Weihrauch problems as containers

Cécilia Pradic[0000−0002−1600−8846] and Ian Price[0009−0009−4112−3385]

Swansea University

Abstract. We note that Weihrauch problems can be regarded as containers over the category of projective represented spaces and that Weihrauch reductions correspond exactly to container morphisms. We also show that Bauer's extended Weihrauch degrees and the posetal reflection of containers over partition assemblies are equivalent. Using this characterization, we show how a number of operators over Weihrauch degrees, such as the composition product, also arise naturally from the abstract theory of polynomial functors.

Keywords: Weihrauch reducibility · Containers · Polynomial functors.

1 Introduction

Weihrauch reducibility allows to compare the computational strength of partial multi-valued functions over Baire space, which are thus called *Weihrauch problems*. Much like Reverse Mathematics, this can be used as a framework to gauge the relative power of certain Π^1_2 statements such as "for every sequence in $\{0,1\}^{\mathbb{N}}$, we can produce a cluster point" [7].

Concretely, a partial multi-valued function $f :\subseteq X \rightrightarrows Y$ is a map $f : X \to \mathcal{P}(Y)$. Its domain $\mathrm{dom}(f) \subseteq X$ is the set of elements such that $f(x) \neq \emptyset$. When $X = Y = \mathbb{N}^{\mathbb{N}}$, we call such f *Weihrauch problems* and define reduction between those as follows.

Definition 1 ([23][1]). *If f and g are two Weihrauch problems, f is said to be* Weihrauch reducible to *g if there exists a pair of partial type 2 computable maps (φ, ψ) such that φ is a map $\mathrm{dom}(f) \to \mathrm{dom}(g)$ and for every $i \in \mathrm{dom}(f)$ and $j \in g(\varphi(i))$, $\psi(i, j)$ is defined and belongs to $f(i)$.*

We write $f \leq_{\mathrm{W}} g$ when a reduction from f to g exists. Since Weihrauch reductions compose as depicted in Figure 1, Weihrauch problems and reductions form a preorder. Equivalence classes of problems are called Weihrauch degrees.

[1] Note that the official definition in other papers in the literature may differ in two ways: problems may allowed to range over arbitrary represented spaces and the definition of "being a reduction" might involve quantifying over sections of the target multi-valued function. The latter aspect does not matter in presence of $2^{\aleph_0}$-choice (otherwise it captures less reductions than the other definition) [8, Proposition 3.2]. For the former, the usage of arbitrary represented spaces comes with the allowance for (φ, ψ) to be multi-valued, which means a problem P is equivalent to its version defined on codes (which are assumed to be elements of Baire space).

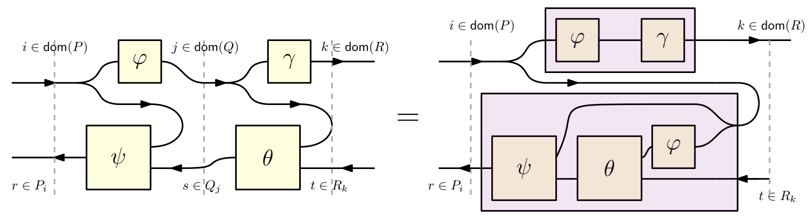


Fig. 1. String diagram representing the composition of Weihrauch reductions from P to Q and Q to R.

Example 1. We can encode the following as Weihrauch problems:

- LPO ("given a bit sequence, tell me if it has a 1"), defined by

$$\mathsf{LPO}(p) = \{0^\omega \mid p = 0^\omega\} \cup \{1^\omega \mid p \in \{0,1\}^{\mathbb{N}}, p \neq 0^\omega\}$$

- LPO' ("given a bit sequence, tell me if it has infinitely many 1s"), defined by

$$\mathsf{LPO}'(p) = \{0^\omega \mid p \in \{0,1\}^{\mathbb{N}}, \forall^\infty n.\ p_n = 0\} \cup \{1^\omega \mid p \in \{0,1\}^{\mathbb{N}}, \exists^\infty n.\ p_n = 1\}$$

- KL: "given an infinite finitely branching tree t, give me an infinite path through t" can also be encoded as a Weihrauch problem, modulo a standard embedding $2^{\mathbb{N}^{<\omega}} \hookrightarrow \mathbb{N}^{\mathbb{N}}$.
- WKL is the restriction of KL to trees containing only binary words.

We have $\mathsf{LPO} <_{\mathrm{W}} \mathsf{LPO}'$, $\mathsf{LPO} <_{\mathrm{W}} \mathsf{KL}$ and $\mathsf{WKL} <_{\mathrm{W}} \mathsf{KL}$. KL and LPO' are incomparable, as well as LPO and WKL.

The Weihrauch degrees actually form a distributive lattice, where meets and joins can be regarded as natural operators on problems: the join $P \sqcup Q$ allows to ask a question either to P or Q and get the relevant answer, while $P \sqcap Q$ requires to ask one question to each and get only one answer. Many other natural operators have been introduced on the Weihrauch lattice, including a parallel product $P \times Q$ (ask questions to both P and Q, get both answers), a composition operator $P \star Q$ (ask a question to Q and then, depending on the answer you got, a question to P) and many others, including residuals and fixpoints of other operators [8, Definition 1.2].

The definition of a number these operators often does not seem to depend much on the specifics of type 2 computability. They also tend to satisfy nice equational theories whose soundness can solely be established through abstract nonsense. It has also been remarked that Weihrauch reducibility shares a likeness with Gödel's Dialectica interpretation [8,34, resp. §1.9 and 6]. This hints that "the category of Weihrauch reductions" should be constructible by applying a functorial construction $\mathsf{Cat} \to \mathsf{Cat}$ (where Cat is the category of all locally small categories) to a somewhat canonical category of spaces and type 2 computable functions between them.

In this paper, we observe that this construction has been known as the category of *containers* [1][2] over projective represented spaces (that is, the full subcategory of represented spaces containing all subspaces of $\mathbb{N}^{\mathbb{N}}$).

Theorem 1. *The Weihrauch degrees are isomorphic to the posetal reflection of the category of answerable[3] containers over the category of projective represented spaces and (type 2) computable functions.*

The proof consists mostly of unpacking definitions; let us try to give a rough idea of its content. Containers over a category $\mathcal{C}$ with pullbacks are bundles, that is, morphisms in $\mathcal{C}$. A morphism from the container $P : X \to U$ to $Q : Y \to V$ consists of a forward map from the base U of P to the base V of Q and a backward map $Y \times_V U \to X$ involving the total spaces X and Y. [4]. One can turn multi-valued functions f into the bundle $F : \{(x, y) \mid y \in f(x)\} \xrightarrow{\pi_1} \mathrm{dom}(f)$ to get half of the correspondence we want and check that this extends to a functor: Weihrauch reductions *are* morphisms between the containers built this way.

Example 2. One can check that the container corresponding to LPO is

$$\{(0^\omega, 0^\omega)\} \cup \{(p, 1^\omega) \mid p \in \{0,1\}^{\mathbb{N}} \setminus \{0^\omega\}\} \longrightarrow \{0,1\}^{\mathbb{N}}$$

It is equivalent (in the sense that there are morphisms both ways) to the surjective bundle $l : \mathbb{N} \twoheadrightarrow \mathbb{N}_\infty \subseteq 2^{\mathbb{N}}$ defined by $l(0) = 0^\omega$ and $l(n+1) = 0^n10^\omega$.

This correspondence applies not only to Weihrauch reducibility, but also to some closely connected variants. One is continuous Weihrauch reducibility, the notion obtained by replacing "computable" by "continuous" in Definition 1, for which the theorem can be adapted straightforwardly as follows.

Theorem 2. *The continuous Weihrauch degrees are isomorphic to the posetal reflection of the category of answerable containers over projective represented spaces and continuous functions.*

Another closely related notion are the extended Weihrauch degrees introduced in [6] by means of a more general notion of reduction between predicates in a topos. Again a correspondence can be made with containers over a relatively natural category of projectives, although this requires a bit of massaging.

Theorem 3. *Assuming the axiom of choice, the extended Weihrauch degrees over a filtered PCA* $(\mathbb{A}', \mathbb{A})$ *are isomorphic to the posetal reflection of the category of containers over partitioned assemblies* $\mathsf{pAsm}(\mathbb{A}', \mathbb{A})$.

---

[2] While the name container comes from research on generic programming, they sometimes also are known under different names in other communities. They may be called *polynomials* due to their correspondence with polynomial functors over locally cartesian closed categories [13].

[3] Meaning those that can be represented by surjections between carriers, or equivalently, those pullback-stable epimorphisms; see Definition 7.

[4] $Y \times_V U$ is the domain of the pullback of Q along the forward map; see Section 2.2 for a fully formal definition.

In making those folklore correspondences explicit, we hope among other things to link the literature on Weihrauch reducibility and containers. After providing the necessary background definitions (Section 2) and proving the theorems stated thus far (Section 3), we discuss some of the algebraic operators of the Weihrauch lattice in the context of the theory of containers (Section 4). We will only focus on the two lattice operations, the parallel product and the composition product. While we will obtain clean correspondences to known functorial constructions for the first three, we will hit a snag when it comes to the composition product. This is due to the fact that the categories of spaces we are taking containers over are only *weakly* locally cartesian closed. This means that the composition product will only be a quasi-functor, which complexifies its systematic study; we leave that for further work.

Related work The idea of regarding a bundle as a problem to be solved by finding one of its section is old one that predates the "container" terminology. For instance, Hirsch [15, Definition 3.4] defines an equivalent category to study the topological complexity of problems and reductions between those. This perspective also already appeared in the literature on Weihrauch problems (see for instance [19, Remark 2.8]), although most of the recent efforts we are aware of to "categorify" Weihrauch reducibility and operators on problems tend to use other tools instead.

One such natural categorical construction that captured Weihrauch degrees is the restriction of Bauer's extended Weihrauch problems [6] to objects that are actually Weihrauch problems, which are characterized as the $\neg\neg$-dense predicates over modest sets. Interestingly, Ahman and Bauer also linked generalized Weihrauch reducibility to containers [2], but by way of the more general notion of instance reducibility that works over families of truth values. They get back to generalized Weihrauch predicated by specializing their constructions to assemblies which do have a natural truth value object which is not partitioned while here we work directly with bundles of partitioned assemblies.

Trotta et. al. also formally linked (extended) Weihrauch reducibility with the Dialectica interpretation [34], which can be regarded bicompletions of fibrations by simple products and sums [16]. Aside from the fact that they work in a posetal setting throughout, it is interesting to note that the category of containers over $\mathcal{C}$ can be recovered by completing the terminal fibration over $\mathcal{C}$ by arbitrary products and then sums and taking the fiber over 1.

Pauly also studied a generic notion of reducibility that encompass Weihrauch reducibility, starting from categories of multivalued functions [30], in which he derived the lattice operators as well as finite parallelizations in a generic way. In contrast, we try to stay as far away as possible from multi-valued functions as a notion of morphism in this paper.

2 Categories for computable analysis

We review briefly some standard terminology from category theory and realizability that we will use in this paper. In the rest of this paper we will favor terminology from realizability rather than computable analysis for convenience as it allows us to state Theorem 3. Aside from the use of partial combinatory algebras, the main terminological differences are summarized in Figure 2. We assume that the reader is familiar with the notion of what are categories, functors, natural transformations, finite (co)limits and what is type-2 computability.

2.1 Categories for type-2 computability

Since we are going to be borrowing notions from realizability, we are going to use definition based on the notion of partial combinatory algebras (PCAs). Without going into details (which the interested reader can find in e.g. [36]), a PCA is given by a set $\mathbb{A}$ (its *carrier*) and a partial binary operation $\cdot : \mathbb{A}^2 \rightharpoonup \mathbb{A}$. The intuition is that elements of $\mathbb{A}$ are to be regarded as codes for programs and that $\cdot$ denotes function application; as such we simply call it application in the sequel. In addition to having this operation, a PCA $\mathbb{A}$ is required to have certain distinguished elements s and k satisfying certain equations, essentially so that the untyped λ-calculus is interpretable in $\mathbb{A}$.

Example 3. The PCA $\mathcal{K}_1$ (Kleene's first algebra) has carrier $\mathbb{N}$ and the application $n \cdot m$ is defined to be the output of running the nth Turing machine on m, otherwise is undefined.

One of the PCAs we will be concerned with will be $\mathcal{K}_2$ (Kleene's second algebra). The carrier of $\mathcal{K}_2$ is $\mathbb{N}^{\mathbb{N}}$, and the intuition is that we can regard elements of Baire space as partial functions $\mathbb{N}^{\mathbb{N}} \rightharpoonup \mathbb{N}$ as follows: up to squinting, we can regard elements of $\mathbb{N}^{\mathbb{N}}$ as countably branching trees whose leaves are labelled by natural numbers. One can then define the partial function represented by such a tree t as mapping $p \in \mathbb{N}$ to $n \in \mathbb{N}$ if and only if following the path p in t leads to a leaf labelled by n (if p is an infinite branch of t, the function is undefined). The obtained partial function is continuous and it can be shown that all partial continuous functions can be defined this way. The application of the PCA, which is meant to regard elements of t as maps $\mathbb{N}^{\mathbb{N}} \rightharpoonup \mathbb{N}^{\mathbb{N}}$ can be defined on this basis using $\mathbb{N}^{\mathbb{N}} \cong \mathbb{N}^{\mathbb{N}} \times \mathbb{N}$.

For the purpose of computable analysis, we need to use computable continuous functions in a setting where all elements of $\mathbb{N}^{\mathbb{N}}$ are in the background. For this we need the notion of a *filtered PCA* $(\mathbb{A}', \mathbb{A})$ [29, Definition 1.5], which is a pair where $\mathbb{A}$ is a PCA and $\mathbb{A}'$ is a subPCA of $\mathbb{A}$ (i.e. it is stable by $\cdot$ and contains the distinguished k and s, i.e., the interpretation of untyped λ-calculus with parameters in $\mathbb{A}'$ using the structure of $\mathbb{A}$ lands in $\mathbb{A}'$).

The one filtered PCA we shall use in this paper is $(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2)$ where $\mathcal{K}_2^{\mathrm{rec}}$ consists of the set of computable elements of $\mathbb{N}^{\mathbb{N}}$. It can be checked that they correspond, through applications, to functions computable by type-2 Turing machines. When defining a category $\mathcal{C}(\mathbb{A}', \mathbb{A})$ parameterized by a filtered PCA $(\mathbb{A}', \mathbb{A})$, we may write $\mathcal{C}(\mathbb{A})$ for $\mathcal{C}(\mathbb{A}, \mathbb{A})$ for brevity's sake.

With this machinery in place, we can start to define categories of sets represented by codes in PCAs. We begin with the more general notion we will encounter in this paper and decline the useful subcategories we will discuss.

Realizability	modest sets (Mod)	assemblies (Asm)	partitioned modest sets (pMod)
Computable analysis	represented spaces	multi-represented spaces	represented subspaces of $\mathbb{N}^{\mathbb{N}}$ (projective represented spaces)

Fig. 2. Categories used in realizability and their counterpart, once they are specialized to the filtered PCA $(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2)$, in computable analysis.

Definition 2. *An assembly over a PCA* $\mathbb{A}$ *is a pair* $(X, \Vdash_X)$ *where* X *is a set called the* carrier *and* $\Vdash_X \subseteq \mathbb{A} \times X$ *is a relation whose image is* X*.*

A map f *between the carriers of the assemblies* $(X, \Vdash_X)$ *and* $(Y, \Vdash_Y)$ *is* tracked *by a code* $\ulcorner f \urcorner \in \mathbb{A}$ *if, whenever* $e \Vdash_X x$*, we have* $\ulcorner f \urcorner \cdot e \Vdash_Y f(x)$*.*

Assuming $(\mathbb{A}', \mathbb{A})$ *is a filtered PCA, the category* $\mathsf{Asm}(\mathbb{A}', \mathbb{A})$ *has assemblies over* $\mathbb{A}$ *as objects and maps tracked by codes in* $\mathbb{A}'$ *as morphisms.*

Given an assembly A, write $|A|$ for its carrier.

Remark 1. An object of $\mathsf{Asm}(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2)$ is essentially the same thing as a multi-represented space [17, Remark 5.2] and the morphisms of $\mathsf{Asm}(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2)$ (respectively $\mathsf{Asm}(\mathcal{K}_2)$) are the computable maps between them.

Definition 3. *Given a filtered PCA* $(\mathbb{A}', \mathbb{A})$*, the category of modest sets* $\mathsf{Mod}(\mathbb{A}', \mathbb{A})$ *is the full subcategory of assemblies* $(X, \Vdash_X)$ *such that any element of* $\mathbb{A}$ *realizes at most one element (i.e.,* $e \Vdash_X x$ *and* $e \Vdash_X x'$ *imply* $x = x'$*).*

Remark 2. The objects of $\mathsf{Mod}(\mathcal{K}_2)$ are essentially *represented spaces.*

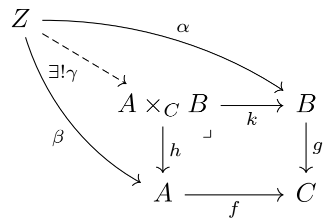
Definition 4. *Call an assembly* partitioned *when every of its element have at most one realizer. We write* $\mathsf{pMod}(\mathbb{A}', \mathbb{A})$ *and* $\mathsf{pAsm}(\mathbb{A}', \mathbb{A})$ *for the respective full subcategories of partitioned modest sets* $\mathsf{Mod}(\mathbb{A}', \mathbb{A})$ *and assemblies* $\mathsf{Asm}(\mathbb{A}', \mathbb{A})$*.*

Remark 3. The objects of $\mathsf{pMod}(\mathcal{K}_2)$ are essentially *represented subspaces* of $\mathbb{N}^{\mathbb{N}}$, which can also be characterized as the regular projective objects of $\mathsf{Mod}(\mathcal{K}_2)$ (Lemma 2).

2.2 Notations for category-theoretical notions

We assume familiarity with the notion of cartesian products and coproducts. We write $\langle f_1, f_2 \rangle : Z \to A_1 \times A_2$ for the pairing of $f_i : Z \to A_i$ $(i \in \{1, 2\})$ and $g : Z \to A_2$ and $\pi_i : A_1 \times A_2 \to A_i$ $(i \in \{1, 2\}$. Dually we write $[f_1, f_2] : A_1 + A_2 \to Z$ and $\mathsf{in}_i : A_i \to A_1 + A_2$ for the copairing and coprojections.

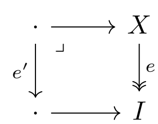
We will heavily use the notion of pullbacks throughout this paper as well as standard notation that involves them. A pullback square is a commuting square as found in the diagram below satisfying the following universal property: if there are morphisms α, β as depicted such that $f \circ \beta = g \circ \alpha$, then there is a unique γ such that $\alpha = k \circ \gamma$ and $\beta = h \circ \gamma$.



In this diagram $A \times_C B$ (together with the projections h and k) is a pullback of f and g. We use the notation around the top-left corner of the commuting square to indicate we intend this to be a pullback in a diagram. Pullbacks are only determined up to unique isomorphisms. Concretely, pullbacks in all categories we are interested in here can be built by taking the cartesian products of the domains of the maps f and g and restricting to the sets of pairs (a, b) with $f(a) = g(b)$. When we fix a choice of pullbacks for each pair of morphisms, we write $f^*(g)$ for the morphism corresponding to h (read "the pullback of g along f"); f^* actually extends to a functor ${}^{\mathcal{C}}/_{C} \to {}^{\mathcal{C}}/_{A}$ where ${}^{\mathcal{C}}/_{A}$ ("the slice over A") is the category whose objects are morphisms of $\mathcal{C}$ with codomain A and morphisms are maps between domains of objects that make the obvious triangle diagram commute.

We recall that an *epimorphism* is a morphism $f : A \to B$ such that, whenever we have $h, k : B \to Z$ with $h \circ f = k \circ f$, we actually know that $h = k$.

Definition 5. *An epimorphism $e : X \twoheadrightarrow I$ is called* pullback-stable *if for every pullback square*



e' is also an epimorphism.

In topological spaces (and a fortiori in represented spaces), the pullback-stable epimorphisms can be characterized as the maps that are surjective in Set. This does not cover all epimorphisms.

Example 4. $\mathbb{Q} \subseteq \mathbb{R}$ gives rise to an epimorphism due to the fact that all maps are continuous, but that epimorphism is not pullback-stable.

Finally, let us recall that preorders (and a fortiori posets) can be regarded as categories whose objects are elements of the preorder and with a unique morphism between two objects x and y if and only if $x \leq y$. Posets form a reflective subcategory of the category of small categories Cat, which means that the inclusion of posets into categories has a left adjoint functor $\mathcal{C} \mapsto {}^{\mathcal{C}}/_{\leftrightarrow}$. ${}^{\mathcal{C}}/_{\leftrightarrow}$ is the quotient of $\mathcal{C}$ where objects X and Y are equivalent when there exists morphisms $X \to Y$ and $Y \to X$.

3 Reducibility as container morphisms

3.1 A container is an internal family

At the intuitive level, a container in a category $\mathcal{C}$ is a family of objects of $\mathcal{C}$ indexed by an object of $\mathcal{C}$. Since objects of $\mathcal{C}$ are not necessarily sets, to make this formal, one needs to define what is a family *internal* to a category. This is done by regarding objects of $\mathcal{C}/_I$ as I-indexed families. The basic idea is that in Set, given a family $(A_i)_{i\in I}$, we can form the disjoint union $\sum_{i\in I} A_i$ and encode the family in this formalism by taking the projection $\sum_{i\in I} A_i \ \rightarrow\ I$. Conversely, to show that the categorical point of view captures something equivalent to the usual notion in Set, we can perform an inverse operation turning any set-theoretic map $f : X \to I$ into the family $(f^{-1}(i))_{i\in I}$.

A fundamental operation one may perform on families is *reindexing*: given a map of index sets $f : I \to J$ and a family over J, one may produce a family over I. Then the fundamental operation of reindexing families along a map $f : I \to J$ is given by taking a pullback.

Families of sets	Families internal to $\mathcal{C}$
Families over $J \to$ Families over I $(B_j)_{j\in J} \mapsto (B_{f(i)})_{i\in I}$	$\mathcal{C}/_J \to \mathcal{C}/_I$ $\begin{array}{ccccc} B & & B' & \longrightarrow & B \\ \downarrow{\scriptstyle\alpha} & \mapsto & {\scriptstyle f^*\alpha}\downarrow & \lrcorner & \downarrow{\scriptstyle\alpha} \\ J & & I & \longrightarrow & J \end{array}$

For any choice[5] of pullbacks, this yield functors $f^* : \mathcal{C}/_J \to \mathcal{C}/_I$ which we call *reindexing along* f. Those functor always have a left adjoint $\Sigma_f : \mathcal{C}/_J \to \mathcal{C}/_I$ given by precomposition that corresponds to *sums along* f. In Set, this corresponds to being able to take set-indexed disjoint unions: given a family $(A_i)_{i\in I}$, f allows us to consider it as a family of families over J we may define the family of families, $((A_i)_{i\in f^{-1}(j)})_{j\in J}$ and then perform the set-theoretic disjoint union component-wise.

Families of sets	Families internal to $\mathcal{C}$
Families over $I \to$ Families over J $(A_i)_{i\in I} \mapsto \left(\sum_{i\in f^{-1}(j)} A_i\right)_{j\in J}$	$\mathcal{C}/_I \to \mathcal{C}/_J$ $\begin{array}{ccc} & & A \\ A & & \downarrow{\scriptstyle\alpha} \\ \downarrow{\scriptstyle\alpha} & \mapsto & I \\ I & & \downarrow{\scriptstyle f} \\ & & J \end{array}$

The existence of right adjoints Π_f to reindexing functors f^* is however not always guaranteed for every category with pullbacks. It is the case if and only

[5] Only unique up to unique isomorphism in $\mathcal{C}/_J$.

if all slice categories of $\mathcal{C}$ are cartesian closed, in which case we call $\mathcal{C}$ *locally cartesian closed*. In such a case, Π_f corresponds to *products* and is right adjoint to f^*.

Families of sets	Families internal to $\mathcal{C}$
Families over $I \to$ Families over J $(A_i)_{i\in I} \mapsto \left(\prod_{i\in f^{-1}(j)} A_i\right)_{j\in J}$	$\mathcal{C}/_I \to \mathcal{C}/_J$ $\alpha \mapsto (f \circ \alpha)^f$

Having access to the functors $\Sigma_f \dashv f^* \dashv \Pi_f$ means that we can interpret closed type theoretic-expressions that involve equalities, sums and products over specified objects of $\mathcal{C}$ as objects of $\mathcal{C}$; open expressions containing variables are interpreted as families whose indexing objects correspond to the variables.

3.2 The category of containers: morphisms

As said previously, the objects of the category $\mathsf{Cont}(\mathcal{C})$ of containers over $\mathcal{C}$ will be morphisms in $\mathcal{C}$. Following terminology from [28], for such a container $P : X \to U$, we will call U the object of *positions* and X the object of *directions*. Now the important thing with containers is the notion of morphism that comes with them. In sufficiently structured categories, they correspond to strong natural transformations between functors induced by the containers [13, Theorem 2.12], but they also admit a more general low-level characterization that is easily seen to correspond to notions of reducibility[6]. With sets, assuming we have $Q : Y \to V$, $P_u = P^{-1}(u)$ and $Q_v = Q^{-1}(v)$, a morphism of container from P to Q is given by a pair of maps

$$\varphi : U \to V \qquad \text{and} \qquad \psi \in \prod_{u\in U} (Y_{\varphi(u)} \to X_u)$$

Note that the backward map ψ is here typed using dependent products, which only exist internally in locally cartesian closed categories. We adopt the more elementary equivalent definition which works in all categories with pullbacks.

Definition 6 ([13, (4)]). *A morphism representative from a container $P : X \to U$ to $Q : Y \to V$ is a pair (φ, ψ) making the following diagram commute, the rightmost square being a pullback*[7]*:*

$$\begin{array}{ccccc}
X & \xleftarrow{\;\psi\;} & \sum_{u\in U} Y_{\varphi(u)} & \longrightarrow & Y \\
{\scriptstyle P}\downarrow & & \downarrow & \lrcorner & \downarrow{\scriptstyle Q} \\
U & = & U & \xrightarrow[\varphi]{} & V
\end{array}$$

[6] And was also independently defined for such purposes, see e.g. [15, Definition 3.4].

[7] While we give a suggestive type-theoretic name for the relevant object, it is determined up to unique isomorphism. This is why we need to quotient morphism representatives; with chosen pullbacks, we could have dispensed with this step.

Two morphisms representatives (φ, ψ) and (φ', ψ') from P to Q are defined to be equivalent if and only if $\varphi = \varphi'$ and $\psi \circ \theta = \psi'$ when θ is the canonical isomorphism between the domain of ψ to the domain of ψ' that witnesses that they both are pullbacks of Q and φ. A morphism of containers $P \to Q$ is an equivalence class of morphism representatives.

Given a morphism (representative) (φ, ψ), we will call φ the forward map and ψ the backward map. Composition is defined essentially as suggested in Figure 1; more formally, it is determined as per the categorical diagram in Figure 3. We will call (φ, ψ) *horizontal* when ψ is the identity and *vertical* when φ is the identity.

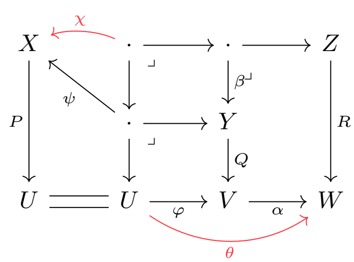


Fig. 3. We say that $(\theta, \chi) : P \to R$ is a composition of the morphism representatives $(\varphi, \psi) : P \to Q$ and $(\alpha, \beta) : Q \to R$. Implicit in Lemma 1 is that pairs of equivalent representative admit the same compositions.

Lemma 1. *Assume $\mathcal{C}$ is a category with pullbacks. Then we can form a category $\mathcal{C}$ where the objects are maps in $\mathcal{C}$, morphisms are given as in Definition 6, identities over $P : X \to U$ are represented by (id_U, id_X) and the composition is given as Figure 3.*

Proof. It can be checked by unpacking the definitions that $\mathsf{Cont}(\mathcal{C})$ is exactly the total category of the fibration $\mathrm{cod}^{\mathrm{op}}$ over $\mathcal{C}$, which is obtained by taking opposite of the codomain fibration over $\mathcal{C}$ (which exists because of pullbacks); see [33, §1 & §5] for details. □

3.3 Weihrauch problems are containers over partitioned modest sets

We are now almost ready to detail how to prove the main theorem, except for one thing: some of the containers in $\mathsf{Cont}(\mathsf{pMod}(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2))$ may represent families of subsets of $\mathbb{N}^{\mathbb{N}}$ that may have the empty set as a member. Weihrauch problems do not allow that, so we need to restrict the class of objects to rule out those containers.

Definition 7. *We call a container* answerable *if the underlying map is a pullback-stable epimorphism.*

As previously mentioned, in Asm and a fortiori pMod, the pullback-stable epimorphisms are surjections over the relevant carriers; this ensures each fiber is non-empty. It will be sometimes interesting in the sequel to also consider containers of $\mathsf{Cont}(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2)$ which are not answerable - so when needed we will refer to objects of (the posetal reflection of) $\mathsf{Cont}(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2)$ as *slightly extended Weihrauch problems (degrees).*

Theorem 1. *The Weihrauch degrees are isomorphic to the posetal reflection of the category of answerable*[8] *containers over the category of projective represented spaces and (type 2) computable functions.*

Proof. Recall that the category of projective represented spaces is isomorphic to $\mathsf{pMod}(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2)$. To each answerable container $P : X \to U$ over $\mathsf{pMod}(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2)$, we can map the P to the Weihrauch problem $\mathsf{w}(P) :\subseteq \mathbb{N}^{\mathbb{N}} \rightrightarrows \mathbb{N}^{\mathbb{N}}$ with $\mathsf{w}(P)(p) = \{q \in \mathbb{N}^{\mathbb{N}} \mid \exists x \in |X|.\ P(x) \Vdash_U p \wedge x \Vdash_X q\}$. For any other $Q : Y \to V$ and a reduction representative $(\varphi, \psi) : P \to Q$, we can find an equivalent one (φ, ψ') where the domain of ψ' is actually the set of pairs (u, y) with $Q(y) = \varphi(u)$ as it is a pullback. Then (φ, ψ') is literally a Weihrauch reduction from $\mathsf{w}(P)$ to $\mathsf{w}(Q)$.

In the other direction, given a Weihrauch problem $f :\subseteq \mathbb{N}^{\mathbb{N}} \rightrightarrows \mathbb{N}^{\mathbb{N}}$, we have a map $\mathsf{c}(f) : \sum_{u \in \mathrm{dom}(f)} f(u) \xrightarrow{\pi_1} \mathrm{dom}(f)$; we can regard its domain and codomain as carriers of partitioned modest sets by taking the trivial realizability relations $\langle e, e' \rangle \Vdash_{\mathrm{dom}(\mathsf{c}(f))} (e, e')$ and $e \Vdash_{\mathrm{dom}(f)} e$. By definition, if a reduction $f \leq_{\mathrm{W}} g$ is represented by a pair of computable maps (φ, ψ), we have $\varphi : \mathrm{dom}(f) \to \mathrm{dom}(g)$ and $\psi : \sum_{u \in \mathrm{dom}(f)} g(\varphi(u)) \to \bigcup_{u \in \mathrm{dom}(f)} f(u)$. Since $\psi(u, y)$ is meant to give an answer to u, by combining ψ with the projection $\sum_{u \in \mathrm{dom}(f)} g(\varphi(u)) \to \mathrm{dom}(f)$, we can form a map $\widetilde{\psi} : \sum_{u \in \mathrm{dom}(f)} g(\varphi(u)) \to \sum_{u \in \mathrm{dom}(f)} f(u)$ so that (φ, ψ) is a morphism representative $\mathsf{c}(f) \to \mathsf{c}(g)$.

Now, to conclude is suffices to show that $\mathsf{c}(\mathsf{w}(P)) \cong P$ and $\mathsf{w}(\mathsf{c}(f)) =_{\mathrm{W}} f$ for all P and f.

For the first part, for $P : X \to U$, because P is answerable, we do also have $\mathrm{dom}(\mathsf{w}(P)) = \{p \in \mathbb{N}^{\mathbb{N}} \mid \exists u \in |U|.\ p \Vdash_U u\}$; this is enough to derive an isomorphism $\theta : \mathrm{dom}(\mathsf{w}(P)) \cong U$ in $\mathsf{pMod}(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2)$ which is tracked by the code for the identity. Then we can easily check that $\mathsf{c}(\mathsf{w}(P))$ is a pullback of P along θ, hence (θ, id) and $(\theta^{-1}, \mathrm{id})$ represent an isomorphism $\mathsf{c}(\mathsf{w}(P)) \cong P$ as expected.

For the second part we have that $\mathsf{w}(\mathsf{c}(f))(p) = \{\langle p, x \rangle \mid x \in f(p)\}$ for every $p \in \mathbb{N}^{\mathbb{N}}$. So the forward component of both reductions we want can be taken to be the identity, the backward component of the reduction $f \leq_{\mathrm{W}} \mathsf{w}(\mathsf{c}(f))$ can

[8] Meaning those that can be represented by surjections between carriers, or equivalently, those pullback-stable epimorphisms; see Definition 7.

project away the first component of the output of $\mathsf{w}(\mathsf{c}(f))$ while the backward part of the converse can be the identity. □

We can note that in the proof above did not use computability aside from the fact type-2 computable maps correspond to morphisms in the category in question. As partial continuous maps are all representable in $\mathcal{K}_2$, Theorem 2 can be proven in the same way.

3.4 Extended Weihrauch degrees and containers over partitioned assemblies

The notion of extended Weihrauch predicates defined by Bauer [6] is a specialization of his notion of instance reducibility to realizability toposes built from filter-PCAs. The rationale is that, in the Kleene-Vesley topos [36, §4.5], a Weihrauch problem P can be translated to predicates $P'(i)$ (over some object of the topos, which corresponds to the inputs of P) such that $\forall i.\exists j.Q'(j) \Rightarrow P'(i)$ holds if and only if P is Weihrauch-reducible to Q. This is essentially because, since we are in a realizability topos, the truth of this statement is necessarily witnessed by a program which encodes the forward part of a reduction due to the $\forall\exists$ and a backwards part due to the $\Rightarrow$ part.

By unwinding the definition, the notion can be stated elementarily from the definition of a filter-PCA, so let us do that, assuming an arbitrary filter-PCA $(\mathbb{A}', \mathbb{A})$ is fixed for the rest of the subsection.

Definition 8 ([6, Definition 3.7], see also [20, Definition 5.7]). *An* extended Weihrauch predicate *is a map $p : \mathbb{A} \to \mathcal{P}(\mathcal{P}(\mathbb{A}))$. Its* support *is the set $\|p\| = \{r \in \mathbb{A} \mid p(r) \neq \emptyset\}$. p is said to be reducible to another extended Weihrauch predicate q when there exists codes $e_{\mathrm{fwd}}, e_{\mathrm{bwd}} \in \mathbb{A}'$ such that, for every $r \in \|p\|$:*

1. *$e_{\mathrm{fwd}} \cdot r$ is defined and belongs to $\|q\|$*
2. *for every $\theta \in p(r)$ there is $\xi \in q(e_{\mathrm{fwd}} \cdot r)$ such that $e_{\mathrm{bwd}} \cdot r \cdot s$ is defined and belongs to θ for any $s \in \xi$.*

For the sequel, let us restate 2, using the axiom of choice, as follows:

2' we have a function $f_r : p(r) \to q(e_{\mathrm{fwd}} \cdot r)$ such that $e_{\mathrm{bwd}} \cdot r \cdot s$ is defined and belongs to θ for any $s \in f(\theta)$.

With this version, it is helpful to think of $(e_{\mathrm{fwd}}, (f_r)_{r \in \|p\|})$ as a code for a forward reduction and e_{bwd} as a code for a backward reduction. In the case of extended Weihrauch predicate p coming from Weihrauch problems (or slightly extended Weihrauch problems), $p(r)$ will be always a (sub)singleton. In the general case, p corresponds to a binary partial multivalued function $\mathbb{A} \times \mathbb{A} \rightrightarrows \mathbb{A}$ where the first input corresponds to r, the second to an element of $p(r)$ and the forward part of the reduction is allowed to be an arbitrary set-theoretic map while the backward reduction cannot compute at all with the second input.

Example 5 (See [6]). An example of an extended Weihrauch predicate which does not match any Weihrauch degree is WLEM which is formally defined as the constant function mapping any $r \in \mathbb{A}$ to $\{\{\underline{0}\}, \{\underline{1}\}\}$, where $\underline{i}$ is the canonical code for the number i in $\mathbb{A}'$.

Assuming the axiom of choice, it is maximal for reducibility among all extended Weihrauch predicates p with $\bigcup p(r) \subseteq \{\underline{0}, \underline{1}\}$; clearly there is no such maximal Weihrauch problem.

With this definition, let us show the following theorem, which will justify calling objects of $\mathsf{pAsm}(\mathbb{A}', \mathbb{A})$ *extended Weihrauch problems.*

Theorem 3. *Assuming the axiom of choice, the extended Weihrauch degrees over a filtered PCA* $(\mathbb{A}', \mathbb{A})$ *are isomorphic to the posetal reflection of the category of containers over partitioned assemblies* $\mathsf{pAsm}(\mathbb{A}', \mathbb{A})$*.*

Note that the proof, as well as the rest of the section, uses liberally the axiom of choice.

Proof. First let us explain how to turn an extended Weihrauch predicate p into a container $\widehat{p}$. We first define the assemblies I_p and X_p of positions and directions of $\widehat{p}$ by taking

$$\begin{array}{ll} |X_p| = \{(r, \theta, s) \in \mathbb{A} \times \mathcal{P}(\mathbb{A}) \times \mathbb{A} \mid r \in \|p\|, \theta \in p(r), s \in \theta\} & \langle r, s\rangle \Vdash_{X_p} (r, \theta, s) \\ |I_p| \ = \{(r, \theta) \in \mathbb{A} \times \mathcal{P}(\mathbb{A}) \mid r \in \|p\|, \theta \in p(r)\} & r \Vdash_{I_p} (r, \theta) \end{array}$$

It is obvious those assemblies are partitioned. $\widehat{p}$ is then defined by the projection $(r, \theta, s) \mapsto (r, \theta)$, which is tracked by code of the first projection in $\mathbb{A}'$. Furthermore, a reduction $((e_{\mathrm{fwd}}, f), e_{\mathrm{bwd}})$ between extended Weihrauch predicates p and q induce a map $\widehat{p} \to \widehat{q}$ in $\mathsf{ContpAsm}(\mathbb{A}', \mathbb{A})$: the forward map $(r, \theta) \mapsto (e_{\mathrm{fwd}} \cdot r, f(\theta))$ is tracked by e_{fwd} and the backwards map $(r, \theta), (e_{\mathrm{fwd}} \cdot r, f(\theta), s) \mapsto (r, \theta, e_{\mathrm{bwd}} \cdot r \cdot s)$ is tracked by $\lambda\langle r, \langle t, s\rangle\rangle.\ e_{\mathrm{bwd}} \cdot r \cdot s$.

Conversely, given a container $P : X \to I$ in $\mathsf{Cont}(\mathsf{pAsm}(\mathbb{A}', \mathbb{A}))$, define the extended Weihrauch predicate

$$\begin{array}{l} \phi_P(r) = \{\Phi_P(i) \mid r \Vdash_I i\} \\ \qquad\qquad \text{where } \Phi_P(i) = \{e \in \mathbb{A} \mid \exists s \in P^{-1}(i).\ e \Vdash_X s\} \end{array}$$

Using the axiom of choice, let us fix a section $\zeta_{P,r}$ of the restriction of Φ_P to the $i \in I$ with $r \Vdash_I i$.

If we have a map (φ, ψ) from P to $Q : Y \to J$ in $\mathsf{Cont}(\mathsf{pAsm}(\mathbb{A}', \mathbb{A}))$, with $e_\varphi \Vdash_{I \to J} \varphi$ and e_ψ tracks ψ, we can turn this into a reduction between extended predicates by taking the forward part to be $(e_\varphi, (\varphi'_r)_{r \in \|\phi_P\|})$, where $\varphi'_r(X)$ is defined by taking

$$\begin{array}{rcl} \varphi'_r : \|\phi_P\|(r) & \longrightarrow & \|\phi_Q\|(e_\varphi \cdot r) \\ X & \longmapsto & \Phi_Q(\varphi(\zeta_{P,r}(X))) \end{array}$$

Assuming without loss of generality that the domain of ψ is the assembly where codes are pairs of codes for I and Y, we can take the code $\lambda x.\lambda y.\ e_\psi \cdot \langle x, y\rangle$ as the code for the backwards part and check that this gives us a reduction from ϕ_P to ϕ_Q.

Now let us show that $\phi_{\widehat{p}}$ is always equivalent to p by exhibiting reductions both ways. First, note that $\|\phi_{\widehat{p}}\| = \{r \mid \exists \theta \in p(r).\ r \Vdash_{I_p} (r, \theta)\} = \|p\|$ and that $\Phi_{\widehat{p}}(r, \theta) = \{\langle r, s\rangle \in \mathbb{A} \mid s \in \theta\}$. For the reduction from $\phi_{\widehat{p}}$ to p, we can take the code for the identity together with the map $\xi \mapsto \{\pi_2 \cdot e \mid e \in \xi\}$ and the code $\lambda r.\lambda s.\langle r, s\rangle$ for the backward part. For the other direction, the forward part is obtained by taking the code for the identity together with the r-indexed family of maps $s \mapsto \langle r, s\rangle$, and the backward part by $\lambda x.\lambda y.\ \pi_2\ y$.

Finally, we can also show that there exists morphisms both ways between $P : X \to I$ and $\widehat{\phi_P}$ in $\mathsf{Cont}(\mathsf{pAsm}(\mathbb{A}', \mathbb{A}))$. First, note that the carrier of the object I' of positions of $\widehat{\phi_P}$ is the set of all pairs $(r, \Phi_P(i))$ with $r \Vdash_I i$. There is an obvious (surjective) map $I \to I'$ taking i to the unique pair $(r, \Phi_P(i))$ such that $r \Vdash_I i$, which is tracked by $\lambda x.\ x$. Thanks to the axiom of choice, this map has a section $s : I' \to I$ which is also tracked by $\lambda x.\ x$. These two maps will be the forward part of the reductions we want.

For the backward part, first note the carrier X' of the object of directions of $\widehat{\phi_P}$ is the set of all triples $(r, \Phi_P(i), e)$ with $r \Vdash_I i$ and $e \in \Phi_P(i)$, that is those e tracking some element of $x \in X$ such $P(x) = i$, and such a triple is tracked by the pair $\langle r, e\rangle$. Since P is a map in $\mathsf{pAsm}(\mathbb{A}', \mathbb{A})$, we have that e determines the value of all such triples. Further, P is tracked by some $e_P \in \mathbb{A}'$, so we have that $e_P \cdot e = r$ for every $(r, \Phi_P(i), e) \in |X'|$. So $\lambda x.\ \langle e_P \cdot x, x\rangle$ tracks (and determines) a map $X \to X'$ which can be used to complete the reduction from $\widehat{\phi_P}$ to P.

In the other direction, let us appeal to the axiom of choice to get functions $\epsilon_{P,r,i}$ such that $e \Vdash_X \epsilon_{P,r,i}(e)$ and $P(x) = i$ for every $(r, \Phi_P(i), e) \in |X'|$. We can use that to define the map

$$\begin{array}{ll} X' & \longrightarrow X \\ (r, \theta, e) & \longmapsto \epsilon_{P,r,s(r,\theta)}(e) \end{array}$$

which is a morphism $X' \to X$ in $\mathsf{pAsm}(\mathbb{A}', \mathbb{A})$ tracked by $\lambda x.\ x$ and can similarly be used to complete the reduction from P to $\widehat{\phi_P}$. ☐

Example 6. The Weihrauch predicate WLEM translates to a container isomorphic to $2 \to \nabla(2)$, where $\nabla : \mathsf{Set} \to \mathsf{pAsm}(\mathbb{A}', \mathbb{A})$ is the right adjoint to the forgetful functor $\mathsf{pAsm}(\mathbb{A}', \mathbb{A}) \to \mathsf{Set}$ sending a partitioned assembly to its carrier. It sends a set X to the partitioned assembly with carrier X and realizability relation induced by $\underline{0} \Vdash_{\nabla(X)} x$ (so all elements share the same code). The underlying map $2 \to \nabla(2)$ between carriers is the identity and is tracked by $\lambda x.\ \underline{0}$.

Remark 4. One may note that the definition of extended Weihrauch predicates and reduction between those comports a certain asymmetry, where the input as well as the forward reduction is split between a computable and non-computable part while the output and backward reduction do not contain any non-computable component. This contrasts the notion of extended Weihrauch problem, where inputs and outputs are treated more uniformly as partitioned assemblies. In this setting, the computational part consist of valid codes for the partitioned assemblies, while the non-computational part consists of the elements of the carrier tracked by the codes, so in principle we could have extended Weihrauch problems with "non-computational outputs".

This does not contradict Theorem 3 because we can essentially turn any extended Weihrauch problem $P : X \to U$ into an equivalent (but not isomorphic!) one where the non-computational is suppressed, assuming the axiom of choice: one can quotient the carrier of X by taking x and x' to be equivalent if $P(x) = P(x')$ and they are tracked by the same code. Then we obtain a partitioned assembly X' and a problem $X' \to U$ which turns out to be equivalent provided that we have a section to the quotient map $X \to X'$; this section always exists if we assume the axiom of choice.

4 Operators on containers and Weihrauch problems

In this section, we list a few classical algebraic operators that appear naturally under some form both in the literature on the Weihrauch lattice [9] and containers [28]. Those operators will typically correspond to (quasi-)functors over containers, implying in particular that they are monotonous over degrees. Those functors can also be generically defined without reference to the specifics of type-2 computability; thus we will define them over $\mathsf{Cont}(\mathcal{C})$, making minimal assumptions about $\mathcal{C}$ while staying consistent with $\mathsf{pMod}(\mathbb{A}', \mathbb{A})$ and $\mathsf{pAsm}(\mathbb{A}', \mathbb{A})$.

4.1 The commutative tensor products

One basic fact about Weihrauch degrees is that they form a distributive lattice. This is reflects the fact that under some mild assumptions, $\mathsf{Cont}(\mathcal{C})$ has products and coproducts that distribute over one another.

Definition 9 ([11]). *A category with finite sums and products is called* extensive *when the canonical functor* $+ : \mathcal{C}/A \times \mathcal{C}/B \to \mathcal{C}\big/(A+B)$ *is an isomorphism. We say a category is* lextensive *if it has all finite limits, all finite coproducts and is extensive.*

Proposition 1 (See also [32, (3), (6) and (7)] for a particular case). *If $\mathcal{C}$ is lextensive,* $\mathsf{Cont}(\mathcal{C})$ *has finite products, coproducts and they distribute.*

Proof. It is easy to check that the maps $\mathrm{id}_0 : 0 \to 0$ and $!_0 : 0 \to 1$ in $\mathcal{C}$ are respectively initial and terminal objects in $\mathsf{Cont}(\mathcal{C})$; for the latter, this is because any pullback along any map $0 \to U$ has an initial object as domain.

Given two containers $P_1 : X_1 \to U_1$ and $P_2 : X_2 \to U_2$, a coproduct $P_1 + P_2 : X_1 + X_2 \to U_1 + U_2$ is obtained by functoriality of $+ : \mathcal{C}^2 \to \mathcal{C}$. The coprojections are horizontal morphisms represented $(\mathsf{in}_i, \mathrm{id}_{X_i})$; note that X_i is only a valid domain for the backward map because of extensivity. The copairing $[\Psi_1, \Psi_2] : X_1 + X_2 \to U$ of two reductions, $\Psi_1 = (\phi_1, \psi_1) : X_1 \to U$ and $\Psi_2 =$

$(\phi_2, \psi_2) : X_1 \to U$, is represented by $([\phi_1, \phi_2], \psi_1 + \psi_2)$. This backwards map has the correct domain since, in an extensive category, the pullback of a coproduct is the coproduct of the respective pullbacks along the same morphism.

A product of P_1 and P_2 can be defined as the map $[P_1 \times \mathrm{id}_{U_2}, \mathrm{id}_{U_1} \times P_2] : X_1 \times U_2 + U_1 \times X_2 \to U_1 \times U_2$. The ith projection $P_1 \times P_2 \to P_i$ in $\mathsf{Cont}(\mathcal{C})$ is represented by (π_i, in_i); again this is only a valid morphism because $\mathcal{C}$ is lextensive. The pairing $\langle \Psi_1, \Psi_2 \rangle : X \to U_1 \times U_2$ of two reductions $\Psi_1 = (\phi_1, \psi_1) : X \to U_1$ and $\Psi_2 = (\phi_2, \psi_2) : X \to U_2$, is represented by $(\langle \phi_1, \phi_2 \rangle, [\phi_1, \phi_2])$. To see that the domain of the backwards map works, we first note if you pull a map P_1 back along ϕ_1, the same object (with different projections) is a pullback of $P_1 \times \mathrm{id}_{U_2}$ along $\langle \phi_1, \phi_2 \rangle$. We can then invoke lextensivity to handle the coproduct of pullbacks.

To see that $\mathsf{Cont}(\mathcal{C})$ is distributive, it is enough to notice that the canonical morphism $(P_1 \times Q) + (P_2 \times Q) \to (P_1 + P_2) \times Q$ is represented by a horizontal morphism (c, id) where c is the canonical isomorphism in $\mathcal{C}$ witnessing distributivity for the base. ◻

Proposition 2. *Assuming $\mathcal{C}$ is lextensive, answerable containers of $\mathsf{Cont}(\mathcal{C})$ are stable under binary products and coproducts. The initial object is an answerable container but not the terminal object.*

Proof. For binary products, we first note that $P_1 \times P_2$ in $\mathsf{Cont}(\mathcal{C})$ is obtained by copairing two pullback-stable epimorphisms: we can show that $P_1 \times \mathrm{id}_{U_2}$ is a pullback-stable epimorphism since it is the pullback of P_1 along the projection $\pi_1 : U_1 \times U_2 \to U_1$, and similarly for $\mathrm{id}_{U_1} \times P_2$. Now, in extensive categories, the pullback of a copairing of the pullbacks is the copairing of the pullbacks, and a copairing is epimorphic as long as one of its component is an epi, so we can conclude.

$P_1 + P_2$ being represented by a pullback-stable epi follows from the fact that coproducts are disjoint in an extensive category. Extensivity also ensures that if we have a map $X \to 0$ in $\mathcal{C}$, X is initial; hence, $\mathrm{id}_0 : 0 \to 0$ is a pullback-stable epimorphism. ◻

Containers also admit another natural monoidal product, which is sometimes called the *parallel product*. The parallel product of $P_1 : X_1 \to U_1$ and $P_2 : X_2 \to U_2$, which we write $P_1 \otimes P_2$, is represented by the morphism $P_1 \times P_2 : X_1 \times X_2 \to U_1 \times U_2$ obtained by the functoriality of $\times$ in $\mathcal{C}$ (assuming $\mathcal{C}$ has cartesian products)[28, Definition 3.65]. Regarded as a Weihrauch problem, this means we may ask a question to both P_1 and P_2 and get back answers for both questions.

It is straightforward to check that $\otimes$ is a symmetric monoidal product in $\mathcal{C}$ (see [24, Chapter VII]) and that answerable containers are stable under $\otimes$. If we further assume that $\mathcal{C}$ is lextensive, we can show that there are further distributive laws.

Proposition 3 (See also [32, (6) and (28)] for a particular case). *If $\mathcal{C}$ is lextensive, then we have natural transformations $P \otimes Q + P \otimes R \cong P \otimes (Q + R)$ and $(P \otimes Q) \times R \to P \otimes (Q \times R)$ in $\mathsf{Cont}(\mathcal{C})$.*

Proof. The proof that $\otimes$ distributes over $+$ is essentially the same as the proof that $\times$ distributes, so we omit it.

For the second claim, assuming that we have $P : X \to U$, $Q : Y \to V$ and $R : Z \to W$, the natural transformation is represented by the vertical map (id, c) with $c = \mathrm{id}_{X \times Y \times Z} + (P \times \mathrm{id}_{V \times Z}) : (X \times Y \times W) + (X \times V \times Z) \to (X \times Y \times W) + (U \times V \times Z)$. Note that it is a valid morphism representative only because the domain of c is isomorphic to $X \times (Y \times W + V \times Z)$; this is true because $\mathcal{C}$ is distributive. ◻

Note that with this and the functoriality of $\times$, when moving to $\mathsf{Cont}(\mathcal{C})/_{\leftrightarrow}$ we recover all of the axioms listed in [27] pertaining to binary joins, meets and parallel products.

4.2 The composition product in nicer categories of containers

Another monoidal product which appears in the literature on containers is the *composition product*. The most natural way to introduce this is maybe by making the link between containers and polynomial functors [13, Theorem 2.12]. In short, assuming that $\mathcal{C}$ is locally cartesian closed, we map any container $P : X \to U$ to the endofunctor over $\mathcal{C}$ defined by $\Sigma_{!_U} \circ \Pi_P \circ !_X^* : \mathcal{C}/_1 \to \mathcal{C}/_1$ and the isomorphism $\mathrm{dom} : \mathcal{C}/_1 \cong \mathcal{C}$. Regarding P as an internal family $(X_u)_{u \in U}$ to $\mathcal{C}$, on objects this functor maps A to $\sum_{u \in U} A^{X_u}$. Those endofunctors are called *polynomial endofunctors* over $\mathcal{C}$, and one important fact is that strong natural transformations between those correspond exactly to container morphisms. One important theorem is that polynomial endofunctors are stable under composition, which induces a monoidal product on polynomials sometimes called the *substitution* or *sequential product* [13, (9) Proposition 1.12].

In the spirit of Weihrauch reducibility, the sequential product $P \star Q$ can be regarded as the problem whose inputs are composed of an input v for Q together with a function f turning a solution for said input into an input for P. A solution for $P \star Q$ then consists of a solution for v and a solution for $f(v)$. In sets, assuming we have $Q : Y \to V$, $X_u = P^{-1}(u)$ and $Y_v = Q^{-1}(v)$ for every u, v, $P \star Q$ is the family $\left(\sum_{y \in Y_v} X_{f(y)}\right)_{(v,f)}$ whose indexing set is $\sum_{v \in V} (Y_v \to X)$. This can be internalized in any locally cartesian closed category (see Figure 4).

In short, reducing to $P \star Q$ is the ability to make an oracle call to Q and then P. The sequential product of Weihrauch degrees is designed to capture this intuition and is thus defined this way [37, Definition 3][9], but for a small wrinkle: $\mathsf{pMod}(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2)$ is not locally cartesian closed!

Proposition 4. *$\mathsf{pMod}(\mathcal{K}_2)$, $\mathsf{pAsm}(\mathcal{K}_2)$, $\mathsf{pMod}(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2)$ and $\mathsf{pAsm}(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2)$ are not cartesian closed.*

Proof. Let us focus on $\mathsf{pAsm}(\mathcal{K}_2)$ (proofs along the same lines work for the other categories). We are going to show there is no exponential object for $2^{\mathbb{N}^{\mathbb{N}}}$;

[9] The original definition in [9] actually has a slightly different flavour, but they correspond to the same Weihrauch degree.

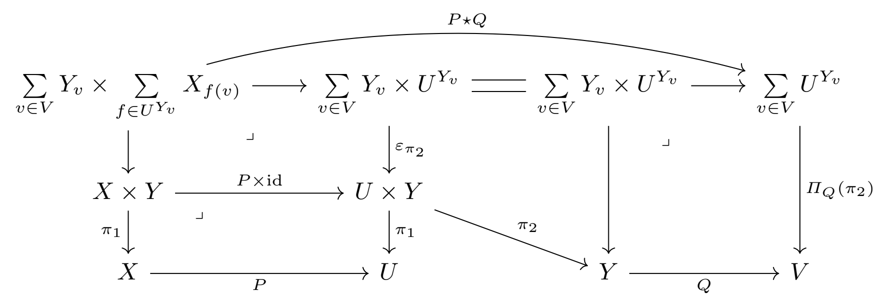


Fig. 4. Diagrammatic definition of the composition product $P \star Q$ of two containers P and Q (simplified version of [13, (9)]). To ease reading, we name objects in the top row according to their definition in type theory; the definition is determined (up to isomorphism) by P, Q, the projections out of $U \times Y$, $\Pi_Q(\pi_2)$, the denoted pullback squares and ε_{π_2} being to the counit of the adjunction $Q^* \dashv \Pi_Q$ at π_2. $P \star Q$, when regarded as a polynomial functor $[\![P \star Q]\!]$, corresponds to the reverse composition $[\![Q]\!] \circ [\![P]\!]$ of functors.

let us call E such an exponential object, which comes with an evaluation map $E \times \mathbb{N}^{\mathbb{N}} \to 2$ and a universal property. First, because of the uniqueness part of the universal property, we must have that E is modest. We can consider the partitioned modest set T of well-founded trees with countable branching whose leaves are labeled in 2. There is a computable map $T \times \mathbb{N}^{\mathbb{N}} \to 2$ that assigns each pair (t, p) to the unique boolean that the path p traverses in t, so, using the exponential structure, we have a fortiori a surjective map $r : T \to E$ in $\mathsf{pMod}(\mathcal{K}_2)$. Using the evaluation map, we can continuously map a code of E into a code tracking the underlying function $f : \mathbb{N}^{\mathbb{N}} \to 2$, and from such a code is up to encoding details an element of T encoding f. Hence we have a section s of r.

Now consider some t which is in the image of s, and replace one of its leaf l by an infinite well-founded tree whose leaves are labelled the same way to obtain t'. We have $s(r(t')) = t$. But since $s \circ r$ is continuous, it must only inspect a finite part of t' to determine the label of l. Then considering t'' obtained from t' by flipping one of the unexamined leaves that did not belong to t leads to a contradiction. □

The failure of cartesian closure is essentially due to a lack of quotients rather than a lack of power: it is not that we cannot build modest sets of codes for function spaces, it is that we cannot guarantee that a function has a unique code. It means in particular that we still have a weak version of local cartesian closure for pMod, which is enough to talk about spaces of codes for functions; this is why the same intuition can be used to build the sequential product of Weihrauch problems. But a disadvantage is that we then need to reconstruct from scratch certain proofs, for instance that the sequential product on the left distributes over coproducts. In the next two subsections, we rather present a strategy to derive those results by observing that pMod is the full subcategory of regular projective objects of the locally-cartesian closed category Mod and that Mod has enough projectives.

4.3 Weak structure, projectives and quasi-functors

The definition of various canonically determined structures like cartesian closure comes from the characterizations of *adjoint functors*. One of the shortest definition states that a functor $L : \mathcal{C} \to \mathcal{D}$ is a left adjoint to the functor $R : \mathcal{D} \to \mathcal{C}$ if we have isomorphisms

$$[L(X), A]_{\mathcal{D}} \cong [X, R(A)]_{\mathcal{C}} \qquad \text{natural in } A \text{ and } B$$

To characterize weak structure, one can take a similar approach. However, this requires weakening the notion of what is a right adjoint.

Definition 10 ([18, §1]). *A weak right adjoint of the functor $F : \mathcal{C} \to \mathcal{D}$ consists of*

- *a map G from objects of $\mathcal{D}$ to objects of $\mathcal{C}$*
- *and natural surjections $[-, G(A)]_{\mathcal{C}} \twoheadrightarrow [F(-), A]_{\mathcal{D}}$.*

Example 7. For any small category $\mathcal{C}$, the category $\mathcal{C} \to {}^{\mathcal{C}}/_{\leftrightarrow}$ has an obvious weak right adjoint if we assume the axiom of choice.

Analogously to the definition of locally-cartesian closed categories, we say that a category is *weakly* locally cartesian closed when all functors f^* have weak right adjoints [10, Remark 3.2].

Definition 11. *A* regular epimorphism *is an epimorphism which is a coequalizer of some pair of morphisms.*

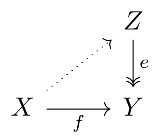
Example 8. In topological spaces, and a fortiori most of the subcategories of those that we consider in this paper, the monomorphisms $2 \rightarrowtail \mathbb{S}$ are epimorphic, pullback-stable but not regular.

A characterization of regular epimorphisms in (multi)-represented spaces is that they have a multi-valued inverse. In the language of assemblies, this translates to the following lemma.

Proposition 5 ([36, Proof of Theorem 1.5.2]). *Both in* $\mathsf{Mod}(\mathbb{A}', \mathbb{A})$ *and* $\mathsf{Asm}(\mathbb{A}', \mathbb{A})$*, a morphism $f : X \to Y$ tracked by $e \in \mathbb{A}'$ is a regular epimorphism if and only if there is $e' \in \mathbb{A}'$ such that*

- *there exists x such that $e' \cdot e'' \Vdash_X x$ for every y and $e'' \Vdash_Y y$*
- *$\lambda x.\, e \cdot (e' \cdot x)$ tracks the identity on Y*

Definition 12. *An object X is called* (regular) projective *when it has the left lifting property against regular epimorphisms, that is, whenever we have a regular epimorphism $e : Z \twoheadrightarrow Y$ and a map $f : X \to Y$, there exists some $g : X \to Z$ such that $f = e \circ g$.*



For a category $\mathcal{C}$, we call $p\mathcal{C}$ its full subcategory consisting of regular projective objects.

A (regular) projective cover *of an object X is a projective object $\widetilde{X}$ together with a regular epimorphism $\widetilde{X} \twoheadrightarrow X$*

Turning back to the specific categories of interest here, for an assembly X of $\mathsf{pAsm}(\mathbb{A}', \mathbb{A})$, a projective cover $\widetilde{X}$ can be defined as follows: the carrier is $|\widetilde{X}| = \Vdash_X \subseteq \mathbb{A} \times X$, i.e. the set of all pairs (e, x) such that e realizes x, and realizability relation is $e \Vdash_{\widetilde{X}} (e, x)$. The same construction works for modest sets, where we can simplify things further by simply restricting the carrier to its first component, since modesty means that a code uniquely determines the elements we want to code. So all in all, we have the following.

Lemma 2. *The projective objects of modest sets (assemblies) are, up to isomorphism, the partitioned modest sets (assemblies). Furthermore, the categories of modest sets and assemblies always have enough projectives.*

Proof. For modest sets, this is [5, Theorem 3.1]. For assemblies, we can deduce that from the characterization of partitioned assemblies as the projectives of the realizability topos $\mathsf{RT}(\mathbb{A}', \mathbb{A})$ [36, Proposition 3.2.7] by noting that coequalizers are preserved by the embedding $\mathsf{Asm}(\mathbb{A}', \mathbb{A}) \to \mathsf{RT}(\mathbb{A}', \mathbb{A})$. □

Categories $\mathcal{C}$ with enough projectives *almost* have a coreflection on their projectives, which we can characterize via a weak right adjoint.

Lemma 3. *If $\mathcal{C}$ has enough regular projectives, the full and faithful embedding induced by the inclusion $p\mathcal{C} \subseteq \mathcal{C}$ has a weak right adjoint $X \mapsto \widetilde{X}$ with $\widetilde{X} = X$ for any object X of $p\mathcal{C}$.*

Proof. For any object X, fix a regular projective cover $\varepsilon_X : \widetilde{X} \twoheadrightarrow X$, taking the identity when X was already regular projective. This gives a counit for our weak adjoint, whose morphism part is then the postcomposition

$$\varepsilon_X \circ - : \left[A, \widetilde{X}\right]_{p\mathcal{C}} \twoheadrightarrow [A, X]_{\mathcal{C}}$$

It is obviously natural in A. It is easily seen to be surjective since A is regular projective and ε_X is a regular epimorphism. □

Remark 5. At this stage, one can use that functor to prove that $p\mathcal{C}$ is weakly locally cartesian closed from the fact that $\mathcal{C}$ is locally cartesian closed and has enough projectives. In fact this can be strengthened to an equivalence [10, Theorem 3.3].

Something to be noted is that weak right adjoints need not be functors; however they are still *quasi-functors* in the following sense.

Definition 13 ([18, §0]). *A quasi-functor $F : \mathcal{C} \rightsquigarrow \mathcal{D}$ maps objects of $\mathcal{C}$ to objects of $\mathcal{D}$ and morphisms $X \to Y$ of $\mathcal{C}$ to non-empty sets of morphisms $F(X) \to F(Y)$ such that*

- $id_{F(X)} \in F(id_X)$ *for every object X of $\mathcal{C}$*
- *if $f' \in F(f)$ and $g' \in F(g)$ for composable arrows f and g, then $f' \circ g' \in F(f \circ g)$.*

While quasi-functors are weaker than functors, we can note that the two notions coincide when the codomain is a preorder. Quasi-functors over $\mathcal{C}$, much like functors, also induce monotonous operations over the poset $\mathcal{C}/_{\leftrightarrow}$.

4.4 The composition product for containers on projectives

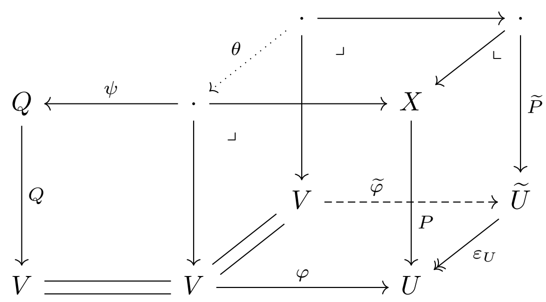
Now we will define a sequential product $\star_p$ on $\mathsf{Cont}(p\mathcal{C})$ in terms of the sequential product $\star$ on $\mathsf{Cont}(\mathcal{C})$. First, we will use the following definition to introduce a nice full subcategory of $\mathsf{Cont}(\mathcal{C})$.

Definition 14. *Say that a morphism $f : X \to U$ is* fibrewise projective *if, for every $g : V \to U$ with V regular projective, then the domain of the pullback $g^*(f)$ is also regular projective.*

Fibrewise projective morphisms are easily seen to be closed under composition and arbitrary pullbacks. Let us call $\mathsf{Cont}_{\mathsf{fp}}(\mathcal{C})$ the full subcategory of $\mathsf{Cont}(\mathcal{C})$ whose objects are those containers that are fibrewise projective when regarded as morphisms of $\mathcal{C}$.

Lemma 4. *If $\mathcal{C}$ has pullbacks and enough regular projectives, the functor induced by the inclusion $\mathsf{Cont}_{\mathsf{fp}}(\mathcal{C}) \subseteq \mathsf{Cont}(\mathcal{C})$ has a weak right adjoint $P \mapsto \widetilde{P}$ with $P = \widetilde{P}$ for any object P of $\mathsf{Cont}_{\mathsf{fp}}(\mathcal{C})$.*

Proof. For any $P : X \to U$, we can obtain $\widetilde{P}$ by pulling back along a regular projective cover $\widetilde{U} \twoheadrightarrow U$; since we assume P was fibrewise projective, this ensures $\widetilde{P}$ belongs to $\mathsf{Cont}_{\mathsf{fp}}$. The pullback itself gives us an horizontal morphism $\varepsilon_P : \widetilde{P} \to P$ in Cont; much like in the proof of Lemma 3, this gives us a counit for a weak right adjoint and its morphism part is obtained by postcomposing ε_P. This is surjective because, given a morphism (φ, ψ) from Q to P, we can obtain a morphism $(\tilde{\varphi}, \psi \circ \theta)$ from Q to $\widetilde{P}$ as shown in the commuting diagram below. A morphism $\widetilde{\varphi}$ such that $\varepsilon_U \circ \widetilde{\varphi} = \varphi$ exists because V is projective and θ is defined using the universal property of the pullback of P along φ.



□

Now, assuming $\mathcal{C}$ is locally cartesian closed and $p\mathcal{C}$ is stable under pullback, the sequential product $\star$ actually restricts to a functor $\mathsf{Cont}(p\mathcal{C}) \times \mathsf{Cont}(p\mathcal{C}) \to \mathsf{Cont}_{\mathsf{fp}}(\mathcal{C})$; this is because the map representing $P \star Q$ is the composition of two maps obtained by pullback from P and Q (see Figure 4). Hence we can define a quasi-functor $- \star_p - : \mathsf{Cont}(p\mathcal{C}) \times \mathsf{Cont}(p\mathcal{C}) \rightsquigarrow \mathsf{Cont}(p\mathcal{C})$ by postcomposing with the weak right adjoint $\mathsf{Cont}_{\mathsf{fp}}(\mathcal{C}) \to \mathsf{Cont}(p\mathcal{C})$ we just defined.

Remark 6. While the precise definition of $- \star_p -$ up to isomorphism[10] does depend on the choice of projective covers, the operator on the posetal reflection $\mathsf{Cont}(p\mathcal{C})/_{\leftrightarrow}$ does not depend on that.

Furthermore, when $\mathcal{C}$ has enough projectives, the weak right adjoint $\widetilde{\Pi}_f$ to f^* actually yields projective covers of Π_f; this means that $P \star_p Q$ may coincide exactly with the construction obtained by mimicking the definition of $P \star Q$ by replacing all instances of Π_Q with its weak version.

Remark 7. The fact that $\star_p$ is defined via a weak right adjoint ensures that for any objects P and Q of $\mathsf{Cont}(p\mathcal{C})$, there is always a map $P \star_p Q \to P \star Q$ in $\mathsf{Cont}_{\mathsf{fp}}(\mathcal{C})$. However we conjecture that for $\mathcal{C} = \mathsf{Mod}(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2)$, there are P and Q such that there are no maps $P \star Q \to P \star_p Q$.

We can now use this to transfer some easy results about $\star$ to results about $\star_p$ (and a fortiori the sequential product in the (extended) Weihrauch degrees) in a generic way. By way of example, let us rederive some known inequalities in the Weihrauch degrees.

Lemma 5 (See e.g. [32, (26)]). *If $\mathcal{C}$ is locally cartesian closed and extensive, then in $\mathsf{Cont}(\mathcal{C})$ we have maps $(P \star Q) \times R \to (P \times R) \star Q$.*

Proof. Calling $[\![P]\!]$ the polynomial functor corresponding to P, going though the equivalence of categories with polynomial functors alluded to before, it suffices to provide a strong natural transformation

$$[\![(P \star Q) \times R]\!] \quad \Longrightarrow \quad [\![(P \times R) \star Q]\!]$$

[10] Up to equality, it depends also on a choice of pullbacks.

Since we have an equivalence of categories, that the cartesian product of (polynomial) functors are computed pointwise in functor categories and that sequential product corresponds to (reverse) composition of polynomial functors, it thus suffices to provide a strong natural transformation

$$(\llbracket Q \rrbracket \circ \llbracket P \rrbracket) \times \llbracket R \rrbracket \quad \Longrightarrow \quad \llbracket Q \rrbracket \circ (\llbracket P \rrbracket \times \llbracket R \rrbracket)$$

For this, it is sufficient to provide a transformation which is natural in A and B

$$\llbracket Q \rrbracket(A) \times B \quad \longrightarrow \quad \llbracket Q \rrbracket(A \times B)$$

But this amounts to showing that $\llbracket Q \rrbracket$ is a strong functor, which is the case as it is a polynomial functor [13, §1.2]. □

Corollary 1. $(a \star b) \sqcap c \leq (a \sqcap c) \star b$ *is valid in the (extended) Weihrauch degrees.*

Proof. First note that if we have that $\mathcal{C}$ is lextensive, has enough projectives and that its projectives are closed under coproduct and pullbacks, then $\times$ is preserved by the embedding $\mathsf{Cont}(p\mathcal{C}) \subseteq \mathsf{Cont}_{\mathsf{fp}}(\mathcal{C})$ preserves the cartesian product and coproducts built in $\mathsf{Cont}(p\mathcal{C})$. As a result, assuming now that $\mathcal{C}$ is also locally cartesian closed, we have that $(P \star_p Q) \times R$ is obtained by applying the weak right adjoint to $\mathsf{Cont}(p\mathcal{C}) \subseteq \mathsf{Cont}_{\mathsf{fp}}(\mathcal{C})$ to $(P \star Q) \times R$; by definition, it is also the case for $(P \times R) \star_p Q$. Hence we can map the morphism $(P \star Q) \times R \to (P \times R) \star Q$ through the weak right adjoint and obtain a morphism that witnesses the validity of the inequality in $\mathsf{Cont}(p\mathcal{C})/_{\leftrightarrow}$. The corollary can then be derived since $\mathsf{Asm}(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2)$ and $\mathsf{Mod}(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2)$ are extensive and locally cartesian closed. □

While this approach can probably also help derive further distributive laws and similar results on other operators that also rely on the weak local cartesian closure to be defined, such as various closures on slightly extended Weihrauch degrees[11], it is not yet clear to us if it is powerful enough to allow to derive all trivial inequalities without introspecting the definitions further. For instance, it is not clear to us how to derive that $\star$ is associative in the Weihrauch degrees just using that $\star$ is a monoidal product in $\mathsf{Cont}_{\mathsf{fp}}(\mathsf{Asm}(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2))$ and that $\star_p$ is $\star$ post-composed with a quasi-functor $\mathsf{Cont}_{\mathsf{fp}}(\mathsf{Asm}(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2)) \rightsquigarrow \mathsf{Cont}(\mathsf{pAsm}(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2))$.

5 Conclusion

We hope that making the connection explicit between Weihrauch reducibility and containers will allow to connect more tightly the literature on the two topics.

[11] In the literature on Weihrauch degrees, the monoidal closure, that is the right adjoint to $Q \mapsto Q \otimes P$, has recently been studied in e.g. [4,21,25]. The cartesian closure has been shown not to exist on the Weihrauch degrees [14, Theorem 4.9]; but it does exist in the slightly extended degrees [3]. The left adjoint to $Q \mapsto P \star Q$ has been studied a bit more systematically and was introduced formally in [9, §3.3]. All of those operators appear in the extended literature on containers - see e.g. [32] for their definition in $\mathsf{Cont}(\mathsf{Set})$.

For instance, it would be nice to relate formally universal equational theories of the Weihrauch lattice with operators, like the ones studied in [27,31], to similar theories interpreted in categories of containers over toposes with $\mathcal{W}/\mathcal{M}$-types. Also, since we have shown that Weihrauch problems are a definable subclasses of containers in assemblies over $(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2)$, it might be a way to formalize the theory of Weihrauch reducibility in in proof assistants in a synthetic style (see e.g. [12] for similar work for Turing-reducibility in Rocq and [22] for a development in a synthetic style in $\mathsf{Asm}(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2)$).

Another area which might be worth investigating is the connections between Weihrauch reducibility and proof theory, as one can naturally interpret linear type theories in fibrations that are closely linked to categories of containers [26]. Could this be exploited to characterize Weihrauch reducibility in a type theory, a bit in the spirit of [35,38]? One difficulty with a naive attempt is that interpreting types directly in $\mathfrak{Fam}(\mathfrak{Fam}^{\mathrm{op}}(\mathrm{id}_{\mathsf{pAsm}(\mathcal{K}_2^{\mathrm{rec}}, \mathcal{K}_2)}))$ (see [16] for a definition) seems somewhat counter-intuitive, since $\mathbf{I} \leq_{\mathrm{W}} P \multimap Q$ holds iff P reduces to ("is less powerful than") Q.

Acknowledgments. Many thanks to Eike Neumann, Arno Pauly and Manlio Valenti for discussions (especially on Proposition 4) and encouragements. Many thanks as well to Takayuki Kihara and an anonymous reviewer for pointers to the literature and comments that helped improve the presentation of the paper.

The authors acknowledge support by Swansea University and grant to Swansea University a non-exclusive, irrevocable, sub-licensable, worldwide license to make the accepted manuscript available on its institutional repository.

References

1. Abbott, M.G., Altenkirch, T., Ghani, N.: Categories of containers. In: Gordon, A.D. (ed.) FOSSACS 2003 proceedings. Lecture Notes in Computer Science, vol. 2620, pp. 23–38. Springer (2003). https://doi.org/10.1007/3-540-36576-1_2
2. Ahman, D., Bauer, A.: Comodule representations of second-order functionals (2024), https://arxiv.org/abs/2409.17664
3. Altenkirch, T., Levy, P., Staton, S.: Higher-order containers. In: Ferreira, F., Löwe, B., Mayordomo, E., Mendes Gomes, L. (eds.) Programs, Proofs, Processes. pp. 11–20. Springer Berlin Heidelberg, Berlin, Heidelberg (2010)
4. Andrews, U., Lempp, S., Marcone, A., Miller, J.S., Valenti, M.: A jump operator on the Weihrauch degrees. arXiv 2402.13163 (2024)
5. Bauer, A.: The realizability approach to computable analysis and topology. Ph.D. thesis, Carnegie Mellon University (2000)
6. Bauer, A.: Instance reducibility and Weihrauch degrees. Log. Methods Comput. Sci. **18**(3) (2022). https://doi.org/10.46298/LMCS-18(3:20)2022, https://doi.org/10.46298/lmcs-18(3:20)2022
7. Brattka, V., Gherardi, G., Marcone, A.: The Bolzano–Weierstrass Theorem is the jump of Weak Kőnig's Lemma. Annals of Pure and Applied Logic **163**(6), 623–655 (Jun 2012). https://doi.org/10.1016/j.apal.2011.10.006
8. Brattka, V., Gherardi, G., Pauly, A.: Weihrauch complexity in computable analysis. CoRR **abs/1707.03202** (2017)

9. Brattka, V., Pauly, A.: On the algebraic structure of Weihrauch degrees. Logical Methods in Computer Science **14**(4) (2018). https://doi.org/10.23638/LMCS-14(4:4)2018
10. Carboni, A., Rosolini, G.: Locally cartesian closed exact completions. Journal of Pure and Applied Algebra **154**(1), 103–116 (2000). https://doi.org/10.1016/S0022-4049(99)00192-9
11. Carboni, A., Lack, S., Walters, R.: Introduction to extensive and distributive categories. Journal of Pure and Applied Algebra **84**(2), 145–158 (1993). https://doi.org/10.1016/0022-4049(93)90035-R, https://www.sciencedirect.com/science/article/pii/002240499390035R
12. Forster, Y., Kirst, D., Mück, N.: Oracle computability and turing reducibility in the calculus of inductive constructions. In: Hur, C. (ed.) Programming Languages and Systems - 21st Asian Symposium, APLAS 2023, Taipei, Taiwan, November 26-29, 2023, Proceedings. Lecture Notes in Computer Science, vol. 14405, pp. 155–181. Springer (2023)
13. Gambino, N., Kock, J.: Polynomial functors and polynomial monads. Mathematical Proceedings of the Cambridge Philosophical Society **154**(1), 153–192 (Sep 2012). https://doi.org/10.1017/s0305004112000394
14. Higuchi, K., Pauly, A.: The degree-structure of Weihrauch-reducibility. Logical Methods in Computer Science **9**(2) (2013). https://doi.org/10.2168/LMCS-9(2:2)2013
15. Hirsch, M.D.: Applications of topology to lower bound estimates in computer science. Ph.D. thesis, University of California, Berkeley (1990)
16. Hofstra, P.J.W.: The Dialectica monad and its cousins. In: Makkai, M., Hart, B. (eds.) Models, Logics, and Higher-dimensional Categories: A Tribute to the Work of Mihály Makkai. CRM proceedings & lecture notes, American Mathematical Society (2011)
17. Iljazović, Z., Kihara, T.: Computability of subsets of metric spaces. In: Handbook of Computability and Complexity in Analysis, pp. 29–69. Springer (2021)
18. Kainen, P.C.: Weak adjoint functors. Mathematische Zeitschrift **122**, 1–9 (1971)
19. Kihara, T.: Borel-piecewise continuous reducibility for uniformization problems. Logical Methods in Computer Science **12**(4) (Apr 2017). https://doi.org/10.2168/lmcs-12(4:4)2016
20. Kihara, T.: Rethinking the notion of oracle: A prequel to Lawvere-Tierney topologies for computability theorists (2024), https://arxiv.org/abs/2202.00188
21. Kihara, T., Marcone, A., Pauly, A.: Searching for an analogue of ATR_0 in the Weihrauch lattice. The Journal of Symbolic Logic **85**(3), 1006–1043 (2020). https://doi.org/10.1017/jsl.2020.12
22. Konečný, M., Park, S., Thies, H.: Certified computation of nondeterministic limits. In: NASA Formal Methods. pp. 771–789. Springer (2022). https://doi.org/10.1007/978-3-031-06773-0_41
23. Lempp, S., Miller, J., Pauly, A., Soskova, M., Valenti, M.: Minimal covers in the Weihrauch degrees. Proceedings of the American Mathematical Society **152**(11), 4893–4901 (2024)
24. Mac Lane, S.: Categories for the working mathematician, vol. 5. Springer Science & Business Media (2013)
25. Marcone, A., Valenti, M.: The open and clopen Ramsey theorems in the Weihrauch lattice. The Journal of Symbolic Logic **86**(1), 316–351 (March 2021). https://doi.org/10.1017/jsl.2021.10

26. Moss, S.K., von Glehn, T.: Dialectica models of type theory. In: Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2018, Oxford, UK, July 09-12, 2018. pp. 739–748 (2018)
27. Neumann, E., Pauly, A., Pradic, C.: The equational theory of the Weihrauch lattice with multiplication. CoRR **abs/2403.13975** (2024). https://doi.org/10.48550/ARXIV.2403.13975, https://doi.org/10.48550/arXiv.2403.13975
28. Niu, N., Spivak, D.I.: Polynomial functors: A mathematical theory of interaction (2024), https://arxiv.org/abs/2312.00990
29. van Oosten, J., Zou, T.: Classical and relative realizability (2016), https://arxiv.org/abs/1603.03621
30. Pauly, A.: Many-one reductions between search problems. CoRR **abs/1102.3151** (2011), http://arxiv.org/abs/1102.3151
31. Pradic, C.: The equational theory of the Weihrauch lattice with (iterated) composition (2025), https://arxiv.org/abs/2408.14999
32. Spivak, D.I.: A reference for categorical structures on **Poly** (2024)
33. Streicher, T.: Fibered categories à la Jean Bénabou (2023)
34. Trotta, D., Valenti, M., de Paiva, V.: Categorifying computable reducibilities. arXiv preprint arXiv:2208.08656 (2022)
35. Uftring, P.: The characterization of Weihrauch reducibility in systems containing $\mathrm{E}-\mathrm{PA}^{\omega}+\mathrm{QF}-\mathrm{AC}^{0,0}$. The Journal of Symbolic Logic **86**(1), 224–261 (Oct 2020). https://doi.org/10.1017/jsl.2020.53, http://dx.doi.org/10.1017/jsl.2020.53
36. Van Oosten, J.: Realizability: an introduction to its categorical side, vol. 152. Elsevier (2008)
37. Westrick, L.: A note on the diamond operator. Computability **10**(2), 107–110 (2021). https://doi.org/10.3233/COM-200295
38. Yoshimura, K.: General treatment of non-standard realizabilites (2016), unpublished