirement $\mathcal{R}$. Assume that $\varphi_k(x)$ converges to, say, y. If $\mathcal{R} = P^{k_0}_{j,i_0}$, for some i_0, k_0, and we have that $y = x^{k_0}_{j,i_0}$, then, as already observed, the conflict is just solved by priority: we enumerate x in V_i, and if $\mathcal{R}$ acts, then $\mathcal{R}$ initializes $P^k_{i,j}$.

The problematic case is when there are j_1, k'_0, and y_1, such that $\mathcal{R} = Q^{k'_0}_{j_1,j}$, and $y = a^{k'_0}_{j_1,j}(y_1)$: then we are able to act as desired, i.e. enumerate y into V_j, but at the same time keeping correctness of $a^{k'_0}_{j_1,h}(y_1)$, only if there is no restraint in enumerating also y_1 into V_{j_1}.

Now in turn, a restraint on y_1 can have been put either by a higher priority $P^{k_1}_{j_1,i_1}$, if $y_1 = x^{k_1}_{j_1,i_1}$, but then again the conflict is solved, as above, by priority; or, y_1 is restrained by a higher priority $Q^{k'_1}_{j_2,j_1}$, if y_1 is of the form $y_1 = a^{k'_1}_{j_2,j_1}(y_2)$.

This suggests the following definition:

Definition 4.2. Define the sequence $y_0, y_1, \dots, y_h, \dots$ by steps:

Step 0: Let $y_0 = y$, and $j_0 = j$.

Step 1: If there is no restraint on y_0, or there are unique i_0, k_0 such that $y_0 = x^{k_0}_{j_0,i_0}$, then y_1 is undefined; otherwise there exist unique j_1, k'_0, y_1 such that $y_0 = a^{k'_0}_{j_1,j_0}(y_1)$;

Step $h+1$: If there is no restraint on y_h, or there are unique i_h, k_h such that $y_h = x^{k_h}_{j_h,i_h}$ then y_{h+1} is undefined; otherwise there exist unique j_{h+1}, k'_h, y_{h+1} such that $y_h = a^{k'_h}_{j_{h+1},j_h}(y_{h+1})$.

Notice that at each step of the above inductive definition, the various disjuncts are exclusive: this claim (and the claims on uniqueness of j_h, i_h, k_h, k'_h) are justified (see Lemma 4.4) by the fact that strategies for different requirements use disjoint sets of witnesses and numbers.

Lemma 4.3. *The sequence $y_0, y_1, \dots, y_h, \dots$ is finite.*

Proof. For every r, if

$$y_r = a^{k'_r}_{j_{r+1},j_h}(y_{r+1})$$

then $y_{r+1} < y_r$. Thus the sequence must terminate. □

As currently $y \notin V_j$, and assuming correctness of the various functions $a^{k'_r}_{j_r,i_r}(\_)$ relative to higher priority requirements, we have that, for every r, $y_r \notin V_{j_r}$. So the strategy for $P^k_{i,j}$ in relation to restraints posed by higher priority requirements is the following:

(1) if the last entry of the sequence is y_h with $y_h \in S^{k'}_{j',i'}$ where $P^{k'}_{j',i'}$ has higher priority, then enumerate $x^k_{i,j}$ into V_i; we have $x^k_{i,j} \in V_i$, but $y = \varphi_k(x^k_{i,j}) \notin V_j$, unless $P^{k_h}_{j_h,i_h}$ acts and places y_h into V_{j_h}, but in this case all requirements of lower priority than $P^{k'}_{j',i'}$, including $P^k_{i,j}$, are initialized;

(2) if the last entry of the sequence is y_{h+1} with $y_h = a^{k'_h}_{j_{h+1},j_h}(y_{h+1})$ where y_{h+1} is not restrained by higher priority requirements and either $j_{h+1} \neq i$ or $y_{h+1} \neq x^k_{i,j}$, then enumerate each y_r with $r \leqslant h+1$ into V_{j_r}. We have, as desired, $y = \varphi_k(x^k_{i,j}) \in V_j$, but $x^k_{i,j} \notin V_i$; our action has not injured the higher priority requirements (in this case, only Q-requirements) since all relative 1-reductions have been corrected, having (for all $r \leqslant h$)

$$y_{r+1} \in V_{j_{r+1}} \Leftrightarrow a^{k_r}_{j_{r+1},j_r}(y_{r+1}) = y_r \in V_{j_r}.$$

In this case, we keep $x^k_{i,j}$ in $S^k_{i,j}$ to restrain lower priority requirements from ever causing $x^k_{i,j}$ to enter V_i.

(3) if the last entry of the sequence is y_{h+1} with $y_h = a^{k'_h}_{j_{h+1},j_h}(y_{h+1})$ where $j_{h+1} = i$ and $y_{h+1} = x^k_{i,j}$, then we cannot keep $x^k_{i,j}$ out of V_i while enumerating y into V_j, due to higher priority Q-requirements. In this case, $P^k_{i,j}$ adds $x^k_{i,j}$ to $S^k_{i,j}$ and then *unassigns* $x^k_{i,j}$ (and will thus choose a new $x^k_{i,j}$ when acting next). We will argue below, using the fact that $\preceq$ is a pre-order, that if $P^k_{i,j}$ is injured infinitely often in this way, then $i \preceq j$.

Construction. At stage $s+1$ we may enumerate new elements into some of the sets $\{V_{i,s} : i \in \omega\}$, thus obtaining their new approximations $\{V_{i,s+1} : i \in \omega\}$. We may also update the definition of some of the parameters. It is understood that if V_i, or a parameter, is not updated then its value is the same as at the previous stage.

At the end of a given stage s, we may *initialize* a requirement $\mathcal{R}$: For this, if $\mathcal{R} = Q^k_{i,j}$, then we set $A^k_{i,j}(s) = \varnothing$, and each $a^k_{i,j}(n,s)$ to be undefined; if $\mathcal{R} = P^k_{i,j}$, then we set $x^k_{i,j}(s)$ to be undefined and $S^k_{i,j}(s) = \varnothing$.

We say that a requirement $\mathcal{R}$ *requires attention* at stage $s > 0$, if $\mathcal{R}$ has not acted since last being initialized, or

(1) $\mathcal{R} = Q^k_{i,j}$ and s is $\langle i,j,k\rangle$-*expansionary*, i.e., $\min(\omega \backslash W_{g(i,j,k),s}) > \min(\omega \smallsetminus W_{g(i,j,k),\ell})$ where ℓ is the last stage where $Q^k_{i,j}$ acted; or
(2) $\mathcal{R} = P^k_{i,j}$ and either $x^k_{i,j}(s)$ is not defined or $\varphi_{k,s}(x^k_{i,j}(s))$ converges and

$$x^k_{i,j}(s) \in V_{i,s} \Leftrightarrow \varphi_{k,s}(x^k_{i,j}(s)) \in V_{j,s}.$$

At odd stages, we take care of P-requirements and Q-requirements. At nonzero even stages, we take care of the I-requirements.

Stage 0. Initialize all requirements.

Stage $2s+1$. Let $\mathcal{R}$ be the least P- or Q-requirement that requires attention. (Notice that cofinitely many such requirements have never acted.) We say that $\mathcal{R}$ *acts* at $2s+1$. For simplicity in the following, when writing down the various parameters, we do not explicitly mention the stage s.

(1) If $\mathcal{R} = Q^k_{i,j}$, then pick a new element a and place it into $A^k_{i,j}$: if a is the m-th element of $A^k_{i,j}$ in order of magnitude then define $a = a^k_{i,j}(m)$. For all $n < m$, if $n \in V_i$, then enumerate $a^k_{i,j}(n)$ into V_j.
(2) If $\mathcal{R} = P^k_{i,j}$ then
  (a) if $x^k_{i,j}$ is not defined, then define it to be a new element and add $x^k_{i,j}$ to $S^k_{i,j}$;
  (b) if $\varphi_k(x^k_{i,j})$ converges and $\varphi_k(x^k_{i,j}) \notin V_j$, then consider the sequence $y_0, y_1, \ldots, y_h, \ldots$ of Definition 4.2 (approximated at stage $2s+1$):
    (i) if the last entry of the sequence is $y_h = x^{k'}_{i',j'}$ with $y_h \in S^{k'}_{i',j'}$ where $P^{k'}_{i',j'}$ has higher priority, then enumerate $x^k_{i,j}$ into V_i, remove $x^k_{i,j}$ from $S^k_{i,j}$, and initialize all lower priority requirements;
    (ii) if the last entry of the sequence is y_{h+1} with $y_h = a^{k'_h}_{j_{h+1},j_h}(y_{h+1})$ where $j_{h+1} \neq i$ or $y_{h+1} \neq x^k_{i,j}$, then enumerate each y_r with $r \leqslant h+1$ into V_{j_r} and initialize all lower priority requirements;
    (iii) if the last entry of the sequence is y_{h+1} with $y_h = a^{k'_h}_{j_{h+1},j_h}(y_{h+1})$ where $j_{h+1} = i$ and $y_{h+1} = x^k_{i,j}$, then unassign $x^k_{i,j}$.

Go to Stage $2s+2$.

Stage $2s+2$**.** If $s = \langle i, k\rangle$, and V_i has less than k elements, then choose new numbers and enumerate them into V_i, so that the set has at least k elements.

This ends the construction.

Verification. It is left to verify that the construction works.

The following Lemma observes that in case (2bi), there is never any injury to enumerating $x^k_{i,j}$ into V_i and in case (2bii), there is never any injury to enumerating the y_r into V_{j_r}.

Lemma 4.4. *For any* i, j, j', k, k'*, if* $(j, k) \neq (j', k')$*, then* $x^k_{i,j}$ *is never in* $S^{k'}_{i,j'}$*. There is never an element* $a^k_{i,j}(y)$ *in* $S^{k'}_{j,j'}$ *for any* i, j, j', k, k'*.*

Proof. Each time $x^k_{i,j}$ is chosen, it is chosen to be a new number, and a number enters $S^{k'}_{i,j'}$ only after it has already been $x^{k'}_{i,j'}$. Each time $a^k_{i,j}(y)$ is chosen and each time $x^{k'}_{j,j'}$ is chosen, they are chosen to be new numbers, and no number enters $S^{k'}_{j,j'}$ unless it has already been $x^{k'}_{j,j'}$. □

Lemma 4.5. *No* P*-requirement initializes lower-priority requirements infinitely often.*

Proof. Let $\mathcal{R}$ be a P-requirement. Suppose, by induction, none of the higher-priority P-requirement initializes lower-priority requirements infinitely often. Let s then be a stage after which $\mathcal{R}$ is never initialized by a higher-priority requirement. If, after stage s, $\mathcal{R}$ ever initializes lower-priority requirements, it is through case (2bi) or (2bii). In either case, then $\mathcal{R}$ never acts again, so it can initialize lower-priority requirements at most once after stage s. □

Lemma 4.6. *If* $i \npreceq j$ *then* $P^k_{i,j}$ *is satisfied.*

Proof. Let s be a stage when $P^k_{i,j}$ is never initialized by a higher-priority requirement after stage s. We first argue that $\mathcal{R}$ cannot be initialized via (2biii) infinitely many times. Suppose otherwise. Then, each time it is initialized, consider the sequence $j_0, j_1, \dots, j_{h+1}$ where $j_{h+1} = i$. Let $a_0, a_1, \dots, a_n$ be a *simple* sub-path (i.e., if j_m and j_n are equal, then we replace the sequence $j_0, \dots, j_m, \dots, j_n, \dots, j_{h+1}$ by the sequence $j_0, \dots, j_m, j_{n+1}, \dots, j_{h+1}$, and repeat this algorithm until all the elements of the sequence are distinct). By the pigeonhole principle, for infinitely many initializations, this sequence $a_0, \dots, a_n$ is the same. But then, the requirements $Q^{k_m}_{a_{m+1},a_m}$ are acting infinitely often. Thus, using that $\preceq$ is a preorder, $a_{m+1} \preceq a_m$ for each $m \leqslant n$, and thus $i \preceq j$.

Thus, we can consider a stage $t > s$ such that $P^k_{i,j}$ is never initialized after stage t. Let $x = x^k_{i,j}$ at some stage after t. This is the final value of $x^k_{i,j}$. Subsequently, either $\varphi_k(x^k_{i,j})$ diverges, in which case $P^k_{i,j}$ does not act anymore, and is satisfied as φ_k is not total; or, $\varphi_k(x^k_{i,j})$ converges. In this latter case, it either never acts, in which case $\varphi_k(x^k_{i,j}) \in V_j$, but since $x^k_{i,j} \in S^k_{i,j}$, we have that $x^k_{i,j} \notin V_i$, so $P^k_{i,j}$ is satisfied; or it acts once more through (2bi), in which case $x^k_{i,j} \in V_i$, but $\varphi_k(x^k_{i,j}) \notin V_j$; or it acts through (2bii): in this case we get $x^k_{i,j} \notin V_i$, and $\varphi_k(x^k_{i,j}) \in V_j$. In all cases, $P^k_{i,j}$ is satisfied. □

Lemma 4.7. *If* $i \npreceq j$*, then* V_i *does not* m*-reduce to* V_j*.*

Proof. By Lemma 4.6, every $P^k_{i,j}$ is satisfied. □

Lemma 4.8. *If* $i \preceq j$*, then* $V_i \leqslant_1 V_j$*.*

Proof. Let k be least number such that $W_{g(i,j,k)} = \omega$. By Lemma 4.5, every P-requirement of priority higher than $Q^k_{i,j}$ initializes $Q^k_{i,j}$ only finitely often. After the last time $Q^k_{i,j}$ is initialized, every time $Q^k_{i,j}$ acts, it defines more and more values of the coding function $a^k_{i,j}(\_)$, and keeps it correct as a 1-reducibility, by putting $a^k_{i,j}(n)$ into V_j if and only if $n \in V_i$. □

Lemma 4.9. *For every pair i, k, the requirement I^k_i is satisfied.*

Proof. The proof is trivial. □

□

We are now ready to show that the pre-order $\leqslant_c$ on indices of ceers is Σ^0_3 complete.

Corollary 4.10. *$\leqslant_c$ is a Σ^0_3 complete pre-order.*

Proof. It is straightforward to check that $\leqslant_c$ is Σ^0_3. Since for infinite c.e. sets X, Y, $R_X \leqslant_c R_Y$ if and only if $X \leqslant_1 Y$ (where R_X is the ceer where aR_Xb if and only if $a = b$ or $a, b \in X$. See e.g. [1, 4, 19, 6, 10]) the above reduction allows us to reduce $\preceq$ into $\leqslant_c$ as well. □

The following corollaries are immediate consequence of Theorem 4.1, the first of which appears in [7]:

Corollary 4.11 ([7]). *The equivalence relation $\equiv_1$ is a Σ^0_3 complete equivalence relation.*

Proof. Trivial by Theorem 4.1, since an equivalence relation is a symmetric pre-ordering relation. □

Corollary 4.12. *$\equiv_c$ is a Σ^0_3 complete equivalence relation.*

Proof. Trivial by Corollary 4.10. □

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, 2014.
[2] C. Bernardi and A. Sorbi. Classifying positive equivalence relations. *J. Symbolic Logic*, 48(3):529–538, 1983.
[3] J. Case. Periodicity in generations of automata. *Math. Syst. Th.*, 8:15–32, 1974.
[4] S. Coskey, J. D. Hamkins, and R. Miller. The hierarchy of equivalence relations on the natural numbers. *Computability*, 1:15–38, 2012.
[5] Yu. L. Ershov. Positive equivalences. *Algebra and Logic*, 10(6):378–394, 1973.
[6] Yu. L. Ershov. *Theory of Numberings*. Nauka, Moscow, 1977.
[7] E. B. Fokina, S. D. Friedman, and A. Nies. Equivalence relations that are Σ^0_3 complete for computable reducibility - (Extended Abstract). In *WoLLIC*, pages 26–33, 2012.
[8] 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.
[9] E. Fokina, B. Khoussainov, P. Semukhin, and D. Turetsky. Linear orders realized by c.e. equivalence relations. *J. Symbolic Logic*, to appear.
[10] S. Gao and P. Gerdes. Computably enumerable equivalence realations. *Studia Logica*, 67:27–59, 2001.
[11] A. Gravruskin, S. Jain, A. Khoussainov, and F. Stephan. Graphs realized by r.e. equivalence relations. *Ann. Pure Appl. Logic*, 165(7):1263–1290, 2014.
[12] A. Gravuskin, A. Khoussainov, and F. Stephan. Reducibilities among equivalence relations induced by recursively enumerable structures. *Theoret. Comput. Sci.*, 612(25):137–152, 2016.

[13] E. Ianovski, R. Miller, A. Nies, and K. M Ng. Complexity of equivalence relations and preorders from computability theory. *J. Symbolic Logic*, 79(3):859–881, 2014.
[14] A. H. Lachlan. A note on positive equivalence relations. *Z. Math. Logik Grundlag. Math.*, 33:43–46, 1987.
[15] F. Montagna. Relative precomplete numerations and arithmetic. *J. Philosphical Logic*, 11:419–430, 1982.
[16] F. Montagna and A. Sorbi. Universal recursion theoretic properties of r.e. preordered structures. *J. Symbolic Logic*, 50(2):397–406, 1985.
[17] P. Odifreddi. *Classical Recursion Theory (Volume II)*, volume 143 of *Studies in Logic and the Foundations of Mathematics*. North-Holland, Amsterdam, 1999.
[18] M. B. Pour-El and S. Kripke. Deduction preserving isomorphisms between theories. *Fund. Math.*, 61:141–163, 1967.
[19] L. San Mauro. Forma e complessità. Uno studio dei gradi delle relazioni di equivalenza ricorsivamente enumerabili. Master's thesis, University of Siena, July 2011. In Italian.
[20] R. I. Soare. *Recursively Enumerable Sets and Degrees*. Perspectives in Mathematical Logic, Omega Series. Springer-Verlag, Heidelberg, 1987.

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/

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/