# NONEMBEDDINGS OF COMBINATORY ALGEBRAS

PATRICK LUTZ, PAUL SHAFER, AND SEBASTIAAN A. TERWIJN

Abstract. In the theory of combinatorial algebras, there is a sequence of embeddings between Kleene's second model, van Oosten's model, and Scott's graph model. We prove that none of these embeddings can be reversed. We also prove nonembedding results for the effective versions of these models, and in addition we discuss relativized embeddings. This answers several questions from the literature.

## 1. Introduction

We study embeddings and nonembeddings of partial combinatory algebras (pcas). Combinatory algebra is a formalism from logic that is closely related to lambda calculus. Both formalisms were conceived as theories of computation. In both formalisms, the application operator is total, since the application of one term to another always yields another term, even if this is computationally meaningless. For a theory of computation, this is not always desirable. It was long known that a treatment of combinatory algebra with a partial application operator is possible (as is clear from the discussion about completions of pcas in the early 1970s), but the first written account by Feferman [6] came only in 1975. For more about the relation between combinatory algebra and lambda calculus we refer the reader to Barendregt [2].

As might be expected, there are numerous connections between combinatory algebra and computability theory, yet another theory that emerged in the 1930s. Still, after the initial period where Kleene established fundamental results such as the equivalence of computability and $\lambda$-definability, the two areas developed largely independently. It took until Scott [16] before further significant connections were made, when he introduced models of the lambda calculus directly based on notions from computability theory. As Scott says in his paper (p183)

*Date*: 14 May 2026.

2010 *Mathematics Subject Classification.* 03B40, 03D25, 03D80.

*Key words and phrases.* partial combinatory algebra, embeddings, isomorphisms.

> $\lambda$-calculus and recursion theory make a good combination.

This statement serves as a motto of his paper, and it was borne out by much of the subsequent work. Indeed, there remain a lot of relations between the two areas to be explored, and the current paper may be seen as a contribution to this area. We use the notions and techniques from computability theory to say something about the relations between some of the standard models in combinatory algebra. We start with a quick review of the basic definitions.

A *partial applicative system (pas)* is a set $A$ with a partial binary application operator $\cdot$. For any two elements $a, b \in A$, we have an application $a \cdot b$, that may or may not be defined. If it is defined we write $a \cdot b \downarrow$. We also often omit the dot and simply write $ab$. We can consider all the terms built from elements of $A$, variables, and application. An applicative system is called a *partial combinatory algebra (pca)* if for every term $t(x_1, \ldots, x_n, x)$, $n \geqslant 0$, with free variables among $x_1, \ldots, x_n, x$, there exists $b \in \mathcal{A}$ such that for all $a_1, \ldots, a_n, a \in \mathcal{A}$,

(i) $ba_1 \cdots a_n \downarrow$,
(ii) $ba_1 \cdots a_n a \simeq t(a_1, \ldots, a_n, a)$.

Here $\simeq$ denotes the Kleene equality, meaning either both sides are defined and equal, or both are undefined. A combinatory algebra is simply a pca for which the application operator is total.

The defining property of pcas (called combinatory completeness) is precisely characterized by the existence of special combinators $k$ and $s$, just as in classical combinatory algebra. The following theorem is implicit in Feferman [6].

**Theorem 1.1.** *A pas $\mathcal{A}$ is a pca if and only if it has elements $k$ and $s$ with the following properties for all $a, b, c \in \mathcal{A}$:*

- *$ka\downarrow$ and $kab = a$,*
- *$sab\downarrow$ and $sabc \simeq ac(bc)$.*

Note that a pca with one element, and $k = s$, satisfies the definition of a pca. van Oosten [23] calls this pca trivial. From now on we will only consider nontrivial pcas. Beeson [3, p100] includes nontriviality in the definition by requiring $k \neq s$. A nontrivial pca is always infinite, and can represent the natural numbers and all partial computable functions on the natural numbers.

Since Theorem 1.1 characterizes pcas, it is sometimes taken as a definition instead of a theorem. However, we would like to stress that $k$ and $s$, though they always exist, need not be considered as part of

the signature of a pca. The reasons for this will become clear below when we talk about embeddings.

There is a large variety of examples of pcas in the literature. For example the lambda calculus itself is an example (where application is total). Further examples may be found in the monographs by Beeson [3], Odifreddi [13], van Oosten [23], and Longley and Normann [11]. Apart from their use in theoretical computer science, extensive use of pcas is being made in constructive mathematics, see Troelstra and van Dalen [21]. In particular, pcas may serve as a basis for models of constructive set theory, as for example in Rathjen [14] and Frittaion and Rathjen [7]. Here we only list the examples of pcas that play a role in this paper.

The most important of all examples is Kleene's first model $\mathcal{K}_1$, which has as its elements the natural numbers, and application defined by

$$n \cdot m = \Phi_n(m),$$

where $\Phi_n$ is the $n^{\text{th}}$ partial computable function. This is the setting of classical computability theory, and it underlines the need to consider partial applications. We can also relativize this to an arbitrary oracle $X$, thus obtaining the relativized pca $\mathcal{K}_1^X$.

Kleene's second model $\mathcal{K}_2$, introduced in Kleene and Vesley [9], is a pca defined on Baire space $\omega^\omega$. Application $f \cdot g$ in this model can be informally described as applying the continuous functional with code $f$ to the real $g$. The original coding of $\mathcal{K}_2$ is somewhat contrived, but it is essentially equivalent to

$$f \cdot g = \Phi_{f(0)}^{f \oplus g}, \tag{1}$$

where the application is understood to be defined if the RHS is total. This coding, used in Shafer and Terwijn [17], is considerably easier to work with. See the appendix of [8] for a proof (and precise statement) of the equivalence with the original coding.

We can relativize the definition of $\mathcal{K}_2$ to an arbitrary oracle $X$ by restricting the definition of $\mathcal{K}_2$ to $X$-computable sequences. This gives a countable pca $\mathcal{K}_2^X$ for every $X$. Note that, in contrast to $\mathcal{K}_1$, the pca $\mathcal{K}_2^X$ is actually smaller than $\mathcal{K}_2$, which is uncountable. Taking $X$ computable gives the pca $\mathcal{K}_2^{\text{eff}}$, consisting of all computable sequences, with application as in $\mathcal{K}_2$.

The van Oosten model $\mathcal{B}$, introduced in van Oosten [22], is a variant of $\mathcal{K}_2$ that is obtained by extending the domain to include partial functions. It can be described succinctly by taking the application in $\mathcal{K}_2$ as in (1) to also include partial functions $f$ and $g$. By definition, an oracle computation $\Phi_e^{f \oplus g}(x)$ for partial functions is undefined as soon

as it hits a query where the oracle is undefined. Like $\mathcal{K}_2$, the definition of $\mathcal{B}$ can be relativized to any oracle $X$ by restricting to the partial $X$-computable functions. This gives a pca $\mathcal{B}^X$ for any $X$, and when $X$ is computable we denote it by $\mathcal{B}^{\text{eff}}$. Note that unlike $\mathcal{K}_2$ and $\mathcal{K}_2^X$, both $\mathcal{B}$ and $\mathcal{B}^X$ are total combinatory algebras.

The graph model $\mathcal{G}$ was introduced in Scott [16], as a continuation and improvement on his earlier work on models of the $\lambda$-calculus.[1] $\mathcal{G}$ is a pca on the power set $\mathcal{P}(\omega)$, with application defined as

$$X \cdot Y = \big\{ n \in \omega \mid \exists u(\langle n, u\rangle \in X \wedge D_u \subseteq Y) \big\}, \tag{2}$$

where $D_u$ denotes the finite set with canonical code $u$. Note that $\mathcal{G}$ is in fact a total combinatory algebra. The connection with Rogers' notion of enumeration reducibility is that $Z \leqslant_e Y$ if and only if $Z = X \cdot Y$ for a c.e. set $X$.

Define $\mathcal{G}^X$ as the smallest sub-pca of $\mathcal{G}$ containing $X$ and all c.e. sets. For $X$ computable, we denote $\mathcal{G}^X$ by $\mathcal{E}$, which is the usual notation for the class of c.e. sets. So $\mathcal{E}$ is the combinatory algebra of c.e. sets with application as in $\mathcal{G}$.

**Proposition 1.2.** $\mathcal{G}^X = \{Z \in \mathcal{P}(\omega) \mid Z \leqslant_e X\}$.

*Proof.* The inclusion $\supseteq$ holds by the remarks about enumeration reducibility above. For $\subseteq$, we need that $X \cdot W_e$ can be written as $W_d \cdot X$ for some $d$. Intuitively it is clear that we can enumerate $X \cdot W_e$ given an enumeration for $X$. We give no formal proof, as this also follows from the results in Scott [16] quoted below. □

Note that $\mathcal{G}^X$ is different from the $X$-c.e. sets. For example, when $X$ is c.e. then $\mathcal{G}^X = \mathcal{E}$, but the $X$-c.e. sets are a larger class when $X$ is not computable.

**Definition 1.3.** Call a pca $\mathcal{A}$ *finitely generated* if there is a finite set $C \subseteq \mathcal{A}$ such that every element of $\mathcal{A}$ can be written as a closed term composed of elements of $C$.

For the record, we list the following examples, most of which are known:

**Theorem 1.4.** *The following pcas are all finitely generated:*
(i) $\mathcal{K}_1$, $\mathcal{K}_2^{\text{eff}}$, $\mathcal{B}^{\text{eff}}$, $\mathcal{E}$,
(ii) $\mathcal{K}_1^X$, $\mathcal{K}_2^X$, $\mathcal{B}^X$, $\mathcal{G}^X$ *for any* $X$,

[1] Scott also gives credit to the earlier authors Myhill and Shepherdson, Rogers, and Plotkin. He suggested the name graph model for $\mathcal{G}$ (p155) to prevent that it would be called Scott's model, so it is ironic that people now commonly refer to it as Scott's graph model.

(iii) *term algebras such as CL and λ-calculus.*

*Proof.* Some of these examples, such as $\mathcal{K}_1^X$ and $\mathcal{B}^X$, have been used implicitly in [8]. Note that the examples in (i) are special cases of those in (ii).

$\mathcal{K}_1$ is finitely generated: Consider the code $e$ of a p.c. function such that $e \cdot e = 0$ and $e \cdot 0 =$ successor function. Then the terms just containing $e$ yield all $n \in \omega$, hence $e$ by itself generates $\mathcal{K}_1$. The same argument shows that $\mathcal{K}_1^X$ is also generated by one element.

That $\mathcal{B}^X$ is finitely generated was discussed in footnote 2 in [8]. Namely we can define the Church numerals using only the basic combinators $s$ and $k$, and using these we can define the functions $h_n(x) = n$ for every $n$. Then we choose $j \in \mathcal{B}^X$ such that

$$\mathcal{B}^X \models j \cdot h_n = \Phi_n^X \tag{3}$$

for every $n$.

That $\mathcal{K}_2^X$ is finitely generated follows from the argument for $\mathcal{B}^X$ just given. (Having the embedding $\mathcal{K}_2^X \hookrightarrow \mathcal{B}^X$ by itself is not enough for this.)[2] Namely, the elements $j$ and $h_n$ in (3) are in fact total, hence elements of $\mathcal{K}_2^X$, and since application in $\mathcal{K}_2^X$ is the same as in $\mathcal{B}^X$, for *total* $\Phi_n^X$ we also have $\mathcal{K}_2^X \models j \cdot h_n = \Phi_n^X$.

$\mathcal{G}^X$ is finitely generated: Scott [16, Theorem 3.5] showed that the finitely generated sub-pcas of $\mathcal{G}$ are precisely those of the form $\mathcal{G}^X$. (He also showed that $\mathcal{G}^X$ can be generated with only one generator.) This also gives another proof of Proposition 1.2.

That CL and the λ-calculus are finitely generated is well-known, cf. Barendregt [2]. □

Since every finitely generated pca is countable, we obviously have that uncountable pcas such as $\mathcal{K}_2$, $\mathcal{B}$, and $\mathcal{G}$ are not finitely generated. However, there are also examples of countable pcas that are not finitely generated.[3]

We now discuss embeddings of pcas. To distinguish applications in different pcas, we write $\mathcal{A} \models ab{\downarrow}$ if this application is defined in $\mathcal{A}$.

[2]Embedding $\mathcal{K}_2^X$ into $\mathcal{B}^X$ (which is finitely generated) does not give that $\mathcal{K}_2^X$ is finitely generated, since a non-finitely generated pca may embed into a finitely generated one. For example, consider an infinite increasing chain $X_i$ of Turing degrees. Then the union of $\mathcal{G}^{X_i}$ is not finitely generated, but by Scott [16] every countably generated sub-pca of $\mathcal{G}$ is included in a finitely generated one, namely we can take the join of the generators.

[3]An example of this is the countable product $\mathcal{K}_1^\omega$, with elements the eventually constant sequences, and pointwise application as in $\mathcal{K}_1$.

**Definition 1.5.** For given pcas $\mathcal{A}$ and $\mathcal{B}$, an injection $f : \mathcal{A} \to \mathcal{B}$ is a *weak embedding* if for all $a, b \in \mathcal{A}$,

$$\mathcal{A} \models ab\downarrow \implies \mathcal{B} \models f(a)f(b)\downarrow = f(ab). \tag{4}$$

If $\mathcal{A}$ embeds into $\mathcal{B}$ in this way we write $\mathcal{A} \hookrightarrow \mathcal{B}$. If in addition to (4), for a specific choice of combinators $k$ and $s$ of $\mathcal{A}$, $f(k)$ and $f(s)$ serve as combinators for $\mathcal{B}$, we call $f$ a *strong embedding*.

Since by Theorem 1.1 the combinators $k$ and $s$ always exist in any pca, it may seem insignificant if we consider them to be part of the signature or not, and consequently the difference between weak and strong embeddings might seem equally small. However, it does make a big difference. This is apparent for example in the study of completions of pcas (i.e. embeddings of pcas into total combinatory algebras). If we do not consider $k$ and $s$ to be part of the signature then (by a result of Engeler [5]) every pca has a completion, but if we do then (by a result of Klop [10]) this is not the case. For more about completions see Terwijn [20]. Both notions of embedding have been studies in the literature, sometimes by the same author, for example Bethke [4]. Consequences of embeddings of pcas (both weak and strong) for the resulting realizability models are discussed in Swan [19]. In addition to this, there is also an even weaker notion called applicative morphism, introduced in Longley [12], that is useful in categorical contexts. In this paper we will focus on weak embeddings. As we will be proving nonembedding results, these will imply automatically the nonembedding results for strong embeddings.

Our notation is mostly standard. For computability theory we follow Odifreddi [13] and for partial combinatory algebra van Oosten [23]. For unexplained notation we refer to these monographs. In particular $\Phi_e^X$, $e \in \omega$, denotes the standard numbering of the partial computable functions relative to oracle $X$.

## 2. Nonembeddings

We consider weak embeddings of pcas. We have the following sequence of embeddings:

$$\mathcal{K}_1 \hookrightarrow \mathcal{K}_2 \hookrightarrow \mathcal{B} \hookrightarrow \mathcal{G}. \tag{5}$$

The first is by Shafer and Terwijn [17], the second is by inclusion, and the third is by Golov and Terwijn [8]. Of course we have that $\mathcal{K}_2 \not\hookrightarrow \mathcal{K}_1$, since $\mathcal{K}_1$ is countable and $\mathcal{K}_2$ is not, but it was open until now whether any of the other embeddings could be reversed. (This was posed as an open question in Golov and Terwijn [8, Question 7.4].) Below we show that $\mathcal{B} \not\hookrightarrow \mathcal{K}_2$ and that $\mathcal{G} \not\hookrightarrow \mathcal{B}$.

Since $\mathcal{G}$ is related to enumeration reducibility, and $\mathcal{K}_2$ to Turing reducibility, the question whether $\mathcal{G} \not\hookrightarrow \mathcal{K}_2$ seems related to embedding the enumeration degrees into the Turing degrees. It is known by a general result of Abraham and Shore [1] (cf. Odifreddi [13, p529]) that it is consistent (assuming CH) that the enumeration degrees are isomorphic to an initial segment of the Turing degrees (as an upper semilattice). Now any ideal $I$ in the Turing degrees $\mathcal{D}_T$ gives a pca with the application of $\mathcal{K}_2$. (E.g. the relativized versions $\mathcal{K}_2^X$ of $\mathcal{K}_2$ are of this form.) In particular this is true for the embedded version of the enumeration degrees $\mathcal{D}_e$. However, the application $\cdot_{\mathcal{K}_2}$ may not match the application in $\mathcal{G}$, so this does not necessarily give an embedding of $\mathcal{G}$ into $\mathcal{K}_2$, and indeed our results shows that such an embedding is impossible.

**Theorem 2.1.** $\mathcal{B} \not\hookrightarrow \mathcal{K}_2$.

*Proof.* Suppose for a contradiction that $F : \mathcal{B} \hookrightarrow \mathcal{K}_2$ is a weak embedding. So we have for all $f, g \in \mathcal{B}$,

$$\mathcal{B} \models f \cdot g = h \Longrightarrow \mathcal{K}_2 \models F(f) \cdot F(g)\downarrow = F(h).$$

Note that application in $\mathcal{B}$ is total, so $f \cdot g$ is always defined. Let $u \in \mathcal{B}$ be totally undefined, i.e. $u(x)\uparrow$ for all $x$, and let $h \in \mathcal{B}$ be an element different from $u$. We can define $a \in \mathcal{B}$ such that for all *total* $f, g$,

$$\mathcal{B} \models a \cdot f \cdot g = \begin{cases} u & \text{if } f = g \\ h & \text{if } f \neq g. \end{cases}$$

The result of $a \cdot f \cdot g$ for $f, g$ nontotal is immaterial.

Since $F$ is injective, we have $F(u) \neq F(h)$, and therefore there are $n$ and $y_0 \neq y_1$ such that $F(u)(n) = y_0$ and $F(h)(n) = y_1$.

For every total $f \in \mathcal{B}$ we have

$$\mathcal{K}_2 \models (F(a) \cdot F(f) \cdot F(f))(n) = y_0.$$

Hence for every total $f \in \mathcal{B}$ there exists a finite initial segment $\sigma_f \sqsubset F(f)$ such that

$$\mathcal{K}_2 \models (F(a) \cdot \sigma_f \cdot \sigma_f)(n) = y_0.$$

(By this we mean that for any $g_0, g_1 \sqsupset \sigma_f$, $\mathcal{K}_2 \models (F(a) \cdot g_0 \cdot g_1)(n) = y_0$.) By counting there exist total $f, g \in \mathcal{B}$ such that $f \neq g$ and $\sigma_f = \sigma_g$. Then we have $(F(a) \cdot \sigma_f \cdot \sigma_g)(n) = y_0$ (since $\sigma_f = \sigma_g$). But $f \neq g$, hence $F(a) \cdot F(f) \cdot F(g) = F(h)$, and $F(h)(n) = y_1$, a contradiction. □

**Theorem 2.2.** $\mathcal{G} \not\hookrightarrow \mathcal{B}$.

*Proof.* Suppose for a contradiction that $F : \mathcal{G} \hookrightarrow \mathcal{B}$ is a weak embedding, so that for all $X, Y, Z \in \mathcal{G}$,

$$\mathcal{G} \models X \cdot Y = Z \Longrightarrow \mathcal{B} \models F(X) \cdot F(Y) = F(Z).$$

Note that both $\mathcal{G}$ and $\mathcal{B}$ are total, so that the applications are always defined.

Call the embedding $F$ *monotone* if

$$\forall A \subseteq B \, \forall n \, (F(A)(n)\!\downarrow \implies F(B)(n)\!\downarrow = F(A)(n)).$$

In the rest of the proof we treat the two cases where $F$ is or is not monotone separately.

*Case 1.* Suppose that $F$ is not monotone. This case of the proof uses an idea similar to that of Theorem 2.1. Since $F$ is not monotone there exist sets $A \subsetneq B$ and $n \in \omega$ such that $F(A)(n)\!\downarrow \not\simeq F(B)(n)$. Fix such $A, B$ and $n$, and suppose $F(A)(n)\!\downarrow = y$.

We can easily define an enumeration operator $C \in \mathcal{G}$ such that for all $X, Y \in \mathcal{G}$,

$$X = Y \implies \mathcal{G} \models C \cdot (X \oplus \overline{X}) \cdot (Y \oplus \overline{Y}) = A,$$
$$X \neq Y \implies \mathcal{G} \models C \cdot (X \oplus \overline{X}) \cdot (Y \oplus \overline{Y}) = B.$$

Since $F$ is an embedding, we have

$$X = Y \iff \mathcal{B} \models (F(C) \cdot F(X \oplus \overline{X}) \cdot F(Y \oplus \overline{Y}))(n)\!\downarrow = y.$$

In particular we have

$$\mathcal{B} \models (F(C) \cdot F(X \oplus \overline{X}) \cdot F(X \oplus \overline{X}))(n)\!\downarrow = y$$

for every $X$. Hence for each $X$ there is a finite initial segment[4] $\sigma_X \sqsubset F(X \oplus \overline{X})$ such that for all $g_0, g_1 \sqsupset \sigma_X$,

$$(F(C) \cdot g_0 \cdot g_1)(n)\!\downarrow = y.$$

By counting we see that there exist $X \neq Y$ such that $\sigma_X = \sigma_Y$. We then have

$$(F(C) \cdot F(X \oplus \overline{X}) \cdot F(Y \oplus \overline{Y}))(n)\!\downarrow = y$$

because $\sigma_X = \sigma_Y \sqsubset F(X \oplus \overline{X}), F(Y \oplus \overline{Y})$. On the other hand, since $X \neq Y$ we have

$$(F(C) \cdot F(X \oplus \overline{X}) \cdot F(Y \oplus \overline{Y}))(n) = F(B)(n) \not\simeq y.$$

Thus we have a contradiction.

*Case 2.* Suppose that $F$ is monotone. Let $R \in \mathcal{G}$ be such that for all sets $X$ and $Y$,

- if $0 \notin X \cup Y$ then $R \cdot X \cdot Y = \{0\}$,
- if $0 \in X \cup Y$ then $R \cdot X \cdot Y = \{0, 1\}$.

[4]Since $F(X \oplus \overline{X})$ is a partial function, the initial segment $\sigma_X$ is a finite partial function. For a partial function $f$, by a finite initial segment $\sigma \sqsubset f$ we mean a finite partial function $\sigma$ such that for every $n$, $\sigma(n)\!\downarrow$ implies $f(n)\!\downarrow = \sigma(n)$ and also $\sigma(n)\!\uparrow$ implies $f(n)\!\uparrow$.

It is easy to see that such an enumeration operator $R$ can be defined in $\mathcal{G}$ (or even in the effective version $\mathcal{E}$).

Now let $X \subseteq Y$ be sets such that $0 \notin X$ and $0 \in Y$. Then $R \cdot X \cdot Y = R \cdot Y \cdot X = \{0, 1\}$ and $R \cdot X \cdot X = \{0\}$. Hence in $\mathcal{B}$ we have $F(R) \cdot F(X) \cdot F(Y) = F(R) \cdot F(Y) \cdot F(X)$, and by monotonicity (and injectivity) of $F$ there exists $n$ in the domain of $F(R) \cdot F(X) \cdot F(Y)$ that is not in the domain of $F(R) \cdot F(X) \cdot F(X)$. Now consider the undefined computation $(F(R) \cdot F(X) \cdot F(X))(n)$. This computation in $\mathcal{B}$ can be undefined in one of two ways: It can be undefined because of a query to an undefined spot in the oracle $F(R)$ or one of the copies of $F(X)$, or all the oracle queries in the computation are defined, but the computation still does not converge. We claim that at least one of the copies of $F(X)$ is not queried on any input where it is undefined. This is clear if the computation does not query any of the three components in an undefined spot, and if the computation queries either $F(R)$ or one of the copies $F(X)$ on an undefined input, then the computation is immediately undefined for this reason, hence the other copy $F(X)$ is never queried on an undefined input. Suppose the first copy of $F(X)$ is never queried on an undefined input. Then by monotonicity, we can substitute $F(Y)$ for $F(X)$ to obtain

$$(F(R) \cdot F(X) \cdot F(X))(n) = (F(R) \cdot F(Y) \cdot F(X))(n),$$

and the latter computation is defined on $n$, contradicting that the first is undefined. If the second copy of $F(X)$ is never queried on an undefined input we reach a contradiction in the same way. □

## 3. Relativized embeddings

For any $X$, the embeddings (5) restrict to:

$$\mathcal{K}_1^X \hookrightarrow \mathcal{K}_2^X \hookrightarrow \mathcal{B}^X \hookrightarrow \mathcal{G}^{X \oplus \overline{X}}.$$

In particular, for $X$ computable we have

$$\mathcal{K}_1 \hookrightarrow \mathcal{K}_2^{\text{eff}} \hookrightarrow \mathcal{B}^{\text{eff}} \hookrightarrow \mathcal{E}.$$

By Golov and Terwijn [8] we know that

$$\mathcal{B}^X \not\hookrightarrow \mathcal{K}_2^X \not\hookrightarrow \mathcal{K}_1^X$$

for any $X$, and also that $\mathcal{E} \not\hookrightarrow \mathcal{K}_2^{\text{eff}}$. Until now it was open whether $\mathcal{E} \not\hookrightarrow \mathcal{B}^{\text{eff}}$. We prove this in Theorem 3.1 below. Note that this is the effective analog of Theorem 2.2. Also note that we do have embeddings $\mathcal{B}^X \hookrightarrow \mathcal{K}_2$ for every $X$, since $\mathcal{B}^X$ is countable and every countable pca embeds into $\mathcal{K}_2$ ([8, Corollary 6.2]). However, it is not the case that for $X \leqslant_T Y$ the embedding of $\mathcal{B}^Y$ extends that of $\mathcal{B}^X$, so the embeddings

of $\mathcal{B}^X$ do not amalgamate to an embedding of $\mathcal{B}$ into $\mathcal{K}_2$, which by Theorem 2.1 is indeed impossible.

**Theorem 3.1.** $\mathcal{E} \not\hookrightarrow \mathcal{B}^{\text{eff}}$.

*Proof.* Suppose for a contradiction that $F : \mathcal{E} \hookrightarrow \mathcal{B}^{\text{eff}}$ is a weak embedding, so that for all $X, Y, Z \in \mathcal{E}$,

$$\mathcal{E} \models X \cdot Y = Z \Longrightarrow \mathcal{B}^{\text{eff}} \models F(X) \cdot F(Y) = F(Z). \tag{6}$$

Since $\mathcal{E}$ and $\mathcal{B}^{\text{eff}}$ are total, all applications are always defined. Although this effective version of Theorem 2.2 is different, if only for cardinality reasons (in the previous proof we used a counting argument), we make the same case distinction as before. So we call the embedding $F$ *monotone* if

$$\forall A \subseteq B \, \forall n \, (F(A)(n)\!\downarrow \Longrightarrow F(B)(n)\!\downarrow = F(A)(n)).$$

In the rest of the proof we treat the two cases where $F$ is or is not monotone separately.

*Case 1.* Suppose that $F$ is not monotone. Then there exist c.e. sets $A, B \in \mathcal{E}$ with $A \subseteq B$ and $x \in \omega$ such that $F(A)(x)\!\downarrow \not\simeq F(B)(x)$. Suppose $F(A)(x)\!\downarrow = y$.

Now we can, using that $\mathcal{E}$ is finitely generated, show that $\emptyset'$ is decidable. Let $d$ be a computable function such that

$$W_{d(n)} = \begin{cases} A & \text{if } n \notin \emptyset' \\ B & \text{if } n \in \emptyset' \end{cases}$$

for every $n$. By Theorem 1.4, $\mathcal{G}^X$ is finitely generated for every $X$. In particular, $\mathcal{E}$ is finitely generated, say by generators $G_1, \ldots, G_k$. (In fact, Scott showed that one generator suffices.) In fact, for every $n$ we can effectively compute a term $t_n(G_1, \ldots, G_k)$ composed entirely of generators of $\mathcal{E}$ such that $\mathcal{E} \models t_n(G_1, \ldots, G_k) = W_n$. From this we can compute a code of $F(W_{d(n)})$ in $\mathcal{B}^{\text{eff}}$ for every $n$. Namely, for every $n$ we have $\mathcal{E} \models t_{d(n)}(G_1, \ldots, G_k) = W_{d(n)}$. By (6) we then have that the term $t_{d(n)}(F(G_1), \ldots, F(G_k))$ denotes $F(W_{d(n)})$ in $\mathcal{B}^{\text{eff}}$. So even if $F$ is noneffective, all we need are codes for the images of the generators $F(G_1), \ldots, F(G_k)$. If $n \notin \emptyset'$ we have $F(W_{d(n)})(x)\!\downarrow = y$, so we can decide whether $n \in \emptyset'$ by searching until either $n \in \emptyset'$ or $F(W_{d(n)})(x)\!\downarrow = y$.

*Case 2.* Suppose that $F$ is monotone. This case is in fact identical to the case in the previous proof that $\mathcal{G} \not\hookrightarrow \mathcal{B}$ (Theorem 2.2). We only need to observe that the enumeration operator $R$ defined there, with the property that $R \cdot X \cdot Y = \{0\}$ if $0 \notin X \cup Y$ and $R \cdot X \cdot Y = \{0, 1\}$ if

$0 \in X \cup Y$, can indeed be chosen to be in $\mathcal{E}$. The rest of the argument is the same as before. □

**Theorem 3.2.** *$\mathcal{G}^{X \oplus \overline{X}} \not\to \mathcal{B}^X$ for any $X$.*

*Proof.* This is the relativization of $\mathcal{E} \not\to \mathcal{B}^{\text{eff}}$. It is straightforward to check that the proof of the latter relativizes. In the first part of the proof, where $F$ is not monotone, we have to replace $\emptyset'$ by $X'$, and use that $\mathcal{G}^X$ is finitely generated by Theorem 1.4. The second part of the proof, where $F$ is monotone, is purely combinatorial and can be copied almost verbatim. □

## 4. Embedding $\mathcal{K}_2^X$ into $\mathcal{K}_1^Y$

Let $f : \mathcal{A} \to \mathcal{B}$ be an embedding from pca $\mathcal{A}$ into pca $\mathcal{B}$, and let $a, b \in \mathcal{A}$. If $\mathcal{A} \models ab\downarrow$, then $\mathcal{B} \models f(a)f(b)\downarrow = f(ab)$. However, if $\mathcal{A} \models ab\uparrow$, then it may be that either $\mathcal{B} \models f(a)f(b)\uparrow$ or $\mathcal{B} \models f(a)f(b)\downarrow$. If also $\mathcal{B} \models f(a)f(b)\uparrow$ whenever $\mathcal{A} \models ab\uparrow$, we say that the embedding *$f$ preserves undefined*.

In this section, we consider the question of whether $\mathcal{K}_2^X$ embeds into $\mathcal{K}_1^Y$ for oracles $X$ and $Y$. We find that the answer depends on whether the embedding is required to preserve undefined. First, we show that $\mathcal{K}_2^X$ embeds into $\mathcal{K}_1^Y$ via an embedding that preserves undefined if and only if $X'' \leqslant_T Y$. Second, we show that $\mathcal{K}_2^X$ embeds into $\mathcal{K}_1^{X'}$ via an embedding that does not preserve undefined. This confirms the first part of [8, Conjecture 6.20]. By [17, Theorem 8.4] (see also [8, Section 4]), $K_1^X \hookrightarrow K_1^Y$ if and only if $X \leqslant_T Y$.[5] Therefore, if $X' \leqslant_T Y$, then $\mathcal{K}_2^X \hookrightarrow \mathcal{K}_1^{X'} \hookrightarrow \mathcal{K}_1^Y$, hence $\mathcal{K}_2^X \hookrightarrow \mathcal{K}_1^Y$. Conversely, if $\mathcal{K}_2^X \hookrightarrow \mathcal{K}_1^Y$, then $X' \leqslant_T Y$ by [8, Theorem 6.9]. Thus $\mathcal{K}_2^X \hookrightarrow \mathcal{K}_1^Y$ if and only if $X' \leqslant_T Y$.

**Lemma 4.1.** *For every oracle $X$, $\mathcal{K}_2^X$ embeds into $\mathcal{K}_1^{X''}$ via an embedding that preserves undefined.*

*Proof.* Using the recursion theorem, let $e$ be an index such that $\Phi_e^{X''}(a)$ produces another index $p = p(a) > 0$ for a machine $\Phi_{p(a)}^{X''}$ behaving as follows.

- $\Phi_{p(a)}^{X''}(0) = a$.
- For $n > 0$, $\Phi_{p(a)}^{X''}(n)$:
  - First runs $\Phi_n^{X''}(0)$. If it halts, lets $b = \Phi_n^{X''}(0)$.
  - Then uses $X''$ to determine whether $(\Phi_a^X) \cdot (\Phi_b^X)\downarrow$ (i.e., is total).

[5] [17, Theorem 8.4] considers embeddings that preserve undefined, but the same proof works for embeddings that do not necessarily preserve undefined.

– If $(\Phi_a^X) \cdot (\Phi_b^X)\downarrow$, uses $X''$ to find the least $c$ such that $\Phi_c^X = (\Phi_a^X) \cdot (\Phi_b^X)$ and returns $\Phi_e^{X''}(c)$.
– Here $\Phi_{p(a)}^{X''}(n)\uparrow$ if either $\Phi_n^{X''}(0)\uparrow$ or $(\Phi_a^X) \cdot (\Phi_b^X)\uparrow$.

For each $f \in \mathcal{K}_2^X$, let $a_f$ be its *least representative*, i.e., the least number $a_f$ such that $\Phi_{a_f}^X = f$. We show that $f \mapsto ea_f$ embeds $\mathcal{K}_2^X$ into $K_1^{X''}$ and preserves undefined.

Note that for any $a$, $ea0 = p(a) \cdot 0 = \Phi_{p(a)}^{X''}(0) = a$. Thus if $f \neq g$, then $a_f \neq a_g$, so $ea_f \neq ea_g$ because $ea_f0 = a_f \neq a_g = ea_g0$. Hence the mapping is injective.

Suppose that $fg\downarrow = h$ in $\mathcal{K}_2^X$. Then

$$(ea_f) \cdot (ea_g) = \Phi_{ea_f}^{X''}(ea_g) = \Phi_{p(a_f)}^{X''}(ea_g) = \Phi_e^{X''}(a_h) = ea_h.$$

For the third equality, $ea_g0 = a_g$ as explained above, and $(\Phi_{a_f}^X) \cdot (\Phi_{a_g}^X) = fg = h$ is total, so $a_h$ is the least $c$ such that $\Phi_c^X = (\Phi_{a_f}^X) \cdot (\Phi_{a_g}^X)$.

If instead $fg\uparrow$ in $\mathcal{K}_2^X$, then $(\Phi_{a_f}^X) \cdot (\Phi_{a_g}^X)\uparrow$, so $(ea_f) \cdot (ea_g) = \Phi_{ea_f}^{X''}(ea_g)\uparrow$ in $\mathcal{K}_1^{X''}$. □

**Lemma 4.2.** *For oracles $X$ and $Y$, if $\mathcal{K}_2^X$ embeds into $\mathcal{K}_1^Y$ via an embedding that preserves undefined, then $X'' \leqslant_T Y$.*

*Proof.* Let $F$ be an embedding of $\mathcal{K}_2^X$ into $\mathcal{K}_1^Y$ that preserves undefined. Then $X' \leqslant_T Y$ by [8, Theorem 6.9]. We use the embedding to show that $\mathrm{TOT}^X = \{e : \Phi_e^X \text{ is total}\}$ is c.e. relative to $Y$. We already know that the complement $\overline{\mathrm{TOT}^X}$ is c.e. relative to $Y$ because $X' \leqslant_T Y$. Hence it follows that $X'' \equiv_T \mathrm{TOT}^X \leqslant_T Y$.

Let $e_0$ be an index so that $\Phi_{e_0}^{f \oplus g} \simeq \Phi_{g(0)}^{g^-}$ for all $f$ and $g$, where $g^-(n) = g(n+1)$. Let $e_1$ be an index so that

$$\Phi_{e_1}^{f \oplus g}(n) = \begin{cases} g(0) + 1 & \text{if } n = 0 \\ g(n) & \text{if } n > 0 \end{cases}$$

for all $f$, $g$, and $n$.

Let $p = e_0{}^\frown 0^\omega$, $q = e_1{}^\frown 0^\omega$, and $g = 0^\frown X$. These are all members of $\mathcal{K}_2^X$. Observe that, for example, $q(q(qg))) = 3^\frown X$. Write $q^{\bar{n}}g$ to denote $q(q \cdots (qg))$, with $n$ right-to-left applications of $q$. Then $q^{\bar{n}}g = n^\frown X$. Thus

$$p \cdot (q^{\bar{n}}g) = \begin{cases} \Phi_n^X & \text{if } \Phi_n^X \text{ is total} \\ \uparrow & \text{otherwise.} \end{cases}$$

Let $a = F(p)$, $b = F(q)$, and $c = F(g)$. Then

$$b^{\bar{n}}c = F(q)^{\bar{n}}F(g) = F(q^{\bar{n}}g) = F(n^\frown X).$$

In particular, $b^{\bar{n}}c{\downarrow}$. Thus

$$a \cdot (b^{\bar{n}}c) = F(p)F(n^\frown X) = \begin{cases} F(p \cdot n^\frown X) = F(\Phi_n^X) & \text{if } \Phi_n^X \text{ is total} \\ \uparrow & \text{otherwise.} \end{cases}$$

because $F$ preserves undefined. Hence we have shown that $a \cdot (b^{\bar{n}}c){\downarrow}$ if and only if $\Phi_n^X$ is total. That is, $\mathrm{TOT}^X = \{n : \mathcal{K}_1^Y \models a \cdot (b^{\bar{n}}c){\downarrow}\}$, which is c.e. relative to $Y$. Thus $\mathrm{TOT}^X$ is c.e. relative to $Y$, so $X'' \leqslant_T Y$ as explained above. □

**Theorem 4.3.** *For oracles $X$ and $Y$, $\mathcal{K}_2^X$ embeds into $\mathcal{K}_1^Y$ via an embedding that preserves undefined if and only if $X'' \leqslant_T Y$.*

*Proof.* Suppose that $X'' \leqslant_T Y$. Then

$$\mathcal{K}_2^X \hookrightarrow \mathcal{K}_1^{X''} \hookrightarrow \mathcal{K}_1^Y$$

by embeddings that preserve undefined by Lemma 4.1 and the assumption $X'' \leqslant_T Y$.

Conversely, if $\mathcal{K}_2^X \hookrightarrow \mathcal{K}_1^Y$ by an embedding that preserves undefined, then $X'' \leqslant_T Y$ by Lemma 4.2. □

Now we embed $\mathcal{K}_2^X$ into $K_1^{X'}$ (by an embedding that cannot preserve undefined) by first embedding $\mathcal{K}_2^X$ into a binary version that we call $\mathcal{K}_{2,01}^X$ and then embedding $\mathcal{K}_{2,01}^X$ into $\mathcal{K}_1^{X'}$. First we define $\mathcal{K}_{2,01}^X$ and show that it is a pca that embeds $\mathcal{K}_2^X$.

**Definition 4.4.** Let $\mathcal{K}_{2,01}$ be the partial applicative system on $2^\omega$ with application

$$A \cdot B = \begin{cases} \uparrow & \text{if } A = 0^\omega \\ \Phi_e^{A \oplus B} & \text{if } e \text{ is least with } A(e) = 1. \end{cases}$$

Here we assume that machines are $\{0,1\}$-valued, say by taking outputs mod 2 as necessary.

For an oracle $X$, $\mathcal{K}_{2,01}^X$ is $\mathcal{K}_{2,01}$ restricted to binary sequences $A \leqslant_T X$.

**Proposition 4.5.** *$\mathcal{K}_{2,01}$ is a pca. For every oracle $X$, $\mathcal{K}_{2,01}^X$ is a pca.*

*Proof.* We prove that $\mathcal{K}_{2,01}$ is a pca. The proof that each $\mathcal{K}_{2,01}^X$ is a pca is the same. Indeed, we exhibit the combinators $k$ and $s$ in $\mathcal{K}_{2,01}$ similar to how one exhibits these combinators in the $f \cdot g = \Phi_{f(0)}^{f \oplus g}$ encoding of $\mathcal{K}_2$. These combinators are computable (they both have the form $0^{e-1}1^\frown 0^\omega$), so they are in every $\mathcal{K}_{2,01}^X$.

Let $e_0$ be an index so that for all $n \in \omega$ and $X, Y \in 2^\omega$, $\Phi_{e_0}^{(0^n 1^\frown X) \oplus Y} = X$. Let $e_1$ be an index so that for all $A, X \in 2^\omega$, $\Phi_{e_1}^{A \oplus X} = 0^{e_0 - 1}1^\frown X$.

Let $k = 0^{e_1-1}1^\frown 0^\omega$. Then for any $X, Y \in 2^\omega$

$$k \cdot X = (0^{e_1-1}1^\frown 0^\omega) \cdot X = \Phi_{e_1}^{(0^{e_1-1}1^\frown 0^\omega)\oplus X} = 0^{e_0-1}1^\frown X$$

and

$$k \cdot X \cdot Y = (0^{e_0-1}1^\frown X) \cdot Y = \Phi_{e_0}^{(0^{e_0-1}1^\frown X)\oplus Y} = X.$$

Thus $kX\!\downarrow$ and $kXY = X$.

Now let $e_0$ be an index so that for all $n \in \omega$ and $X, Y, Z \in 2^\omega$, $\Phi_{e_0}^{(0^n1^\frown(X\oplus Y))\oplus Z} \simeq XZ(YZ)$. Let $e_1$ be an index so that for all $n \in \omega$ and $X, Y \in 2^\omega$, $\Phi_{e_1}^{(0^n1^\frown X)\oplus Y} = 0^{e_0-1}1^\frown(X \oplus Y)$. Let $e_2$ be an index so that for all $A, X \in 2^\omega$, $\Phi_{e_2}^{A\oplus X} = 0^{e_1-1}1^\frown X$. Let $s = 0^{e_2-1}1^\frown 0^\omega$. Then for any $X, Y, Z \in 2^\omega$

$$s \cdot X = (0^{e_2-1}1^\frown 0^\omega) \cdot X = \Phi_{e_2}^{(0^{e_2-1}1^\frown 0^\omega)\oplus X} = 0^{e_1-1}1^\frown X$$

and

$$s \cdot X \cdot Y = (0^{e_1-1}1^\frown X) \cdot Y = \Phi_{e_1}^{(0^{e_1-1}1^\frown X)\oplus Y} = 0^{e_0-1}1^\frown(X \oplus Y)$$

and

$$s \cdot X \cdot Y \cdot Z = (0^{e_0-1}1^\frown(X \oplus Y)) \cdot Z = \Phi_{e_0}^{(0^{e_0-1}1^\frown(X\oplus Y))\oplus Z} \simeq XZ(YZ)$$

Thus $sXY\!\downarrow$ and $sXYZ \simeq XZ(YZ)$. So $\mathcal{K}_{2,01}$ is a pca. □

**Proposition 4.6.** *$\mathcal{K}_2$ embeds into $\mathcal{K}_{2,01}$ via an embedding that preserves undefined. Likewise, for every $X$, $\mathcal{K}_2^X$ embeds into $\mathcal{K}_{2,01}^X$ via an embedding that preserves undefined.*

*Proof.* We show that $\mathcal{K}_2$ embeds into $\mathcal{K}_{2,01}$ preserving undefined. The relativized version is the same.

For a partial function $\varphi\colon \omega \to \omega$, let $\mathrm{graph}(\varphi) = \{\langle x, y\rangle : x \in \mathrm{dom}(\varphi) \wedge \varphi(x) = y\}$. View any $X \in 2^\omega$ as the partial function $\psi_X\colon \omega \to \omega$ via

$$\psi_X(n) = \begin{cases} \text{the least } y \text{ such that } \langle n, y\rangle \in X & \text{if there is such a } y \\ \uparrow & \text{otherwise.} \end{cases}$$

Observe that $\psi_{\mathrm{graph}(\varphi)} = \varphi$ for any partial function $\varphi$.

Using the recursion theorem, let $e$ be an index so that for all $X, Y \in 2^\omega$

$$\Phi_e^{(0^{e-1}1^\frown X)\oplus(0^{e-1}1^\frown Y)} = \begin{cases} 0^{e-1}1^\frown \mathrm{graph}\left(\Phi_{\psi_X(0)}^{\psi_X\oplus\psi_Y}\right) & \text{if } \psi_X,\ \psi_Y, \text{ and } \Phi_{\psi_X(0)}^{\psi_X\oplus\psi_Y} \text{ are total} \\ \uparrow & \text{otherwise.} \end{cases}$$

Let $F(f) = 0^{e-1}1^\frown \operatorname{graph}(f)$. This embeds $\mathcal{K}_2$ into $\mathcal{K}_{2,01}$ and preserves undefined. Suppose that $f \cdot g\!\downarrow = \Phi_{f(0)}^{f\oplus g} = h$. Then

$$\begin{aligned} F(f) \cdot F(g) &= (0^{e-1}1^\frown \operatorname{graph}(f)) \cdot (0^{e-1}1^\frown \operatorname{graph}(g)) \\ &= \Phi_e^{(0^{e-1}1^\frown \operatorname{graph}(f))\oplus(0^{e-1}1^\frown \operatorname{graph}(g))} \\ &= 0^{e-1}1^\frown \operatorname{graph}\left(\Phi_{\psi_{\operatorname{graph}(f)}(0)}^{\psi_{\operatorname{graph}(f)}\oplus\psi_{\operatorname{graph}(g)}}\right) \\ &= 0^{e-1}1^\frown \operatorname{graph}\left(\Phi_{f(0)}^{f\oplus g}\right) \\ &= 0^{e-1}1^\frown \operatorname{graph}(h) \\ &= F(h). \end{aligned}$$

Likewise, if $f \cdot g = \Phi_{f(0)}^{f\oplus g}\!\uparrow$, then $\Phi_{\psi_{\operatorname{graph}(f)}(0)}^{\psi_{\operatorname{graph}(f)}\oplus\psi_{\operatorname{graph}(g)}} = \Phi_{f(0)}^{f\oplus g}\!\uparrow$, so $F(f) \cdot F(g)\!\uparrow$. □

Now we embed $\mathcal{K}_{2,01}^X$ into $\mathcal{K}_1^{X'}$ for a given oracle $X$. For this, recall that a *Scott set* is a collection $\mathcal{S} \subseteq 2^\omega$ that is closed under Turing reducibility, Turing join, and such that every infinite tree $T \subseteq 2^{<\omega}$ in $\mathcal{S}$ has a path in $\mathcal{S}$. That is, the Scott sets are the second-order parts of $\omega$-models of weak König's lemma. A Scott set $\mathcal{S}$ is a Turing ideal and therefore a sub-pca of $\mathcal{K}_{2,01}$. If $X \in \mathcal{S}$, then $\mathcal{K}_{2,01}^X$ is a sub-pca of $\mathcal{S}$.

If $Y$ has PA degree relative to $X$, then $Y$ computes a set $Z$ such that $\mathcal{S} = \{Z^{[i]} : i \in \omega\}$ is a Scott set containing $X$, where $Z^{[i]} = \{n : \langle i, n\rangle \in Z\}$ denotes the $i^{\text{th}}$ column of $Z$ ([15], see also [18, Theorem 4.11]). It thus follows from the low basis theorem that there is a set $Z$ with $Z' \leqslant_T X'$ such that $\{Z^{[i]} : i \in \omega\}$ is a Scott set containing $X$.

**Lemma 4.7.** *For every oracle $X$, $\mathcal{K}_{2,01}^X$ embeds into $\mathcal{K}_1^{X'}$.*

*Proof.* As discussed above, let $Z$ be such that $Z' \leqslant_T X'$ and $\mathcal{S} = \{Z^{[i]} : i \in \omega\}$ is a Scott set containing $X$. We embed $\mathcal{S}$ into $\mathcal{K}_1^{X'}$, thereby embedding $\mathcal{K}_{2,01}^X$ into $\mathcal{K}_1^{X'}$.

Scott sets are closed with respect to containing total $\{0,1\}$-valued extensions of partial $\{0,1\}$-valued functions. This means that for every $e$ and $i$, there is a $j$ such that $Z^{[j]}$ is a total extension of $\Phi_e^{Z^{[i]}}$:

$$\forall n(\Phi_e^{Z^{[i]}}(n)\!\downarrow \implies \Phi_e^{Z^{[i]}}(n) = Z^{[j]}(n))$$

Of course, if $\Phi_e^{Z^{[i]}}$ is total, then $Z^{[j]} = \Phi_e^{Z^{[i]}}$. Notice that "$Z^{[j]}$ is a total extension of $\Phi_e^{Z^{[i]}}$" is a $\Pi_1$ property of $e$, $i$, and $j$ relative to $Z$. Thus, given $e$ and $i$, $X' \geqslant_T Z'$ can find the least $j$ such that $Z^{[j]}$ is a total extension of $\Phi_e^{Z^{[i]}}$. Furthermore, for any $a$ and $b$, $Z^{[a]} \oplus Z^{[b]}$ is in $\mathcal{S}$ and

therefore some $Z^{[c]}$ is a total extension of $Z^{[a]} \cdot Z^{[b]}$. Thus the function

$$g(a,b) = \text{the least } c \text{ such that } Z^{[c]} \text{ is a total extension of } Z^{[a]} \cdot Z^{[b]}$$

is computable from $X'$. Note that if $(Z^{[a]} \cdot Z^{[b]}) \downarrow$, then $Z^{[g(a,b)]} = Z^{[a]} \cdot Z^{[b]}$.

From here on, the proof is similar to that of Lemma 4.1. Using the recursion theorem, let $e$ be an index such that $\Phi_e^{X'}(a)$ produces another index $p = p(a) > 0$ for a machine $\Phi_{p(a)}^{X'}$ behaving as follows.

- $\Phi_{p(a)}^{X'}(0) = a$.
- For $n > 0$, $\Phi_{p(a)}^{X'}(n)$:
  - First runs $\Phi_n^{X'}(0)$. If it halts, lets $b = \Phi_n^{X'}(0)$.
  - Returns $\Phi_e^{X'}(g(a,b))$.

Note that for any $a$, $ea0 = p(a) \cdot 0 = \Phi_{p(a)}^{X'}(0) = a$. Thus if $a \neq b$, then $ea \neq eb$ because $ea0 = a \neq b = eb0$.

Say that $a$ is the *least representative* of $Z^{[a]}$ if there is no $a' < a$ with $Z^{[a']} = Z^{[a]}$. We show that the mapping $Z^{[a]} \mapsto ea$ on the least representatives of the elements of $\mathcal{S}$ embeds $\mathcal{S}$ into $\mathcal{K}_1^{X'}$. This mapping is injective because if $Z^{[a]} \neq Z^{[b]}$, then $a \neq b$, so $ea \neq eb$.

Suppose that $a$, $b$, and $c$ are the least representatives of $Z^{[a]}$, $Z^{[b]}$, and $Z^{[c]}$ and that $(Z^{[a]} \cdot Z^{[b]})\downarrow = Z^{[c]}$. Then $(ea) \cdot (eb) = ec$ in $\mathcal{K}_1^{X'}$:

$$(ea) \cdot (eb) = \Phi_{ea}^{X'}(eb) = \Phi_{p(a)}^{X'}(eb) = \Phi_e^{X'}(g(a,b)) = \Phi_e^{X'}(c) = ec.$$

The third equality is because $eb0 = b$ as discussed above, so $\Phi_{p(a)}^{X'}(eb) = \Phi_e^{X'}(g(a,b))$. The fourth equality is because $g(a,b) = c$ since $Z^{[c]}$ is the least representative of $Z^{[a]} \cdot Z^{[b]}$. □

**Theorem 4.8.** *For oracles $X$ and $Y$, $\mathcal{K}_2^X$ embeds into $\mathcal{K}_1^Y$ if and only if $X' \leqslant_T Y$.*

*Proof.* Suppose that $X' \leqslant_T Y$. Then

$$\mathcal{K}_2^X \hookrightarrow \mathcal{K}_{2,01}^X \hookrightarrow \mathcal{K}_1^{X'} \hookrightarrow \mathcal{K}_1^Y.$$

The first embedding is by Proposition 4.6. The second embedding is by Lemma 4.7. The third embedding is because $X' \leqslant_T Y$.

Conversely, if $\mathcal{K}_2^X \hookrightarrow \mathcal{K}_1^Y$, then $X' \leqslant_T Y$ by [8, Theorem 6.9]. □

## Acknowledgments

We thank Michael Rathjen, Shuwei Wang, Andrew Brooke-Taylor, and Asaf Karagila for discussions about pcas and helpful comments about related set theoretic problems.

## References


1. Uri Abraham and Richard A. Shore, *Initial segments of the degrees of size* $\aleph_1$, Israel Journal of Mathematics **53** (1986), no. 1, 1–51.
2. Henk P. Barendregt, *The Lambda Calculus: Its Syntax and Semantics*, revised ed., Studies in Logic (London), vol. 40, College Publications, London, 2012, With addenda for the 6th imprinting, Mathematical Logic and Foundations.
3. Michael J. Beeson, *Foundations of Constructive Mathematics*, Ergebnisse der Mathematik und ihrer Grenzgebiete (3) [Results in Mathematics and Related Areas (3)], vol. 6, Springer-Verlag, Berlin, 1985, Metamathematical studies.
4. Ingemarie Bethke, *Notes on Partial Combinatory Algebras*, Ph.D. thesis, Universiteit van Amsterdam, 1988.
5. Erwin Engeler, *Algebras and combinators*, Algebra Universalis **13** (1981), no. 3, 389–392.
6. Solomon Feferman, *A language and axioms for explicit mathematics*, Algebra and Logic (Fourteenth Summer Res. Inst., Austral. Math. Soc., Monash Univ., Clayton, 1974), Lecture Notes in Math., vol. Vol. 450, Springer, Berlin-New York, 1975, pp. 87–139.
7. Emanuele Frittaion and Michael Rathjen, *Extensional realizability for intuitionistic set theory*, Journal of Logic and Computation **31** (2021), no. 2, 630–653.
8. Anton Golov and Sebastiaan A. Terwijn, *Embeddings between partial combinatory algebras*, Notre Dame Journal of Formal Logic **64** (2023), no. 1, 129–158.
9. Stephen Cole Kleene and Richard Eugene Vesley, *The Foundations of Intuitionistic Mathematics, Especially in Relation to Recursive Functions*, North-Holland Publishing Co., Amsterdam, 1965.
10. Jan Willem Klop, *Extending partial combinatory algebras*, Bulletin of the European Association for Theoretical Computer Science **16** (1982), 472–482.
11. John Longley and Dag Normann, *Higher-Order Computability*, Theory and Applications of Computability, Springer, Heidelberg, 2015.
12. John R. Longley, *Realizability Toposes and Language Semantics*, Ph.D. thesis, University of Edinburgh, 1994.
13. Piergiorgio Odifreddi, *Classical Recursion Theory*, Studies in Logic and the Foundations of Mathematics, vol. 125, North-Holland Publishing Co., Amsterdam, 1989.
14. Michael Rathjen, *Realizability for constructive Zermelo-Fraenkel set theory*, Logic Colloquium '03, Lect. Notes Log., vol. 24, Assoc. Symbol. Logic, La Jolla, CA, 2006, pp. 282–314.
15. Dana Scott, *Algebras of sets binumerable in complete extensions of arithmetic*, Proc. Sympos. Pure Math., Vol. V, Amer. Math. Soc., Providence, RI, 1962, pp. 117–121.
16. ———, *Lamba calculus and recursion theory*, Proceedings of the Third Scandinavian Logic Symposium (Univ. Uppsala, Uppsala, 1973), Stud. Logic Found. Math., vol. Vol. 82, North-Holland, Amsterdam-Oxford, 1975, pp. 154–193.
17. Paul Shafer and Sebastiaan A. Terwijn, *Ordinal analysis of partial combinatory algebras*, The Journal of Symbolic Logic **86** (2021), no. 3, 1154–1188.
18. Stephen G. Simpson, $\Pi^0_1$ *sets and models of* $\mathrm{WKL}_0$, Reverse Mathematics 2001, Lect. Notes Log., vol. 21, Assoc. Symbol. Logic, La Jolla, CA, 2005, pp. 352–378.

19. Andrew W. Swan, *Automorphisms of Partial Combinatory Algebras and Realizability Models of Constructive Set Theory*, Ph.D. thesis, University of Leeds, 2012.
20. Sebastiaan A. Terwijn, *Completions of Kleene's second model*, Logical Methods in Computer Science **21** (2025), no. 2, Paper No. 17, 10.
21. Anne S. Troelstra and Dirk van Dalen, *Constructivism in Mathematics. Vol. II*, Studies in Logic and the Foundations of Mathematics, vol. 123, North-Holland Publishing Co., Amsterdam, 1988.
22. Jaap van Oosten, *A combinatory algebra for sequential functionals of finite type*, Models and Computability (Leeds, 1997), London Math. Soc. Lecture Note Ser., vol. 259, Cambridge Univ. Press, Cambridge, 1999, pp. 389–405.
23. ———, *Realizability: An Introduction to its Categorical Side*, Studies in Logic and the Foundations of Mathematics, vol. 152, Elsevier B. V., Amsterdam, 2008.

(Patrick Lutz) University of Michigan, Department of Mathematics, 2074 East Hall, 530 Church Street, Ann Arbor, MI 48109-1043, U.S.A.
*Email address*: pglutz@umich.edu
*URL*: https://websites.umich.edu/~pglutz/

(Paul Shafer) School of Mathematics, University of Leeds, Leeds, LS2 9JT, United Kingdom
*Email address*: p.e.shafer@leeds.ac.uk
*URL*: https://peshafer.github.io

(Sebastiaan A. Terwijn) Radboud University Nijmegen, Department of Mathematics, P.O. Box 9010, 6500 GL Nijmegen, the Netherlands.
*Email address*: terwijn@math.ru.nl
*URL*: https://www.math.ru.nl/~terwijn/