# Logical relations for call-by-push-value models, via internal fibrations in a 2-category

Pedro H. Azevedo de Amorim
*Dept. of Computer Science*
*University of Oxford*
Oxford, UK
pedro.azevedodeamorim@cs.ox.ac.uk

Satoshi Kura
*Fac. of Ed. and Integrated A&S*
*Waseda University*
Tokyo, Japan
satoshi.kura@aoni.waseda.jp

Philip Saville
*Dept. of Informatics*
*University of Sussex*
Brighton, UK
p.saville@sussex.ac.uk

***Abstract*—We give a denotational account of logical relations for call-by-push-value (CBPV) in the fibrational style of Hermida, Jacobs, Katsumata and others. Fibrations—which axiomatise the usual notion of sets-with-relations—provide a clean framework for constructing new, logical relations-style, models. Such models can then be used to study properties such as effect simulation.**

**Extending this picture to CBPV is challenging: the models incorporate both adjunctions and enrichment, making the appropriate notion of fibration unclear. We handle this using 2-category theory. We identify an appropriate 2-category, and define CBPV fibrations to be fibrations internal to this 2-category which strictly preserve the CBPV semantics.**

**Next, we develop the theory so it parallels the classical setting. We give versions of the codomain and subobject fibrations, and show that new models can be constructed from old ones by pullback. The resulting framework enables the construction of new, logical relations-style, models for CBPV.**

**Finally, we demonstrate the utility of our approach with particular examples. These include a generalisation of Katsumata's $\top\top$-lifting to CBPV models, an effect simulation result, and a relative full completeness result for CBPV without sum types.**

## I. Introduction

This paper is about extending the denotational theory of logical relations from effectful call-by-value languages to Levy's call-by-push-value [1], [2]. Logical relations are a fundamental tool for proving metatheoretic properties of logics and programming languages. We begin with a brief overview; a more detailed account can be found in e.g. [3, §2.2].

In its simplest form, a logical relation $R$ for a typed programming language consists of a predicate $R_A \subseteq [\![A]\!]$ on the (set-theoretic) interpretation of each type $A$, such that the predicate at complex types is determined inductively by a *logical relations condition*. This condition typically encodes the elimination rules of the corresponding type. This is particularly clear in operationally-motivated examples, where $[\![A]\!]$ is a set of closed terms of type $A$. For product types $A_1 \times A_2$, for example, one typically requires that $R_{A_1 \times A_2}$ consists of those $M$ such that the projections $\pi_i(M)$ are elements of $R_{A_i}$ for $i = 1, 2$. The key property of logical relations—which follows from the logical relations conditions—is that they are determined by their base types: if the interpretation $[\![c]\!] \in [\![\beta]\!]$ of every constant $\diamond \vdash c : \beta$ is in the relation—so that $[\![M]\!] \in R_\beta$—then for every closed term $N$ we have $[\![N]\!] \in R_A$. This fact, which is often proven by induction on the terms, is called the *basic lemma* of logical relations.

Viewed in this way, logical relations are already a powerful tool. For example, Sieber used a similar construction to show that Plotkin's parallel-or function [4] is not definable in the standard domains model of PCF [5]. Moreover, a host of sophisticated refinements have extended these ideas to reason about subtle properties of rich languages (e.g. [6], [7], [8]).

Logical relations also fit into a very general denotational story, originally due to Ma & Reynolds [9] and Mitchell & Scedrov [10] and extended by many authors since (e.g. [11], [3], [12], [13], [14]). The central technical aim of this paper is to extend this story to call-by-push-value (CBPV) [1], [2]. Achieving this requires two main technical steps. First, defining an appropriate notion of 'fibration for logical relations' (cf. [15]) by restricting to fibrations which strictly preserve the model structure. Second, showing that we can universally construct new fibrations for logical relations from old ones. The first step tells you how to define logical relations; the second shows how to construct a wide variety of examples.

To explain these steps, and the obstructions to extending this straightforwardly to CBPV, we begin by outlining the story in two simpler cases, namely the simply-typed λ-calculus (STLC) and Moggi's monadic metalanguage $\lambda_{\mathrm{ml}}$ [16]. We assume some familiarity with (Grothendieck) fibrations and their theory; for a detailed introduction see [13], [3].

### A. Logical relations from fibrations for logical relations

First let us sketch how relations models—and, more generally, fibrations for logical relations—determine logical relations. The high-level picture is as follows. One starts with a (category-theoretic) model $\mathbb{M}$ and constructs a *relations model* $\dot{\mathbb{M}}$, together with a functor $p : \dot{\mathbb{M}} \to \mathbb{M}$ which strictly preserves the model structure. The objects of $\dot{\mathbb{M}}$ are thought of as objects paired with relations, and the morphisms as maps preserving those relations. Following Jacobs [13] and Hermida [3], we encode this relation-like structure by asking for $p$ to be a fibration; we then say a *fibration for logical relations* is a fibration which strictly preserves the model structure.

PAdA was supported by the ERC Consolidator Grant BLAST, and the ARIA programme on Safeguarded AI. SK was supported by JST ACT-X Grant Number JPMJAX2104; JSPS Overseas Research Fellowships. PS was partially supported by the Air Force Office of Scientific Research (award number FA9550-21-1-0038).

Within this framework the logical relations conditions are embodied by the model structure of $\dot{\mathbb{M}}$, the relations $R_A$ are replaced by an interpretation $[\![A]\!]^{\dot{\mathbb{M}}}$ in $\dot{\mathbb{M}}$, and the basic lemma may be proven by induction—or, more abstractly, follows from the initiality of the syntactic model. Indeed, choosing the relation $R_\beta$ for each base type $\beta$ above amounted to choosing an interpretation of the base types in a relational model. So long as the interpretation $[\![\beta]\!]^{\dot{\mathbb{M}}}$ of base types and constants in $\dot{\mathbb{M}}$ lie above their interpretation $[\![\beta]\!]^{\mathbb{M}}$ in $\mathbb{M}$—in the sense that $[\![\beta]\!]^{\mathbb{M}} = p([\![\beta]\!]^{\dot{\mathbb{M}}})$—then, because $p$ strictly preserves the model structure, we get $[\![A]\!]^{\mathbb{M}} = p([\![A]\!]^{\dot{\mathbb{M}}})$ for every type $A$.

**Example I.1.** Consider STLC with a single base type $\beta$, together with its usual interpretation in the cartesian closed category $\mathbf{Set}$ of sets and functions (see e.g. [17]). A natural choice of relations model is the category $\mathbf{Pred}$, which has objects pairs $(X, \overline{X})$ consisting of a set $X$ and a predicate $\overline{X} \subseteq X$, and morphisms $(X, \overline{X}) \to (Y, \overline{Y})$ given by functions $f : X \to Y$ preserving the relation: $x \in \overline{X} \implies f(x) \in \overline{Y}$. This category is cartesian closed, and the forgetful functor $p : \mathbf{Pred} \to \mathbf{Set}$ strictly preserves this structure (e.g. [15]).

Now choose a predicate $R_\beta \subseteq [\![\beta]\!]$ on the interpretation of $\beta$. This amounts to choosing $([\![\beta]\!], R_\beta) \in \mathbf{Pred}$ such that $p([\![\beta]\!], R_\beta) = [\![\beta]\!]$. Since $p$ strictly preserves cartesian closed structure, $p([\![M]\!]^{\mathbf{Pred}}) = [\![M]\!]^{\mathbf{Set}}$ for every term $M$. Setting $[\![A]\!]^{\mathbf{Pred}} := ([\![A]\!]^{\mathbf{Set}}, R_A)$, the family $\{R_A \mid A \text{ a type}\}$ is exactly a logical relation in the sense sketched above.

In recent years there has been extensive work extending fibrational techniques to Moggi-style monadic models of call-by-value (CBV) languages (e.g. [18], [19], [20], [21], [22], [23]). The next example is a simple instance of this framework.

**Example I.2.** We extend Example I.1 from STLC to $\lambda_{\mathrm{ml}}$. A semantic model is now a cartesian closed category equipped with a *strong monad* [24] (for an overview, see [16], [25]). We take the exception monad $\mathrm{Ex} := (-) + E$ on $\mathbf{Set}$, where $E$ is a fixed set of exception names. Since $\mathbf{Pred}$ has coproducts, which we denote $\sum_{i=1}^{n}(X_i, \overline{X_i}) := \left(\sum_{i=1}^{n} X_i, \bigoplus_{i=1}^{n} \overline{X_i}\right)$, for any subset $\overline{E} \subseteq E$ we get a (strong) exception monad $\dot{\mathrm{Ex}} := (-) + (E, \overline{E})$ which is strictly preserved by $p : \mathbf{Pred} \to \mathbf{Set}$. Hence $p$ preserves the interpretation of every monadic metalanguage term and, as above, we may define a logical relation: writing $[\![A]\!]^{\mathbf{Pred}} := ([\![A]\!], R_A)$ for all types $A$, we now get $R_{TA} := R_A \oplus \overline{E}$.

### B. Constructing fibrations for logical relations

We have seen that fibrations for logical relations encode logical relations. But how do we construct them in practice? There is a canonical way to do this. Building on Example I.1, say a fibration $p : \mathbb{E} \to \mathbb{B}$ is an *STLC fibration* if both $\mathbb{E}$ and $\mathbb{B}$ are cartesian closed and $p$ strictly preserves the cartesian closed structure: this is the appropriate form of fibrations for logical relations for STLC. One then observes the following.

**Lemma I.3** (e.g. [3, §4.3.1])**.** *Let $p : \mathbb{E} \to \mathbb{B}$ be an STLC fibration and $F : \mathbb{C} \to \mathbb{B}$ be any product-preserving functor. Then the pullback of $p$ along $F$ is also an STLC fibration.*

In various levels of generality, this has been called *sconing* (e.g. [10], [26], [3]), *(Artin) glueing* (e.g. [27], [28]), or *change-of-base* (e.g. [13], [3]). The objects of the pullback $\mathbb{P}$ are pairs $(C \in \mathbb{C}, X \in \mathbb{E})$ such that $FC = p(X)$. We think of this as pairing $C$ with a generalised 'relation' $X$. Thus, $\mathbb{P}$ is the model obtained by 'glueing' the models $\mathbb{C}$ and $\mathbb{E}$.

**Example I.4.** We define the category of binary predicates $\mathbf{BPred}$ by pullback from the fibration $p : \mathbf{Pred} \to \mathbf{Set}$, as shown below. The objects are triples $(X, Y, R \subseteq X \times Y)$ and morphisms $(f, g) : (X, Y, R) \to (X', Y', R')$ are pairs of set-maps preserving the relation: if $(x, y) \in R$ then $(fx, gy) \in R'$.

$$\begin{array}{ccc} \mathbf{BPred} & \longrightarrow & \mathbf{Pred} \\ {\scriptstyle q}\downarrow & \lrcorner & \downarrow{\scriptstyle p} \\ \mathbf{Set} \times \mathbf{Set} & \xrightarrow[\times]{} & \mathbf{Set} \end{array}$$

Since $q$ is an STLC fibration then—as in Example I.1—we obtain a notion of (binary) logical relations for STLC.

This approach extends smoothly to the effectful setting. Let us call a $\lambda_{\mathrm{ml}}$-*model* a pair $(\mathbb{C}, T)$ consisting of a cartesian closed category $\mathbb{C}$ and a strong monad $T$ (see e.g. [16], [25]). To capture the situation of Example I.2, where $p$ strictly preserves the interpretations of terms, our notion of fibration for logical relations for $\lambda_{\mathrm{ml}}$ must also preserve the monadic structure. Say that a $\lambda_{\mathrm{ml}}$-*fibration* from a $\lambda_{\mathrm{ml}}$-model $(\dot{\mathbb{C}}, \dot{T})$ to $(\mathbb{C}, T)$ is a fibration $p : \dot{\mathbb{C}} \to \mathbb{C}$ which strictly preserves both cartesian closed structure and monadic structure, so that $p \circ \dot{T} = T \circ p$ and for all $X \in \dot{\mathbb{C}}$ we have:

$$p(\dot{\mu}_X) = \mu_{pX} \qquad p(\dot{\eta}_X) = \eta_{pX} \qquad p(\dot{t}_X) = t_{pX}$$

The fibration $p$ in Example I.2 is a $\lambda_{\mathrm{ml}}$-fibration. More generally, say that a $\lambda_{\mathrm{ml}}$-*model morphism* $(\mathbb{B}, S) \to (\mathbb{C}, T)$ is a *strong monad morphism* $(F, \gamma)$ such that $F$ preserves products. Here $\gamma$ is a natural transformation $FS \Rightarrow TF$ which is compatible with the units, multiplications, and strengths (cf. [29, §3.6]). We write $\lambda\mathbf{ML}^{\mathrm{fo}}$ for the category of $\lambda_{\mathrm{ml}}$-models and their morphisms. The superscript emphasizes that we only require preservation of first-order structure.

Lemma I.3 now extends to the following observation, which is essentially due to Katsumata [20], [15] (see also [23]).

**Lemma I.5.** *For any $\lambda_{\mathrm{ml}}$-fibration $p$ and $\lambda\mathbf{ML}^{\mathrm{fo}}$-morphism $(F, \gamma)$, the pullback below exists in $\lambda\mathbf{ML}^{\mathrm{fo}}$. Moreover, $\dot{\mathbb{B}}$ is computed as the pullback in $\mathbf{Cat}$ and $q$ is a $\lambda_{\mathrm{ml}}$-fibration.*

$$\begin{array}{ccc} (\dot{\mathbb{B}}, \dot{S}) & \dashrightarrow & (\dot{\mathbb{C}}, \dot{T}) \\ {\scriptstyle q}\,\downarrow & & \downarrow{\scriptstyle p} \\ (\mathbb{B}, S) & \xrightarrow[(F,\gamma)]{} & (\mathbb{C}, T) \end{array}$$

Katsumata's $\top\top$-*lifting* [20] is a special case of this result. For any strong monad $T$ on a cartesian closed category and $T$-algebra $(R, r)$ there is a canonical strong monad morphism $\sigma$

from $T$ to the continuation monad $(- \Rightarrow R) \Rightarrow R$ with result type $R$. Setting $(F, \gamma) := (\mathrm{id}, \sigma)$ above yields $\top\top$-lifting.

Lemma I.5 is a powerful tool for constructing new semantic models. For example, it is at the heart of the characterisation of the definable morphisms in an effectful CBV model in [23]. It also provides a categorical framework for proving *effect simulation* results, which show two different ways of modelling the same effect satisfy a kind of bisimilarity property: see e.g. [30], [15]. There it is crucial that the pullback exists even though we only require preservation of first-order structure.

### C. This paper: from $\lambda_{\mathrm{ml}}$ to CBPV

This paper extends the theory outlined above from $\lambda_{\mathrm{ml}}$ to CBPV. As just outlined, this requires (1) a definition of fibration for logical relations, and (2) a theorem showing how to construct new models from old ones. Accordingly, we provide a definition of CBPV fibrations (Section IV) and show how to universally construct new models in a way paralleling the two lemmas above (Theorem V.1). To validate the theory, we show how to recover particular cases from the literature and give a version of $\top\top$-lifting for CBPV (Section V-A). We also show how to extend Katsumata's approach to effect simulation [15] from $\lambda_{\mathrm{ml}}$ to CBPV (Section VI).

The main technical obstacle is that we cannot simply define a CBPV fibration to be a functor which is both a fibration and preserves the CBPV model structure. Because CBPV is a rich language, a CBPV model consists of an adjunction enriched in presheaves over a certain category of values. Morphisms of CBPV models, therefore, make use of enriched functors. So the usual definition of fibration cannot be applied. Nor is it straightforward to simply write a definition by hand, because there are choices in how to define the universal property (see Remark IV.8). To obtain a principled definition of CBPV fibration, therefore, we must look elsewhere.

A natural first guess would be to use enriched fibrations. However, it is not clear this works. First, morphisms of CBPV models change the category of values, and hence the base of enrichment, so it is not clear what enriching base one should choose. Second, the theory of enriched fibrations [31], [32], [33] is motivated by quite different concerns, namely the correspondence with the Grothendieck construction.

Our solution is to turn to 2-category theory. 2-categories axiomatise the structure formed by categories, functors, and natural transformations. In particular, one can define a notion of fibration internally to any 2-category. To define CBPV fibrations, we first construct an appropriate 2-category (Section IV-A) and then identify its internal fibrations. A CBPV fibration is then such a fibration which also preserves the CBPV model structure (Section IV-D). As we outline in Section VIII, these steps—and indeed our main theorem—are particular instances of a general approach that applies likewise to models of STLC and $\lambda_{\mathrm{ml}}$. More generally, we conjecture that fibrations for logical relations typically arise as internal fibrations in an appropriate 2-category of models.

A benefit of our 2-categorical approach is that we can employ the well-developed theory of internal fibrations. This also provides canonical ways to construct important examples, such as versions of the codomain and subobject fibrations. To emphasise the value of this abstraction we also prove a relative conservativity result using an adaptation of Lafont's argument [34], [17] (Section VII). This shows that, in the language without sum types, function types are a conservative extension of the first-order language. We view this as laying the technical groundwork for future proofs of definability and normalisation in the style of Fiore [35].

*Related work:* We believe this work represents the first full denotational account of logical relations for CBPV. Unsurprisingly, however, logical relations-style arguments have been studied for CBPV as long as the language has existed. Indeed, Levy uses a logical relations argument in [1, §3.3] (see also [2, §2.3]), and the theory has been studied extensively from an operational perspective. For example, building on the work of [36] and prior work on higher-order mathematical operational semantics [37], [38], Goncharov, Tsampas & Henning [39] give a theory of logical relations for CBPV which includes sophisticated techniques such as step-indexing.

On the more denotational side, Kammar [40, Chapter 9] gives a detailed semantic study of logical relations for CBPV, and hence effect simulation, using algebra models. McDermott [41] presents a denotational perspective for the more sophisticated setting of graded CBPV. His account is slightly more general than Kammar's, in that he only asks for a certain form of lifting (Definition 4.3.4, ibid.). Motivated by this account, he also shows how to define logical relations in the traditional style as relations on sets of terms (Figure 5.5, ibid.). Azevedo de Amorim [42] presents logical relations for reasoning about the soundness of his expected cost semantics by phrasing it as an effect simulation property for a CBPV metalanguage. These developments are particular examples of our theory. In particular, our account covers not just algebra models but the more general adjunction models as well.

*Notation.* We assume some basic enriched category theory, as in e.g. [43, Chapter 6]. We write $\mathbb{C}^{\to}$ for the arrow category, which has objects maps in $\mathbb{C}$ and morphisms commuting squares, and $\mathrm{Sub}\,\mathbb{C}$ for the full subcategory of monomorphisms. In both cases we write $\mathrm{cod}$ for the codomain functor into $\mathbb{C}$. If $p : \mathbb{E} \to \mathbb{B}$ is a fibration, we denote the products and exponentials in $\mathbb{E}$ by $*$ and $\supset$. We assume throughout that all fibrations are split. Finally, because we work with enrichment in presheaf categories, size issues are relevant, especially in Section VII. Following [44], we handle these by assuming a hierarchy of universes of sets when required.

## II. 2-CATEGORY THEORY

We assume the basics of 2-category theory, in particular the definition of 2-categories, 2-functors, and transformations. For a detailed introduction, see e.g. [45], [46]. To fix notation, recall that a category is *cartesian* if it has finite products, and a cartesian category is *distributive* if it has finite coproducts and the canonical morphism $[X \times \mathrm{inj}_1, X \times \mathrm{inj}_2] : (X \times B) + (X \times C) \to X \times (B + C)$ is invertible. A *bicartesian* *closed category* (or *biCCC*) is a cartesian closed category with finite coproducts. $\mathbf{Cat}$ is then the 2-category of categories. $\mathbf{CartCat}$ and $\mathbf{DistCat}$ are the 2-categories of cartesian categories and distributive categories, respectively; the 1-cells are functors preserving the structure up to isomorphism. We call such functors *cartesian* and *bicartesian*, respectively. The 2-cells are all natural transformations. We write $\mathbf{CartCat}_{\mathrm{st}}$ and $\mathbf{DistCat}_{\mathrm{st}}$ for the sub-2-categories with the same objects and functors strictly preserving the structure.

The 2-categories of 2-functors $\mathcal{C} \to \mathcal{D}$ with strict (resp. pseudo / lax / oplax) natural transformations and modifications are denoted $[\mathcal{C}, \mathcal{D}]_{\mathrm{st}}, [\mathcal{C}, \mathcal{D}]_{\mathrm{ps}}, [\mathcal{C}, \mathcal{D}]_{\mathrm{lx}}$ and $[\mathcal{C}, \mathcal{D}]_{\mathrm{oplx}}$, respectively. Lax natural transformations are directed as follows:

$$\begin{array}{ccc} FC & \xrightarrow{Ff} & FC' \\ {\scriptstyle \sigma_C}\downarrow & \overset{\sigma_f}{\swarrow\!\!\!\swarrow} & \downarrow{\scriptstyle \sigma_{C'}} \\ GC & \xrightarrow[Gf]{} & GC' \end{array}$$

### A. Adjunctions and their morphisms

CBPV models are defined using adjunctions internal to a 2-category. We recall the definition.

**Definition II.1.** An *adjunction* in a 2-category $\mathcal{C}$ consists of 1-cells $f : A \leftrightarrows B : u$ together with 2-cells $\eta : \mathrm{id}_A \Rightarrow u \circ f$ and $\varepsilon : f \circ u \Rightarrow \mathrm{id}_B$ satisfying the usual triangle laws.

An adjunction in $\mathbf{Cat}$ is an adjunction in the usual sense. We shall also need morphisms between adjunctions. For this we shall see adjunctions as certain 2-functors and then define maps of adjunctions and their 2-cells as the corresponding transformations and modifications (cf. [47], [48], [49]).

Let $\mathbf{Adj}$ be the 2-category freely generated by the data of an adjunction, namely two objects $\bullet$ and $*$, 1-cells $f : \bullet \leftrightarrows * : u$, and 2-cells $\eta : \mathrm{id}_\bullet \Rightarrow u \circ f$ and $\varepsilon : f \circ u \Rightarrow \mathrm{id}_*$ satisfying the triangle laws. A 2-functor $\mathbf{Adj} \to \mathcal{C}$ is then equivalently an adjunction in $\mathcal{C}$. It follows immediately that any 2-functor preserves adjunctions.

**Definition II.2.** We write $\mathrm{Adj}(\mathcal{C})_w$ for the 2-functor category $[\mathbf{Adj}, \mathcal{C}]_w$, where $w \in \{\mathrm{st}, \mathrm{ps}, \mathrm{lx}, \mathrm{oplx}\}$. We call the 1-cells *strict / pseudo / lax / oplax adjunction morphisms* and the 2-cells *adjunction modifications*.

A lax adjunction map $(\ell : X \leftrightarrows Y : r) \to (f : A \leftrightarrows B : u)$ consists of 1-cells $m : X \to A$ and $n : Y \to B$ with 2-cells as shown below. The map is strict if $\alpha$ and $\beta$ are identities.

$$\begin{array}{ccc} X & \xrightarrow{\ell} & Y \\ {\scriptstyle m}\downarrow & \overset{\alpha}{\swarrow\!\!\!\swarrow} & \downarrow{\scriptstyle n} \\ A & \xrightarrow[f]{} & B \end{array} \qquad \begin{array}{ccc} Y & \xrightarrow{r} & X \\ {\scriptstyle n}\downarrow & \overset{\beta}{\swarrow\!\!\!\swarrow} & \downarrow{\scriptstyle m} \\ B & \xrightarrow[u]{} & A \end{array}$$

The 2-cells $\alpha$ and $\beta$ must be compatible with the units and counits, in the sense that the following two diagrams commute:

$$\begin{array}{ccc} n\ell r & \xrightarrow{n\varepsilon^{\ell,r}} & n \\ {\scriptstyle \alpha r}\downarrow & & \uparrow{\scriptstyle \varepsilon^{f,u}n} \\ fmr & \xrightarrow[f\beta]{} & fun \end{array} \qquad \begin{array}{ccc} \ell & \xrightarrow{\ell\eta^{\ell,r}} & mr\ell \\ {\scriptstyle \eta^{f,u}\ell}\downarrow & & \downarrow{\scriptstyle \beta\ell} \\ uf\ell & \xleftarrow[n\alpha]{} & un\ell \end{array} \qquad (1)$$

### B. Fibrations

We shall make extensive use of fibrations internal to a 2-category. These have been studied in great detail (e.g. [50], [51]); for a readable introduction to the theory, see [52].

**Definition II.3.** Let $\mathcal{C}$ be a 2-category. A *fibration* in $\mathcal{C}$ is a 1-cell $p : E \to B$ such that

1) For every $X \in \mathcal{C}$, the functor $p \circ (-) : \mathcal{C}(X, E) \to \mathcal{C}(X, B)$ is a fibration in $\mathbf{Cat}$, and
2) For every $h : Y \to X$ the following defines a morphism of fibrations, in the sense that cartesian liftings are preserved (see e.g. [52, Definition 3.1.1]):

$$\begin{array}{ccc} \mathcal{C}(X, E) & \xrightarrow{(-)\circ h} & \mathcal{C}(Y, E) \\ {\scriptstyle p\circ(-)}\downarrow & & \downarrow{\scriptstyle p\circ(-)} \\ \mathcal{C}(X, B) & \xrightarrow[(-)\circ h]{} & \mathcal{C}(Y, B) \end{array}$$

An *opfibration* is (somewhat unfortunately) defined to be a fibration in $\mathcal{C}^{\mathrm{co}}$. A *bifibration* is a 1-cell that is both a fibration and an opfibration.

Fibrations in a 2-category inherit many of the properties of fibrations in $\mathbf{Cat}$. For example, it is immediate that the identity is always a fibration and that fibrations are closed under composition. An (op)fibration in $\mathbf{Cat}$ is exactly an (op)fibration in the usual sense (see [53, Proposition 3.6]).

The next result shows that fibrations in the 2-category of algebras for a 2-monad are exactly fibrations in the base which preserve the structure. For an introduction to the powerful theory of algebraic structure on categories via 2-monads, see [54]. For the definition of algebras, see e.g. [55], [56].

**Proposition II.4.** *1) If $T$ is a 2-monad on a 2-category $\mathcal{C}$, and $(f, \overline{f})$ is a pseudomorphism of $T$-pseudoalgebras such that $f$ is a fibration in $\mathcal{C}$, then $(f, \overline{f})$ is a fibration in $T$-*Alg*.*
*2) Right adjoint 2-functors preserve fibrations.*
*Hence, $(f, \overline{f})$ is a fibration in $T$-*Alg* if and only if its underlying map is a fibration.*

This theorem covers $\mathbf{CartCat}, \mathbf{DistCat}$ and similar cases. To characterise fibrations in our particular example, we will need some simple 2-categorical limits.

**Definition II.5** (For details, see e.g. [57])**.** Let $(A \xrightarrow{f} C \xleftarrow{g} B)$ be a cospan in a 2-category $\mathcal{C}$. The *comma object* $f \downarrow g$ is the universal object with a 2-cell as shown below.

$$\begin{array}{ccc} f \downarrow g & \xrightarrow{q} & B \\ {\scriptstyle p}\downarrow & \underset{\lambda}{\Longrightarrow} & \downarrow{\scriptstyle g} \\ A & \xrightarrow[f]{} & C \end{array}$$

Comma objects in $\mathbf{Cat}$ are just comma categories. The *pullback* of $g$ along $f$ is defined analogously, except the square must be filled by an identity.

It follows from the corresponding fact in $\mathbf{Cat}$ that fibrations in any 2-category are closed under pullbacks.

**Example II.6.** The comma object $(F \downarrow G)$ in $\mathbf{DistCat}$ is the usual comma category. The product $(A, B, j) \times (A', B', j')$ is

$$F(A \times A') \xrightarrow{\cong} FA \times FA' \xrightarrow{j \times j'} GB \times GB' \xrightarrow{\cong} G(B \times B')$$

and coproducts are given similarly.

**Example II.7.** In any 2-category with comma objects the *arrow object* $C^{\rightarrow}$ on $C$ is defined to be the comma object $(\mathrm{id}_C \downarrow \mathrm{id}_C)$. In $\mathbf{Cat}$ this is exactly the arrow category $\mathbb{C}^{\rightarrow}$. The definition of comma objects also gives a map $\mathrm{cod} : C^{\rightarrow} \rightarrow C$; by [57, Theorem 2.11] this is always an opfibration.

**Example II.8.** $\mathbf{CartCat}$ and $\mathbf{DistCat}$ do not have all pullbacks. Indeed, the underlying 1-categories have products but do not have all equalizers, so cannot have pullbacks; hence the 2-categories cannot have them either. For example, the two maps $\{*\} \rightrightarrows (0 \xrightarrow{\cong} 1)$ from the terminal category to the walking isomorphism do not have an equalizer in either category. However, if $p$ is a fibration which strictly preserves products then the pullback along any map exists in $\mathbf{CartCat}$ (cf. [15, Proposition 6]). Similar remarks apply to $\mathbf{DistCat}$ when $p$ is also a bifibration and strictly preserves coproducts.

## III. Denotational models of CBPV

We refer to Levy's extensive works [2], [58], [59] for the syntax and semantics of CBPV. For definiteness, we use 'book CBPV', namely the basic language and complex values described in Chapters 2 & 3 of [2]. In particular, we only ask for finite sum types and finite product types.

There are several equivalent ways to phrase the data of a CBPV model (see [1, Chapter 11] and [1, §15.1]), so we make our choice explicit. The basic data is a *locally indexed adjunction*. We refer to [2, §9.3] for the details on locally $\mathbb{C}$-indexed categories, locally $\mathbb{C}$-indexed functors and locally $\mathbb{C}$-indexed transformations for a cartesian category $(\mathbb{C}, \times, 1)$, and write $\mathbb{C}$-$\mathbf{LInd}$ for the 2-category these form. $\mathbb{C}$-$\mathbf{LInd}$ is equivalently the 2-category $\widehat{\mathbb{C}}$-$\mathbf{Cat}$ of categories enriched in the presheaf category $(\widehat{\mathbb{C}}, \times, 1)$ (see e.g. [60, §1.2]).

We denote locally $\mathbb{C}$-indexed categories in calligraphic font, as $\mathcal{C}, \mathcal{D}, \ldots$. Maps over $c \in \mathbb{C}$ are denoted $A \underset{c}{\rightarrow} B$ and the category of maps over $c$ by $\mathcal{C}_c$. The reindexing functor $\mathcal{C}_d \rightarrow \mathcal{C}_c$ induced by $\rho : c \rightarrow d$ is denoted, in a slight departure from Levy's notation, by $(-) \lhd \rho$.

**Example III.1.** For a biCCC $\mathbb{C}$ the locally $\mathbb{C}$-indexed category $\mathsf{self}\,\mathbb{C}$ has objects as in $\mathbb{C}$ and hom-presheaves $(\mathsf{self}\,\mathbb{C})_C(A, B) := \mathbb{C}(C \times A, B)$.

**Definition III.2.** A *locally $\mathbb{C}$-indexed adjunction* is an adjunction in $\mathbb{C}$-$\mathbf{LInd}$. This is a pair of locally $\mathbb{C}$-indexed functors $F : \mathcal{C} \leftrightarrows \mathcal{D} : U$ with locally $\mathbb{C}$-indexed transformations $\eta : \mathrm{id}_{\mathcal{C}} \Rightarrow UF$ and $\varepsilon : \mathrm{id}_{\mathcal{D}} \Rightarrow FU$ satisfying the usual triangle equalities as composites in $\mathcal{D}_1$.

A CBPV model is a locally $\mathbb{C}$-indexed adjunction which also models the products, sums, and function types.

**Definition III.3** (e.g. [58, §5])**.** Let $\mathbb{C}$ be a distributive category and $\mathcal{C}$ be a locally $\mathbb{C}$-indexed category.

1) $\mathcal{C}$ has *(finite) products* if for every finite family of objects $B_1, \ldots, B_n$ there exists an object $\prod_{i=1}^n B_i \in \mathcal{C}$ and arrows $\pi_i : \prod_{i=1}^n B_i \underset{1}{\rightarrow} B_i$ inducing an isomorphism $\mathcal{C}_c(A, \prod_{i=1}^n B_i) \cong \prod_{i=1}^n \mathcal{C}_c(A, B_i)$.
2) $\mathcal{C}$ has *($\mathbb{C}$-indexed) powers* if for every $c \in \mathbb{C}$ and $B \in \mathcal{C}$ there exists an object $c \Rightarrow B \in \mathcal{C}$ and an arrow $\mathrm{eval} : (c \Rightarrow B) \underset{c}{\rightarrow} B$ inducing an isomorphism $\mathcal{C}_{b \times c}(A, B) \cong \mathcal{C}_b(A, c \Rightarrow B)$.
3) $\mathbb{C}$'s coproducts are *(finitely) distributive in* $\mathcal{C}$ if for all $a, b_1, \ldots, b_n \in \mathbb{C}$ and $A, B \in \mathcal{C}$ the following is invertible:

$$\mathcal{C}_{a \times \sum_{i=1}^n b_i}(A, B) \rightarrow \prod_{i=1}^n \mathcal{C}_{a \times b_i}(A, B)$$
$$f \mapsto \left(f \lhd (\mathrm{id}_a \times \mathrm{inj}_i)\right)_{i=1,\ldots,n}$$

A CBPV model is now defined by taking the appropriate universally-defined structure for each CBPV construct.

**Definition III.4** (e.g. [58, §5])**.** A *CBPV model* consists of a distributive category $\mathbb{C}$ and a locally $\mathbb{C}$-indexed adjunction $F : \mathsf{self}\,\mathbb{C} \leftrightarrows \mathcal{C} : U$ such that $\mathcal{C}$ has products and powers, and the coproducts in $\mathbb{C}$ are distributive in $\mathcal{C}$.

Value terms $\Gamma \vdash^{\mathsf{v}} V : A$ are interpreted as maps in $\mathbb{C}$, i.e. as elements of $\mathbb{C}(\llbracket \Gamma \rrbracket, \llbracket A \rrbracket)$. Computation terms $\Gamma \vdash^{\mathsf{c}} M : \overline{B}$ are interpreted in $\mathbb{C}(\llbracket \Gamma \rrbracket, U\llbracket \overline{B} \rrbracket)$. Stacks $\Gamma \mid \overline{B} \vdash^{\mathsf{k}} K : \overline{C}$ are interpreted in $\mathcal{C}_{\llbracket \Gamma \rrbracket}(\llbracket \overline{B} \rrbracket, \llbracket \overline{C} \rrbracket)$.

**Remark III.5.** Because we interpret computations in $\mathbb{C}(\llbracket \Gamma \rrbracket, U\llbracket \overline{B} \rrbracket)$ rather than the isomorphic $\mathcal{C}_{\llbracket \Gamma \rrbracket}(F1, \llbracket \overline{B} \rrbracket)$, the interpretation of $\mathsf{force}$ is invisible: it is the identity. Operationally, this reflects the fact that forcing a term does not change its behaviour.

For the sake of exposition, in this paper we will focus on relatively simple classes of CBPV models. We refer to [58, §4.4] and [2] for the details of these and many other models.

**Example III.6** ([58, §7])**.** The syntax of CBPV forms a model. For any *signature* $\mathcal{S}$ of value base types, computation base types, and operations one may freely generate a *theory* and its classifying syntactic model $\mathbf{Syn}_{\mathcal{S}}$.

**Example III.7** (Algebra models)**.** Let $\mathbb{C}$ be a biCCC and $T$ a strong monad on $\mathbb{C}$. The category $\mathbb{C}^T$ of $T$-algebras becomes a locally $\mathbb{C}$-indexed category $\mathcal{EM}(T)$ with maps $(A, a) \underset{c}{\rightarrow} (B, b)$ the right-linear morphisms $c \times A \rightarrow B$. The free–forgetful adjunction then becomes a CBPV model $F^T : \mathsf{self}\,\mathbb{C} \leftrightarrows \mathcal{EM}(T) : U^T$.

**Example III.8** (Storage)**.** Let $(\mathbb{C}, \times, 1, \Rightarrow, 0, +)$ be a biCCC and $S \in \mathbb{C}$ be an object of "states". The adjunction $(-) \times S \dashv S \Rightarrow (-)$ defines a CBPV model $\mathsf{self}\,\mathbb{C} \leftrightarrows \mathsf{self}\,\mathbb{C}$.

## IV. CBPV fibrations

In this section we introduce CBPV fibrations. Our approach closely follows the pattern for monadic models of CBV in the style of Moggi [61], [16] pioneered by Katsumata [20], [14], [15] and others (e.g. [19], [18], [21], [23]).

We begin by defining a 2-category of locally indexed categories $\mathbf{LInd}$ (Section IV-A). This will play the role for

CBPV that **CartCat** plays for STLC: it collects together the models of the basic judgements for contexts, so we can isolate models of the full language as a sub-2-category. With this in mind, we shall define *locally indexed fibrations* to be fibrations internal to **LInd** (Section IV-B), and define a CBPV fibration to be a morphism of locally indexed adjunctions which strictly preserves the model structure and is componentwise a locally indexed fibration (Section IV-D). Along the way we shall also see that the general theory leads directly to a variety of examples, in the form of locally indexed versions of the codomain and subobject fibrations.

### A. The 2-category LInd

We begin by defining the 2-category of locally indexed categories. The objects are pairs $(\mathbb{C}, \mathcal{C})$ consisting of a cartesian category $\mathbb{C}$ and a locally $\mathbb{C}$-indexed category $\mathcal{C}$.

We want morphisms between CBPV models which may have different interpretations of values, so our 1-cells can't just be morphisms in $\mathbb{C}$-**LInd** for a fixed $\mathbb{C}$. Instead, observe that any cartesian functor $f : \mathbb{C} \to \mathbb{D}$ induces a product-preserving functor $f^* := (-) \circ f : \widehat{\mathbb{D}} \to \widehat{\mathbb{C}}$ and hence, by *change of base* (e.g. [43, §6.4]), a 2-functor $\mathbb{D}$-**LInd** $\to$ $\mathbb{C}$-**LInd** we also denote by $f^*$. Explicitly, if $\mathcal{D} \in \mathbb{D}$-**LInd** then $f^*\mathcal{D}$ has the same objects but hom-presheaves defined by $(f^*\mathcal{D})_c := \mathcal{D}_{fc}$. Composition and identities are as in $\mathcal{D}$, and reindexing along $\rho$ in $f^*\mathcal{D}$ is the reindexing along $f(\rho)$ in $\mathcal{D}$.

We may now define a *locally indexed functor* $(f, F) : (\mathbb{C}, \mathcal{C}) \to (\mathbb{D}, \mathcal{D})$ to be a cartesian functor $f : \mathbb{C} \to \mathbb{D}$ together with a locally $\mathbb{C}$-indexed functor $F : \mathcal{C} \to f^*\mathcal{D}$. This smoothly handles reindexing: a map $k \in \mathcal{C}_c(C, C')$ is sent to a map $Fk \in (f^*\mathcal{D})_c(FC, FC') = \mathcal{D}_{fc}(FC, FC')$.

The 2-cells are defined similarly. Both change-of-base and the passage from functors $f$ between categories to functors $f^*$ between presheaf categories are 2-functorial [62], [63] so every natural transformation $\gamma : f \Rightarrow g : \mathbb{C} \to \mathbb{D}$ defines a strict natural transformation $\gamma^* : g^* \Rightarrow f^* : \mathbb{D}$-**LInd** $\to$ $\mathbb{C}$-**LInd**. The component $(\gamma^*)_{\mathcal{C}}$ at $\mathcal{C} \in \mathbb{D}$-**LInd** is the identity-on-objects $\mathbb{C}$-**LInd**-functor which reindexes along $\gamma$:

$$(\gamma^*)_{\mathcal{C}}(c) := (g^*\mathcal{C})_c = \mathcal{C}_{gc} \xrightarrow{(-) \lhd \gamma_c} \mathcal{C}_{fc} = (f^*\mathcal{C})_c$$

We define a *locally indexed 2-cell* $(f, F) \Rightarrow (g, G) : (\mathbb{C}, \mathcal{C}) \to (\mathbb{D}, \mathcal{D})$ to be a natural transformation $\gamma : f \Rightarrow g$ together with a locally $\mathbb{C}$-indexed transformation $\overline{\gamma} : F \Rightarrow (\alpha^*)_{\mathcal{D}} \circ G$. Concretely, $\overline{\gamma}$ is a family $\left(\overline{\gamma}_C : FC \underset{1}{\to} GC\right)_{C \in \mathcal{C}}$ such that for any $k : C \underset{c}{\to} C'$ in $\mathcal{C}$ the following diagram commutes:

$$\begin{array}{ccc} FC & \xrightarrow{F(k)} & FC' \\ {\scriptstyle \overline{\gamma}_C \lhd !_{fc}} \big\downarrow & f(c) & \big\downarrow {\scriptstyle \overline{\gamma}_{C'} \lhd !_{fc}} \\ GC & \xrightarrow[G(k) \lhd \gamma_c]{} & GC' \end{array} \tag{2}$$

**Notation IV.1.** We henceforth adopt the notation used in (2): when writing a diagram in a locally indexed category, we indicate the index by writing it in the centre of the shape.

**Definition IV.2.** We write **LInd** for the 2-category of locally indexed categories, locally indexed functors, and locally indexed transformations.

**Remark IV.3.** This definition is canonical: abstractly, **LInd** is the 2-Grothendieck construction [64], [65] of the 2-functor $K$ : $\mathbf{CartCat}^{\mathrm{coop}} \to 2\text{-}\mathbf{CAT}$ which acts on objects by $K(\mathbb{C}) :=$ $\mathbb{C}$-**LInd** and on 1-cells and 2-cells by change of base.

### B. Locally indexed fibrations

We now characterise the fibrations internal to **LInd**. We do this using [57, Theorem 2.7], so we need to construct comma objects. The following two constructions follow directly from Remark IV.3 and the fact that, as in the 1-categorical setting, the 2-Grothendieck construction for $K : \mathcal{C}^{\mathrm{coop}} \to 2\text{-}\mathbf{CAT}$ has those limits which exist in $\mathcal{C}$ and each $K(A)$ and are preserved by every $K(f)$ (cf. [53, §4]).

**Construction IV.4.** The comma object $\left(f \downarrow g, F \downarrow G\right)$ of a cospan $(\mathbb{A}, \mathcal{A}) \xrightarrow{(f,F)} (\mathbb{C}, \mathcal{C}) \xleftarrow{(g,G)} (\mathbb{B}, \mathcal{B})$ in **LInd** is defined as follows. The indexing category is the comma object in **CartCat** (Example II.6). The objects in $(F \downarrow G)$ are triples $\left(A \in \mathcal{A}, B \in \mathcal{B}, k : FA \underset{1}{\to} GB\right)$, while maps $(A, B, k) \underset{j}{\to} (A', B', k')$ over $j : fa \to gb$ are pairs $\left(u : A \underset{a}{\to} A', v : B \underset{b}{\to} B'\right)$ such that following commutes:

$$\begin{array}{ccc} FA & \xrightarrow{F(u)} & FA' \\ {\scriptstyle k \lhd !_{fa}} \big\downarrow & f(a) & \big\downarrow {\scriptstyle k' \lhd !_{fa}} \\ GB & \xrightarrow[G(v) \lhd j]{} & GB' \end{array}$$

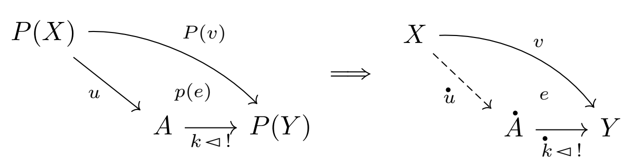
Composition, identities, and reindexing are componentwise.

Similarly, the pullback of the cospan above exists in **LInd** when the pullback $\mathbb{A} \times_{\mathbb{C}} \mathbb{B}$ exists in **CartCat**, and is given by restricting $F \downarrow G$ to the pairs $(A, B)$ such that $F(A) = G(B)$.

**Construction IV.5.** The product $(\mathbb{C}, \mathcal{C}) \times (\mathbb{D}, \mathcal{D})$ in **LInd** is the $(\mathbb{C} \times \mathbb{D})$-indexed category $\mathcal{C} \times \mathcal{D}$ with objects pairs $(C \in \mathcal{C}, D \in \mathcal{D})$ and hom-presheaves $(\mathcal{C} \times \mathcal{D})_{(c,d)} := \mathcal{C}_c \times \mathcal{D}_d$.

Now, working through the condition in [57, Theorem 2.7] yields the following. To simplify notation we elide the iso $p(1) \cong 1$ given by the fact $p$ is cartesian.

**Proposition IV.6.** *A locally indexed functor $(p, P) : (\mathbb{E}, \mathcal{E}) \to (\mathbb{B}, \mathcal{B})$ is a fibration in* **LInd** *if and only if $p$ is a fibration and $P$ satisfies the following lifting property for any $k : A \underset{1}{\to} PY$:*




*Thus, there exists $\dot{A} \in \mathcal{E}$ and $\dot{k} : \dot{A} \underset{1}{\to} Y$ in $\mathcal{E}$ such that, for any triangle in $p^*\mathcal{B}$ as on the left above, there exists a unique lift $\dot{u}$ making the triangle on the right commute in $\mathcal{E}$.*

**Definition IV.7.** A *locally indexed fibration / opfibration / bifibration* is a fibration / opfibration / bifibration in $\mathbf{LInd}$.

**Remark IV.8.** A priori there are many possible choices for defining locally indexed fibrations. If one were giving a definition by hand, one might be tempted allow maps over any index to have a cartesian lift, or specify that the unique arrow $\dot{u}$ must be over 1. Because it is derived from the mathematical theory, our definition is canonical and immediately satisfies useful properties like closure under pullback.

We now use our mathematical framework to define locally indexed versions of the core building blocks for constructing new models, namely the codomain and subobject fibrations. The construction of the codomain opfibration follows directly from Construction IV.4 and Example II.7.

**Construction IV.9.** The *locally indexed arrow category* of $(\mathbb{C}, \mathcal{C}) \in \mathbf{LInd}$ is the $\mathbb{C}^{\to}$-indexed category with objects arrows $A \underset{1}{\to} B$ in $\mathcal{C}_1$. There is a canonical locally indexed *codomain* opfibration $(\mathrm{cod}, \mathrm{cod}) : (\mathbb{C}^{\to}, \mathcal{C}^{\to}) \to (\mathbb{C}, \mathcal{C})$.

This leads naturally to the subobject fibration.

**Definition IV.10.** We write $\mathrm{Sub}(\mathcal{C})$ for the $\mathrm{Sub}(\mathbb{C})$-indexed category obtained by restricting the objects of $\mathcal{C}^{\to}$ to arrows $A \underset{1}{\to} B$ that are monic in $\mathcal{C}_1$. Since $\mathrm{Sub}(\mathbb{C})$ is closed under products, reindexing is as in $\mathcal{C}^{\to}$.

In $\mathbf{Cat}$, the codomain functor is a fibration if and only if the base category $\mathbb{C}$ has pullbacks; then $\mathrm{cod} : \mathrm{Sub}\,\mathbb{C} \to \mathbb{C}$ is also a fibration. A corresponding fact is true here.

**Definition IV.11.** A locally indexed category $(\mathbb{C}, \mathcal{C})$ *has locally indexed pullbacks* if $\mathcal{C}_1$ has pullbacks, and these are preserved by $(-) \lhd !_c$ for every $c \in \mathbb{C}$.

**Lemma IV.12.** *Let $(\mathbb{C}, \mathcal{C})$ be a locally indexed category. The locally indexed codomain functor is a locally indexed bifibration if and only if $\mathcal{C}$ has locally indexed pullbacks. In this situation, the fibration structure restricts to make $(\mathrm{Sub}\,\mathbb{C}, \mathrm{Sub}\,\mathcal{C}) \to (\mathbb{C}, \mathcal{C})$ a fibration as well.*

**Example IV.13.** Suppose $\mathbb{C}$ is finitely complete. Then $\mathsf{self}\,\mathbb{C}$ has locally indexed pullbacks. Moreover, for any strong monad $T$ on $\mathbb{C}$ the locally $\mathbb{C}$-indexed category $\mathcal{EM}(T)$ of $T$-algebras also has locally indexed pullbacks.

### C. The 2-category of CBPV models

With locally indexed fibrations in hand, it remains to define a 2-category $\mathbf{CBPV}^{\mathrm{fo}}_{\mathrm{lx}}$ of CBPV models. In this section we isolate $\mathbf{CBPV}^{\mathrm{fo}}_{\mathrm{lx}}$ as a sub-2-category of $\mathrm{Adj}(\mathbf{LInd})_{\mathrm{lx}}$. In Section IV-D we will combine this with Definition IV.7 to define CBPV fibrations. We begin by defining preservation-of-structure.

**Definition IV.14.** A locally $\mathbb{C}$-indexed functor $F : \mathcal{C} \to \mathcal{D}$ *preserves finite products* if for any $n \in \mathbb{N}$ the canonical map $\langle F\pi_i \rangle_i : F(\prod_{i=1}^n C_i) \underset{1}{\to} \prod_{i=1}^n F(C_i)$ is an isomorphism in $\mathcal{D}_1$. It preserves products *strictly* if all the structure is preserved on the nose, so that $F(\prod_{i=1}^n C_i) = \prod_{i=1}^n FC_i$, $F\pi_i = \pi_i$, and $F(\langle f_i \rangle_i) = \langle Ff_i \rangle_i$. The definition of (strict) preservation of powers is likewise.

The 1-cells in $\mathbf{CBPV}^{\mathrm{fo}}_{\mathrm{lx}}$ are the ones we shall pull back along in our lifting theorem (Theorem V.1), so we will only ask for preservation of first-order structure. This matches the situation for CBV, where one only needs preservation of products to construct new models from old (recall Lemma I.5).

We have one more constraint to impose. A CBPV model is special kind of object in $\mathrm{Adj}(\mathbf{LInd})_{\mathrm{lx}}$: the domain of the left adjoint is of the form $\mathsf{self}\,\mathbb{C}$ and the adjunction is over a single base. We therefore want 1-cells in $\mathbf{CBPV}^{\mathrm{fo}}_{\mathrm{lx}}$ to be maps in $\mathrm{Adj}(\mathbf{LInd})_{\mathrm{lx}}$ whose first component is determined by their action on the values. For this we use the following lemma.

**Lemma IV.15.** *The map $\mathbb{C} \mapsto \mathsf{self}\,\mathbb{C}$ extends to 2-functors $\mathbf{CartCat} \to \mathbf{LInd}$ and $\mathbf{DistCat} \to \mathbf{LInd}$ which preserve products, comma objects, pullbacks, and fibrations whose underlying functor is strict cartesian (resp. cartesian and cocartesian). We denote these both by $\mathsf{self}$.*

We now give the definition for $w \in \{\mathrm{lx}, \mathrm{oplx}, \mathrm{ps}\}$. The objects of $\mathbf{CBPV}^{\mathrm{fo}}_{w}$ are CBPV models $(\mathbb{C}, \mathcal{C}, F, U)$. A 1-cell $(\mathbb{C}, \mathcal{C}, F^{\mathcal{C}}, U^{\mathcal{C}}) \to (\mathbb{D}, \mathcal{D}, F^{\mathcal{D}}, U^{\mathcal{D}})$ consists of

- A bicartesian functor $h : \mathbb{C} \to \mathbb{D}$,
- A locally $\mathbb{C}$-indexed functor $H : \mathcal{C} \to h^*\mathcal{D}$, and
- Locally $\mathbb{C}$-indexed transformations $\alpha$ and $\beta$,

such that $H$ preserves products and $\big(\mathsf{self}\,h, H, (\mathrm{id}, \alpha), (\mathrm{id}, \beta)\big)$ is a 1-cell in $\mathrm{Adj}(\mathbf{LInd})_w$. A 2-cell $(h, H, \alpha, \beta) \Rightarrow (h', H', \alpha', \beta')$ is a pair $\big(\gamma : h \Rightarrow h', \overline{\gamma} : H \Rightarrow H'\big)$ such that $(\mathsf{self}\,\gamma, \overline{\gamma})$ is a 2-cell in $\mathrm{Adj}(\mathbf{LInd})_w$.

In the lax case this means that 1-cells look like

$$\begin{array}{ccccc} \mathsf{self}\,\mathbb{C} & \xrightarrow{(\mathrm{id}_{\mathbb{C}}, F^{\mathcal{C}})} & \mathcal{C} & \xrightarrow{(\mathrm{id}_{\mathbb{C}}, U^{\mathcal{C}})} & \mathsf{self}\,\mathbb{C} \\ {\scriptstyle \mathsf{self}\,h}\downarrow & (\mathrm{id}, \alpha) \Downarrow & \downarrow{\scriptstyle (h,H)} & \Downarrow (\mathrm{id}, \beta) & \downarrow{\scriptstyle \mathsf{self}\,h} \\ \mathsf{self}\,\mathbb{D} & \xrightarrow[(\mathrm{id}_{\mathbb{D}}, F^{\mathcal{D}})]{} & \mathcal{D} & \xrightarrow[(\mathrm{id}_{\mathbb{D}}, U^{\mathcal{D}})]{} & \mathsf{self}\,\mathbb{D} \end{array} \tag{3}$$

so for each $c \in \mathbb{C}$ and $C \in \mathcal{C}$ we have arrows

$$\alpha_c : (HF^{\mathcal{C}})c \underset{1}{\to} (F^{\mathcal{D}}h)c \qquad \beta_C : (hU^{\mathcal{C}})C \underset{1}{\to} (U^{\mathcal{D}}H)C$$

natural in the sense of (2) and satisfying the compatibility axioms (1) as composites over 1.

### D. Defining CBPV fibrations

We can finally define CBPV fibrations as locally indexed fibrations which strictly preserve structure. Note that we require $p$ to be a *bi*fibration because of the sum types; this is needed so that pullbacks along $p$ exist in $\mathbf{DistCat}$. Without sum types, it would be sufficient to ask for just a fibration.

**Definition IV.16.** A $\mathbf{CBPV}^{\mathrm{fo}}_{\mathrm{lx}}$ 1-cell $(h, H, \alpha, \beta)$ is *strict* if

1) $h$ strictly preserves bicartesian structure,
2) $H$ strictly preserves products and powers, and
3) $(h, H)$ is a 1-cell in $\mathrm{Adj}(\mathbf{LInd})_{\mathrm{st}}$, i.e. $\alpha$ and $\beta$ are both the identity.

This is a *CBPV (op)fibration* if $(h, H)$ is a locally indexed (op)fibration and $h$ is a bifibration.

**Example IV.17** (Recall Example III.6)**.** Levy's proof of [58, Proposition 7.3] essentially shows that for any CBPV model $(\mathbb{C}, \mathcal{C}, F, U)$ with an interpretation of the base types and operations in a signature $\mathcal{S}$ there exists a strict $\mathbf{CBPV}^{\mathrm{fo}}_{\mathrm{lx}}$ 1-cell $\mathbf{Syn}_{\mathcal{S}} \to (\mathbb{C}, \mathcal{C}, F, U)$ extending the interpretation of $\mathcal{S}$. Moreover, this is unique up to isomorphism.

By Lemma IV.15, a CBPV fibration is a 1-cell in $[\mathbf{Adj}, \mathbf{LInd}]_{\mathrm{st}}$ which is componentwise a fibration.

Turning now to examples, one simple class of CBPV fibrations comes via monad liftings. A particular instance of the following result has been studied by Kammar [40, §9.2], who constructs CBPV fibrations over $\mathbf{Set}$ using the free lifting.

**Lemma IV.18.** *Let $p : (\dot{\mathbb{C}}, \dot{T}) \to (\mathbb{C}, T)$ be a $\lambda_{\mathrm{ml}}$-fibration. Then $p$ extends to a fibration $\widetilde{p} : \dot{\mathbb{C}}^{\dot{T}} \to \mathbb{C}^{T}$, and this makes $(p, \widetilde{p}) : \mathcal{EM}(\dot{T}) \to \mathcal{EM}(T)$ a CBPV fibration.*

A further set of examples corresponds to the classical fact that, if $\mathbb{C}$ is a cartesian closed category with pullbacks, then the codomain fibration over $\mathbb{C}$ is an STLC fibration. Our corresponding result is the following.

**Lemma IV.19.** *Let $\mathbb{C}$ be a cartesian category with pullbacks and $\mathcal{C}$ be a locally $\mathbb{C}$-indexed category with locally indexed pullbacks, products, and powers. Then $\mathcal{C}^{\to}$ and $\mathrm{Sub}\,\mathcal{C}$ both have products and powers, and the codomain locally indexed functors strictly preserve this structure.*

Now, the $(-)^{\to}$ operation is 2-functorial so from a CBPV model $(\mathbb{C}, \mathcal{C}, F, U)$ we obtain a lifted locally $\mathbb{C}^{\to}$-indexed adjunction $(\mathbb{C}^{\to}, \mathcal{C}^{\to}, F^{\to}, U^{\to})$. Combining the preceding lemma with the observation that $(\mathsf{self}\,\mathbb{C})^{\to} \cong \mathsf{self}\,(\mathbb{C}^{\to})$ in $\mathbb{C}^{\to}$-$\mathbf{LInd}$, we obtain the following.

**Proposition IV.20.** *Let $(\mathbb{C}, \mathcal{C}, F, U)$ be a CBPV model such that $\mathbb{C}$ has pullbacks and $\mathcal{C}$ has locally indexed pullbacks. Then the codomain functor $(\mathrm{cod}, \mathrm{cod}) : (\mathbb{C}^{\to}, \mathcal{C}^{\to}) \to (\mathbb{C}, \mathcal{C})$ is a CBPV fibration.*

The only obstacle to applying a similar argument to the subobject fibration is that the left adjoint $F$ may not preserve monics, and therefore may not restrict to a locally indexed functor $\mathsf{self}\,(\mathrm{Sub}\,\mathbb{C}) \to \mathrm{Sub}\,\mathcal{C}$ (right adjoints always preserve monics). We expect this can be rectified by taking an appropriate factorisation system, in the style of [18], [66], [19], [21]. For reasons of space, however, we content ourselves to the case when $F$ preserves monics. This turns out to be common: for example, it applies to the storage model of Example III.8), the erratic choice and continuation models of [58, §5.7], and any algebra model over $\mathbf{Set}$ (see [67, p. 89-90]).

**Corollary IV.21.** *In the situation of Proposition IV.20, if $F_1$ also preserves monics then the codomain functor $(\mathrm{cod}, \mathrm{cod}) : (\mathrm{Sub}\,\mathbb{C}, \mathrm{Sub}\,\mathcal{C}) \to (\mathbb{C}, \mathcal{C})$ is a CBPV fibration.*

**Example IV.22.** Consider the model of Example III.8 in the case where $\mathbb{C}$ has pullbacks. Since both the left and right adjoints preserve monics, this lifts to a storage model $(-) * \overline{S} \dashv \overline{S} \supset (-)$ on $\mathrm{Sub}\,\mathbb{C}$ for any subobject $\overline{S} \rightarrowtail S$. Then the subobject locally indexed fibration is a CBPV fibration; Corollary IV.21 is the case where $\overline{S} := (S \xrightarrow{\mathrm{id}} S)$.

## V. A lifting theorem for CBPV models

As we saw in Section I-B, for applications we want a universal way to construct new fibrations for logical relations from old ones. In this section we present our central technical result, which shows this is possible for CBPV in a manner paralleling that for STLC and $\lambda_{\mathrm{ml}}$ (cf. Lemmas I.3 and I.5).

**Theorem V.1.** *Let $(p, P)$ be a CBPV fibration and $(h, H, \alpha, \beta)$ be a $\mathbf{CBPV}^{\mathrm{fo}}_{\mathrm{lx}}$ 1-cell. Then the pullback shown below exists in $\mathbf{CBPV}^{\mathrm{fo}}_{\mathrm{lx}}$ and $(q, Q)$ is a CBPV fibration.*

$$\begin{array}{ccc} (\dot{\mathbb{C}}, \dot{\mathcal{C}}, F^{\dot{\mathcal{C}}}, U^{\dot{\mathcal{C}}}) & \xrightarrow{(\dot{h}, \dot{H}, \dot{\alpha}, \dot{\beta})} & (\dot{\mathbb{D}}, \dot{\mathcal{D}}, F^{\dot{\mathcal{D}}}, U^{\dot{\mathcal{D}}}) \\ {\scriptstyle (q,Q)} \downarrow & & \downarrow {\scriptstyle (p,P)} \\ (\mathbb{C}, \mathcal{C}, F^{\mathcal{C}}, U^{\mathcal{C}}) & \xrightarrow[(h, H, \alpha, \beta)]{} & (\mathbb{D}, \mathcal{D}, F^{\mathcal{D}}, U^{\mathcal{D}}) \end{array} \tag{4}$$

This theorem is an instance of a general fact about lax transformations: see Section VIII. Here we sketch the concrete construction. First, $q$ and $Q$ are defined as pullbacks in $\mathbf{DistCat}$ and $\mathbf{LInd}$ respectively; these exist because $p$ is strict and a bifibration (Example II.8 and Construction IV.9).

$$\begin{array}{ccc} \dot{\mathbb{C}} & \xrightarrow{\dot{h}} & \dot{\mathbb{D}} \\ {\scriptstyle q} \downarrow & \lrcorner & \downarrow {\scriptstyle p} \\ \mathbb{C} & \xrightarrow[h]{} & \mathbb{D} \end{array} \qquad \begin{array}{ccc} (\dot{\mathbb{C}}, \dot{\mathcal{C}}) & \xrightarrow{(\dot{h}, \dot{H})} & (\dot{\mathbb{D}}, \dot{\mathcal{D}}) \\ {\scriptstyle (q,Q)} \downarrow & \lrcorner & \downarrow {\scriptstyle (p,P)} \\ (\mathbb{C}, \mathcal{C}) & \xrightarrow[(h,H)]{} & (\mathbb{D}, \mathcal{D}) \end{array} \tag{5}$$

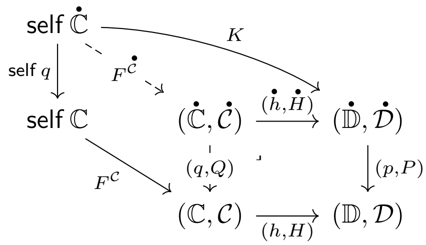
An argument similar to that for cartesian closed structure (e.g. [15, Proposition 6]) shows $(\dot{\mathbb{C}}, \dot{\mathcal{C}})$ has products and powers. We define the adjunction $(\mathsf{self}\,\dot{\mathbb{C}} \leftrightarrows \dot{\mathcal{C}})$ and the 2-cells $\dot{\alpha}$ and $\dot{\beta}$ in (4) using the universal property of the fibrations. Observe first that the following diagram commutes because $\mathsf{self}$ is a 2-functor and $(p, P)$ is a strict adjunction morphism:

$$\begin{array}{ccccc} \mathsf{self}\,\dot{\mathbb{C}} & \xrightarrow{\mathsf{self}\,\dot{h}} & \mathsf{self}\,\dot{\mathbb{D}} & \xrightarrow{F^{\dot{\mathcal{D}}}} & (\dot{\mathbb{D}}, \dot{\mathcal{D}}) \\ {\scriptstyle \mathsf{self}\,q} \downarrow & & \downarrow {\scriptstyle \mathsf{self}\,p} & & \downarrow {\scriptstyle (p,P)} \\ \mathsf{self}\,\mathbb{C} & \xrightarrow[\mathsf{self}\,h]{} & \mathsf{self}\,\mathbb{D} & \xrightarrow[F^{\mathcal{D}}]{} & (\mathbb{D}, \mathcal{D}) \end{array}$$

For any $(c, \dot{d}) \in \mathsf{self}\,\dot{\mathbb{C}}$ we may therefore apply the universal property of the fibration $(p, P)$ to $\alpha_{q(c,\dot{d})} = \alpha_c$: For this, fix any object $(C, \dot{D}) \in \dot{\mathcal{C}}$ (recall Construction IV.9) and apply the universal property of the fibration to the arrow $\alpha_{q(c,\dot{d})} = \alpha_c$:

$$\begin{array}{ccc} \dot{\alpha}_c(F^{\dot{\mathcal{D}}}\dot{h}c) & \xrightarrow[1]{\dot{\alpha}_c} & (F^{\dot{\mathcal{D}}}\dot{h})c \\ & & \\ (HF^{\mathcal{C}})c & \xrightarrow[1]{\alpha_c} & (F^{\mathcal{D}}h)c = (PF^{\dot{\mathcal{D}}}\dot{h})c \end{array} \qquad \begin{array}{c} \dot{\mathcal{D}} \\ \downarrow {\scriptstyle (p,P)} \\ \mathcal{D} \end{array}$$

This definition extends to a locally indexed functor $K$ : $\mathsf{self}\,\dot{\mathbb{C}} \to (\dot{\mathbb{D}}, \mathcal{D})$, so we may use the universal property of the pullback in (5) to define $F^{\dot{\mathcal{C}}}(c, \dot{d})$ as the unique locally indexed functor filling the next diagram:




The right adjoint $U^{\dot{\mathcal{C}}}$ and 2-cell $\dot{\beta}$ are constructed similarly.

*Lifting via opfibrations.* As a consequence of our general theory (see Section VIII), Theorem V.1 has a dual, as follows.

**Corollary V.2.** *Let $(p, P)$ be a CBPV opfibration and $(h, H, \alpha, \beta)$ be a $\mathbf{CBPV}^{\mathrm{fo}}_{\mathrm{oplx}}$ 1-cell. Then the pullback* (4) *exists in $\mathbf{CBPV}^{\mathrm{fo}}_{\mathrm{oplx}}$ and $(q, Q)$ is a CBPV opfibration.*

Concretely the construction is similar to that outlined above, except $\dot{\alpha}$ and $\dot{\beta}$ are defined using *op*fibration structure.

**Remark V.3.** Corollary V.2 is useful because in general a monad morphism $\sigma : S \Rightarrow T$ (see e.g. [68]) induces an *oplax* adjunction morphism from the Eilenberg–Moore adjunction of $T$ to the Eilenberg–Moore adjunction of $S$ [47]. In such situations Corollary V.2 applies even though Theorem V.1 does not. For a concrete example, see Example VI.1.

### A. Examples

In this section we sketch some simple applications of our theorem. We leave a detailed exploration of the models for elsewhere: the aim is simply to show how our theorem yields a framework for building CBPV models, just as previous results do this for STLC and CBV models (cf. e.g. [10], [3], [21]).

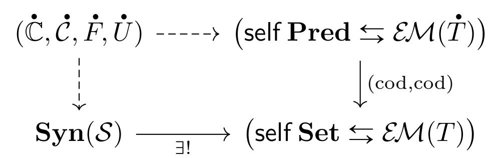
**Example V.4.** We start with the storage model as in Example III.8. There is a lax adjunction morphism from the storage model $(-) \times S \dashv S \Rightarrow (-)$ on $\mathbb{C}$ to the storage model $(-) \times \mathbb{C}(1, S) \dashv \mathbb{C}(1, S) \Rightarrow (-)$ on $\mathbf{Set}$ as follows. The functors $h$ and $H$ are both given by $\mathbb{C}(1, -)$. The 2-cell $\alpha$ is the isomorphism $\mathbb{C}(1, - \times S) \cong \mathbb{C}(1, -) \times \mathbb{C}(1, S)$, and $\beta_A : \mathbb{C}(1, S \Rightarrow A) \to \big(\mathbb{C}(1, S) \Rightarrow \mathbb{C}(1, A)\big)$ sends $t$ to $\lambda u \in \mathbb{C}(1, S) \,.\, \mathrm{eval} \circ \langle t, u\rangle$. The model in $\mathbf{Set}$ is easily lifted to $\mathbf{Pred}$: we take any subset $\overline{S} \subseteq \mathbb{C}(1, S)$ and consider the corresponding storage model on $\mathbf{Pred}$ (Example IV.22). Applying our construction, we get a CBPV model indexed by the category $\dot{\mathbb{C}}$ with objects pairs $\big(C \in \mathbb{C}, R \subseteq \mathbb{C}(1, X)\big)$. Since $\mathsf{self}$ commutes with pullbacks (Lemma IV.15), the locally indexed category must also be $\mathsf{self}\,\dot{\mathbb{C}}$. The lifted left and right adjoints $\dot{F}$ and $\dot{U}$ are defined by $\dot{F}(C, R) = (C \times S, R \times \overline{S})$ and $\dot{F}(C, R) = (S \Rightarrow C, \overline{S} \supset R)$

Next we use the universal property of the syntactic model (Example IV.17) to recover a definition of CBPV logical relations in the syntactic style. More precisely, from purely semantic reasoning we recover a version of the logical relations used by McDermott [41, p. 114].

**Example V.5.** Let $\mathcal{S}$ be a single-sorted signature. Then there is an associated free monad $T$ on $\mathbf{Set}$ which supports these operations, and the algebra model $F^T : \mathsf{self}\,\mathbf{Set} \leftrightarrows \mathcal{EM}(T) : U^T$ is a sound model of CBPV with basic operations from $\mathcal{S}$. Explicitly, $T$ sends a set $X$ to the set of terms generated using the basic operations with variables in $X$ (cf. [58, Remark 7.2]).

Now define an interpretation of base types and operations in $(\mathbf{Set}, \mathbf{Set}^T, F^T, U^T)$ by setting the interpretation of a value type $A$ to be the set of closed value terms of type $A$, and the interpretation of a computation type $\overline{A}$ to be the set of closed computations of type $\overline{A}$. By the free property of $\mathbf{Syn}_{\mathcal{S}}$, this extends to a strict map $\mathbf{Syn}_{\mathcal{S}} \to \big(\mathbf{Set}, \mathcal{EM}(T), F^T, U^T\big)$.

Finally, let $\dot{T}$ be a lifting of $T$ to $\mathbf{Pred}$; for definiteness, we choose the *free lifting* [40], [21]. Now apply Lemma IV.18 and Theorem V.1 to obtain a model $(\dot{\mathbb{C}}, \dot{\mathcal{C}}, \dot{F}, \dot{U})$ as shown:




Objects in $\dot{\mathbb{C}}$ consist of a value type $A$ and a set $V_A$ of closed value terms of type $A$. Objects in $\dot{\mathcal{C}}$ consist of a computation type $\overline{A}$ and a set of $C_{\overline{A}}$ of closed terms of type $\overline{A}$, equipped with $\dot{T}$-algebra structure. The action of the adjoints $\dot{F}$ and $\dot{U}$ in the lifted model are as follows:

$$\dot{F}(A, V_A) = \big(FA, F^{\dot{T}} V_A\big) \qquad \dot{U}(\overline{B}, C_{\overline{A}}) = \big(U\overline{B}, U^{\dot{T}} C_{\overline{A}}\big)$$

Since $\dot{T}$ is the free lifting, $\dot{F}(A, V_A)$ consists of the type $FA$ and the smallest relation containing $V_A$ that is closed under $\mathtt{return}$ and the operations in $\mathcal{S}$. On the other hand, $\dot{U}(\overline{B}, C_{\overline{A}})$ consists of the type $U\overline{B}$ and the set $C_{\overline{A}}$ with its algebra structure forgotten; this reflects the fact that $\mathtt{force}$ is invisible (recall Remark III.5). The action on products, sums, and function types is exactly as given by McDermott.

Our final example is a version of Katsumata's $\top\top$-lifting [20], adapted for CBPV models. Katsumata's construction relies on the fact that for any strong monad $T$ and any $T$-algebra there is a canonical strong monad morphism into the corresponding continuation monad. Because monad morphisms induce adjunction morphisms contravariantly (Remark V.3), this approach is not immediately available for adjunction models. Our strategy, therefore, is to first pass from our starting CBPV model to its corresponding algebra model, and then ask for a lifting of that model via Lemma IV.18.

In the next example we focus on $\top\top$-lifting but the construction is parametric in this choice: the argument works verbatim for any other lifting (e.g. the free lifting [40], [21], codensity lifting [22] or the monadic lifting of [18], [19]).

**Construction V.6** ($\top\top$-lifting for CBPV)**.** Let $(\mathbb{C}, \mathcal{C}, F, U)$ be a CBPV model in which $\mathbb{C}$ is also cartesian closed. Write $T$ for the induced (strong) monad $UF$ on $\mathbb{C}$. By [2, §11.6.2] there is a strict map into the algebra model $\big(\mathbb{C}, \mathcal{EM}(T), F^T, U^T\big)$ for $T$. Now fix an STLC fibration $p : \mathbb{E} \to \mathbb{C}$ and an object

$R \in \mathbb{E}$ as a *lifting parameter*. Finally, let $\dot{T}$ be the $\top\top$-lifting of $T$ with this parameter. By Lemma IV.18, we obtain a CBPV fibration $\big(\mathbb{E}, \mathcal{EM}(\dot{T}), F^{\dot{T}}, U^{\dot{T}}\big) \to \big(\mathbb{C}, \mathcal{EM}(T), F^T, U^T\big)$ and hence, by Theorem V.1, a lifted model as shown below:

$$\begin{array}{ccc} (\dot{\mathbb{C}}, \dot{\mathcal{C}}, \dot{F}, \dot{U}) & \dashrightarrow & \big(\mathbb{E}, \mathcal{EM}(\dot{T}), F^{\dot{T}}, U^{\dot{T}}\big) \\ \big\downarrow & & \big\downarrow{\scriptstyle (p,\tilde{p})} \\ (\mathbb{C}, \mathcal{C}, F, U) & \longrightarrow & \big(\mathbb{C}, \mathcal{EM}(T), F^T, U^T\big) \end{array} \qquad (6)$$

We call this the $\top\top$-*lifting* of the starting model.

**Example V.7.** We construct the $\top\top$-lifting of Levy's model of erratic choice [2, §5.5]. Thus, in our starting model the category of values is $\mathbf{Set}$ and the adjunction is the Kleisli resolution $J : \mathbf{Set} \leftrightarrows \mathbf{Rel} : K$ of the powerset monad $\mathcal{P}$. To lift $\mathcal{P}$ to $\mathbf{Pred}$, we take as our lifting parameter the $\mathcal{P}$-algebra $\big((\{0,1\},\{1\}), \mathbb{T}\big)$ where $(\{0,1\},\{1\} \subseteq \{0,1\})$ is an object of $\mathbf{Pred}$ and the arrow $\mathbb{T} : \mathcal{P}(\{0,1\}) \to \{0,1\}$ maps $p \subseteq \{0,1\}$ to 1 if $1 \in p$ and 0 otherwise. Applying $\top\top$-lifting, we obtain a strong monad $\dot{\mathcal{P}}$ on $\mathbf{Pred}$. This acts as $\dot{\mathcal{P}}(A,R) := (\mathcal{P}A, \dot{\mathcal{P}}R)$ where $p \in \dot{\mathcal{P}}R$ if and only if for all $f : X \to 2$ satisfying $\forall x \in R\,.\, f(x) = 1$ we have $\sum_{x \in X} f(x)\,p(x) = 1$. A direct calculation then shows that $p \in \dot{\mathcal{P}}R$ if and only if every $x \in p$ is in $R$. Applying our $\top\top$-lifting construction (6), the resulting model is the Kleisli adjunction $\mathbf{Pred} \leftrightarrows \mathbf{Pred}_{\dot{\mathcal{P}}}$ for $\dot{\mathcal{P}}$.

## VI. Effect Simulation

The *effect simulation problem* [15] is about relating different interpretations of the same computational effect. For example, one can give semantics to non-deterministic computation using either the finite powerset monad or the list monad. The effect simulation problem asks if these semantics are "the same", which one could state formally as asking if the sets of possible elements denoted by the list and powerset semantics are the same. Katsumata has studied this problem in detail for Moggi's computational λ-calculus [15]. As we shall see, the theory we have developed thus far means we can readily extend Katsumata's approach from CBV to CBPV.

The key idea, which has deep roots in the history of logical relations (e.g. [30, §2.2]), is that effect simulation is about constructing a non-standard model over the product of the models we are trying to relate. Products of CBPV models are given componentwise: for models $\big\{(\mathbb{C}_i, \mathcal{C}_i, F_i, U_i)\big\}_{i=1,\ldots,n}$ we get a product model $\big(\prod_{i=1}^n \mathbb{C}_i, \prod_{i=1}^n \mathcal{C}_i, \prod_{i=1}^n F_i, \prod_{i=1}^n U_i\big)$. This is because $\mathbf{LInd}$ has products and the 2-functor $\prod_{i=1}^n(-)$ lifts to a 2-functor on $\mathrm{Adj}(\mathbf{LInd})_{\mathrm{lx}}$; since self also preserves products, this restricts to a product on $\mathbf{CBPV}^{\mathrm{fo}}_{\mathrm{lx}}$.

Semantic effect simulation now arises from Theorem V.1 as follows. We start with two CBPV models, which for brevity we denote $\underline{\mathcal{C}}_i := (\mathbb{C}_i, \mathcal{C}_i, F_i, U_i)$ for $i = 1, 2$, a CBPV fibration $(p, P)$, and a (lax or oplax) CBPV model 1-cell as shown:

$$\begin{array}{ccc} & & (\mathbb{E}, \mathcal{E}, F^{\mathcal{E}}, U^{\mathcal{E}}) \\ & & \big\downarrow{\scriptstyle (p,P)} \\ \underline{\mathcal{C}}_1 \times \underline{\mathcal{C}}_2 & \xrightarrow[(h,H,\alpha,\beta)\times\mathrm{id}]{} & \underline{\mathcal{C}}_1 \times \underline{\mathcal{C}}_2 \end{array}$$

The effect simulation model is then constructed by applying Theorem V.1 or Corollary V.2.

**Example VI.1.** We relate the algebra models (Example III.7) for the finite powerset monad $\mathcal{P}_{\mathrm{fin}}$ and list $L$ monad on $\mathbf{Set}$. Their categories of algebras are, respectively, the category $\mathbf{SLat}$ of sup-semilattices and $\mathbf{Mon}$ of monoids. There is a canonical strong monad morphism $\gamma : L \to \mathcal{P}_{\mathrm{fin}}$ sending a list to its set of elements. This gives rise to the oplax adjunction morphism below, which in turn extends to an oplax CBPV model morphism $(\mathrm{id}, K, \gamma, \mathrm{id}) : \mathcal{EM}(\mathcal{P}_{\mathrm{fin}}) \to \mathcal{EM}(L)$ between the two algebra models.

$$\begin{array}{ccccc} \mathbf{Set} & \xrightarrow{\mathcal{P}_{\mathrm{fin}}} & \mathbf{SLat} & \xrightarrow{U} & \mathbf{Set} \\ {\scriptstyle \mathrm{id}}\big\downarrow & {\scriptstyle \gamma}\nearrow & \big\downarrow{\scriptstyle K} & & \big\downarrow{\scriptstyle \mathrm{id}} \\ \mathbf{Set} & \xrightarrow[L]{} & \mathbf{Mon} & \xrightarrow[U]{} & \mathbf{Set} \end{array}$$

Next consider the product model $\mathcal{EM}(L)^2$. We define a CBPV fibration into this model. By [15, Proposition 6], the category $\mathbf{BPred}$ of Example I.4 is a CCC with all coproducts. It follows that the free monoid monad $\dot{L}$ sends $(X, Y, R) \in \mathbf{BPred}$ to $\sum_{n \in \omega}(X, Y, R)^n$; since the fibration $q : \mathbf{BPred} \to \mathbf{Set}^2$ strictly preserves products and coproducts, this is a lifting of $L \times L$. Hence, by Lemma IV.18, we get a CBPV fibration $(q, \tilde{q}) : \big(\mathbf{BPred}, \mathcal{EM}(\dot{L})\big) \to (\mathbf{Set}, \mathcal{EM}(L))^2$.

Now we build our model for effect simulation. Applying Corollary V.2, we pullback $(q, \tilde{q})$ along $(\mathrm{id}, K, \gamma, \mathrm{id}) \times \mathrm{id}$ to obtain a CBPV model $(\mathbf{BPred}, \mathcal{SLMLift}, \dot{F}, \dot{U})$. The hom-presheaves of $\mathcal{SLMLift}$ are constructed componentwise by pullback so, in particular, the category $\mathcal{SLMLift}_1$ of arrows over the terminal object arises as the pullback shown below. $\mathbf{BPredMon}$ has objects triples $(M, N, R)$ such that $M$ and $N$ are monoids and $R$ is a submonoid of $M \times N$, and maps given by pairs of monoid morphisms that preserve the relation.

$$\begin{array}{ccc} \mathcal{SLMLift}_1 & \longrightarrow & \mathbf{BPredMon} \\ \big\downarrow & \lrcorner & \big\downarrow \\ \mathbf{SLat} \times \mathbf{Mon} & \longrightarrow & \mathbf{Mon} \times \mathbf{Mon} \end{array}$$

$\dot{F}$ acts on objects as $\dot{F}\big(X, Y, R\big) = \big(\mathcal{P}_{\mathrm{fin}}X, LY, \dot{R}\big)$, where for a finite set $p \subseteq X$ and list $l \in LY$, $(p, l) \in \dot{R}$ if and only if (1) for every $x \in p$ there is an element $y$ in $l$ such that $(x, y) \in R$, and (2) for every element $y$ in $l$ there's an element $x \in p$ such that $(x, y) \in R$. In this new model base types $\beta$ are interpreted as the diagonal relation $(\llbracket\beta\rrbracket \times \llbracket\beta\rrbracket, =)$ over an object $\llbracket\beta\rrbracket$ and the semantics of closed programs of type $F\beta$ are of the shape $(\gamma(l), l)$ for some list $l$.

## VII. Relative full completeness

In this section we show how our 2-categorical perspective leads relatively easily to a proof of *relative full completeness*, which establishes semantically that—absent sum types—function types are a conservative extension of the first-order fragment. Our proof follows the classic Lafont argument [34]: this argument is well-known, and has been applied in many differing situations (e.g. [17], [69], [70], [71]). Thus, our contribution here is not the proof strategy, but showing how to construct the ingredients to feed into the proof. Indeed, as several authors have noted [69], [70], the proof relies on:

1) A suitable "presheaf" model and a "nerve" construction;
2) The existence of certain comma objects ("glueing").

In what follows we shall outline how each of these ingredients arises for CBPV$^-$, the fragment of CBPV without sum types. The rest of the argument follows the classical pattern, as in e.g. [17, §4.10] so, for reasons of space, we omit it.

**Remark VII.1.** We omit sum types here because of (1). We want to have a map of models given by the Yoneda embedding, but the Yoneda functor does not generally preserve coproducts. There is a natural fix, namely to restrict to product-preserving presheaves (see e.g. [72]), but this introduces extra subtleties. Since this section is already both technical and rather compressed, we leave this for elsewhere. We write $\mathbf{CBPV}^{-,\mathrm{fo}}_{\mathrm{lx}}$ for the 2-category of CBPV$^-$ models and their morphisms, defined analogously to $\mathbf{CBPV}^{\mathrm{fo}}_{\mathrm{lx}}$.

As well as being of interest in its own right, we view the theory sketched here as a first step towards a semantic account of Kripke relations of varying arity for full CBPV, and thereby a characterisation of definability (cf. [12], [11], [23]) and normalisation-by-evaluation in the style of [35]. This would provide a completely-denotational counterpart to [73].

### A. Presheaf locally indexed categories

We construct our presheaf models for CBPV$^-$ using the corresponding structure in $\mathbf{LInd}$. The idea is to combine the enriched presheaf construction available on each 2-category $\mathbb{C}$-$\mathbf{LInd}$ (see e.g. [60, §2.2 & §4.4]) with the presheaf construction on $\mathbf{Cat}$.

First, as a 2-category of categories enriched in a presheaf category, each $\mathbb{C}$-$\mathbf{LInd}$ has a $\widehat{\mathbb{C}}$-enriched presheaf construction: for every $\mathcal{C} \in \mathbb{C}$-$\mathbf{LInd}$ there is a $\widehat{\mathbb{C}}$-category $\widehat{\mathcal{C}}$ of $\widehat{\mathbb{C}}$-functors $\mathcal{C}^{\mathrm{op}} \to \widehat{\mathbb{C}}$. This extends to a pseudofunctor $\mathrm{P} : \mathbb{C}\text{-}\mathbf{LInd} \to \mathbb{C}\text{-}\mathbf{LIND}$. The action on 1-cells is by left Kan extension, which determines the action on 2-cells. Applying P to $\mathcal{C} \in \mathbb{C}$-$\mathbf{LInd}$ yields a presheaf-like locally indexed category, but over the wrong base: it is still $\mathbb{C}$-indexed. We therefore apply change-of-base and define $\mathcal{P}$ as the composite

$$\mathbb{C}\text{-}\mathbf{LInd} \xrightarrow{\mathrm{P}} \mathbb{C}\text{-}\mathbf{LIND} \xrightarrow{y} \widehat{\mathbb{C}}\text{-}\mathbf{LIND}$$

Using standard enriched category-theoretic techniques, together with Levy's explicit identification of $\mathrm{P}(\mathcal{C})$ [1, p. 184], we arrive at the following characterisation of this composite.

Recall from e.g. [58, p. 84] that, for a locally $\mathbb{C}$-indexed category $\mathcal{C}$, the category $\mathsf{opGr}\,\mathcal{C}$ has objects $(c \in \mathbb{C}, C \in \mathcal{C})$ and morphisms $(c, C) \to (d, D)$ pairs of a map $\rho : d \to c$ in $\mathbb{C}$ and $f : C \underset{d}{\to} D$ in $\mathcal{C}$.

**Definition VII.2.** The *presheaf locally indexed category* $\mathcal{P}(\mathbb{C}, \mathcal{C}) := (\widehat{\mathbb{C}}, \widehat{\mathcal{C}})$ is defined as follows. The objects of $\widehat{\mathcal{C}}$ are functors $H : \mathsf{opGr}\,\mathcal{C}^{\mathrm{op}} \to \mathbf{Set}$ and maps $\tau : H \underset{P}{\to} H'$ are families of maps $\tau_{c,C} : P(c) \times H(c, C) \to H'(c, C)$ natural in each argument. Composition, identities, and reindexing are as in $\mathsf{self}\,[\mathsf{opGr}\,\mathcal{C}^{\mathrm{op}}, \mathbf{Set}]$.

A short end calculation shows that $(\widehat{\mathbb{C}}, \widehat{\mathcal{C}})$ is equivalently the locally $\widehat{\mathbb{C}}$-indexed category obtained by reindexing $\mathsf{self}\,[\mathsf{opGr}\,\mathcal{C}^{\mathrm{op}}, \mathbf{Set}]$ along the cartesian functor $\pi \circ (-) : \widehat{\mathbb{C}} \to [\mathsf{opGr}\,\mathcal{C}^{\mathrm{op}}, \mathbf{Set}]$ induced by the first projection $\pi : \mathsf{opGr}\,\mathcal{C} \to \mathbb{C}$. Moreover, if $\mathbb{C}$ and $\mathbb{D}$ are cartesian closed categories, and $f : \mathbb{C} \to \mathbb{D}$ preserves products, then $f^*(\mathsf{self}\,\mathbb{D}) \in \mathbb{C}$-$\mathbf{LInd}$ has products and $\mathbb{C}$-powers. Hence $\widehat{\mathcal{C}}$ has products and $\widehat{\mathbb{C}}$-powers.

### B. Presheaf CBPV models

Since pseudofunctors preserve adjunctions, $\mathcal{P}$ sends a CBPV model $F : \mathsf{self}\,\mathbb{C} \leftrightarrows \mathcal{C} : U$ to a locally $\widehat{\mathbb{C}}$-indexed adjunction $\mathcal{P}(\mathsf{self}\,\mathbb{C}) \leftrightarrows \mathcal{P}\mathcal{C}$ in which the adjoints $F_!$ and $U_!$ are computed using the left Kan extension in $\widehat{\mathbb{C}}$-$\mathbf{Cat}$. To make this into a CBPV model, observe there exists an adjunction

$$[\mathbb{C}^{\mathrm{op}}, \mathbf{Set}] \leftrightarrows [\mathsf{opGr}\,(\mathsf{self}\,\mathbb{C})^{\mathrm{op}}, \mathbf{Set}]$$

in which the left adjoint acts by $P \mapsto P(- \times =)$ and the right adjoint acts by $H \mapsto H(-, 1)$. Using the explicit characterisation above, one sees this extends to a locally $\widehat{\mathbb{C}}$-indexed adjunction $L : \mathsf{self}\,\widehat{\mathbb{C}} \leftrightarrows \mathcal{P}(\mathsf{self}\,\mathbb{C}) : R$. The *presheaf CBPV model* is then defined to be the composite adjunction

$$F_! \circ L : \mathsf{self}\,\widehat{\mathbb{C}} \leftrightarrows \mathcal{P}(\mathsf{self}\,\mathbb{C}) \leftrightarrows \mathcal{P}\mathcal{C} : R \circ U_!$$

We also obtain a Yoneda map. The first component is $y : \mathbb{C} \to \widehat{\mathbb{C}}$. For the second component we need a locally $\mathbb{C}$-indexed-functor $Y : \mathcal{C} \to y^*(\mathcal{P}\mathcal{C})$. Another end calculation shows that $y^*(\mathcal{P}\mathcal{C})$ is isomorphic to $[\mathcal{C}^{\mathrm{op}}, \widehat{\mathbb{C}}]$ in $\mathbb{C}$-$\mathbf{LIND}$, so we define $Y$ to be the $\widehat{\mathbb{C}}$-enriched Yoneda embedding: $Y(C) := \mathcal{C}_{-}(=, C)$. This extends to a pseudonatural transformation from the inclusion $\mathbb{C}$-$\mathbf{LInd} \hookrightarrow \mathbb{C}$-$\mathbf{LIND}$ to P (cf. [74, Lemma 3.7]) so there exists a pseudo adjunction map in the right square below; the left square is a strict adjunction map.

$$\begin{array}{ccccc}
\mathsf{self}\,[\mathbb{C}^{\mathrm{op}}, \mathbf{Set}] & \underset{\longleftarrow}{\overset{\perp}{\longrightarrow}} & \mathcal{P}(\mathsf{self}\,\mathbb{C}) & \underset{\longleftarrow}{\overset{\perp}{\longrightarrow}} & (\widehat{\mathbb{C}}, \widehat{\mathcal{C}}) \\
{\scriptstyle \mathsf{self}\,y}\uparrow & & {\scriptstyle (y,Y)}\uparrow & \cong & \uparrow{\scriptstyle \mathsf{self}\,y} \\
\mathsf{self}\,\mathbb{C} & = & \mathsf{self}\,\mathbb{C} & \underset{\longleftarrow}{\overset{\perp}{\longrightarrow}} & (\mathbb{C}, \mathcal{C})
\end{array}$$

Altogether, we have shown the next proposition.

**Definition VII.3.** A locally indexed functor $(f, F) : (\mathbb{C}, \mathcal{C}) \to (\mathbb{D}, \mathcal{D})$ is *full / faithful / fully faithful* if both $f$ and every functor $F_c : \mathcal{C}_c \to \mathcal{D}_{fc}$ are full / faithful / fully faithful. A $\mathbf{CBPV}^{-,\mathrm{fo}}_{\mathrm{lx}}$ 1-cell $(f, F, \alpha, \beta)$ is fully faithful if $(f, F)$ is.

**Proposition VII.4.** *For any CBPV model $\underline{\mathcal{C}}$ there is a fully faithful $\mathbf{CBPV}^{-,\mathrm{fo}}_{\mathrm{ps}}$ 1-cell $\underline{\mathcal{C}} \to \widehat{\underline{\mathcal{C}}}$ into the presheaf CBPV model. We denote this by $\underline{Y}$.*

Note that $\mathsf{self}\,f$ is fully faithful if $f$ is. The final observation about presheaves we need is the following.

**Proposition VII.5.** *For any $\mathbf{CBPV}^{-,\mathrm{fo}}_{\mathrm{lx}}$ morphism $\underline{F} : \underline{\mathcal{B}} \to \underline{\mathcal{C}}$ there exists a $\mathbf{CBPV}^{-,\mathrm{fo}}_{\mathrm{oplx}}$ 1-cell $\underline{\langle F \rangle} : \underline{\mathcal{C}} \to \widehat{\underline{\mathcal{B}}}$ and a $\mathbf{CBPV}^{-,\mathrm{fo}}_{\mathrm{oplx}}$ 2-cell $\Gamma : \underline{Y} \Rightarrow \underline{\langle F \rangle} \circ \underline{F}$.*

Indeed, for any locally indexed functor $(f, F) : (\mathbb{B}, \mathcal{B}) \to (\mathbb{C}, \mathcal{C})$ we obtain $\langle f \rangle : \mathbb{C} \to \widehat{\mathbb{B}}$ and $\langle F \rangle : \mathcal{C} \to \langle f \rangle^*(\widehat{\mathcal{B}})$ by taking $\langle f \rangle c := \mathbb{C}(f-, c)$ and $\langle F \rangle(C) := \mathcal{C}_{f-}(=, C)$. Note that $\langle f \rangle$ preserves products because $f$ does. The rest of the calculation essentially follows by unwinding the standard fact—which holds equally in the enriched setting—that $\langle f \rangle$ is the left Kan extension of $y$ along $f$.

### C. Completing the proof

Let $\mathbf{Syn}_\mathcal{S}$ be the syntactic model over a signature $\mathcal{S}$ (recall Example III.6). Also let $\mathbf{FOSyn}_\mathcal{S}$ be the first-order syntactic model, with function types omitted. Both these models are free (Example IV.17) so there is a canonical strict map of first-order CBPV$^-$ models $(i, I) : \mathbf{FOSyn}_\mathcal{S} \to \mathbf{Syn}_\mathcal{S}$. We prove the following.

**Theorem VII.6** (Relative full completeness)**.** *For any signature $\mathcal{S}$, the locally indexed functor $(i, I)$ is fully faithful.*

The remaining difficulty lies in seeing that for any $\mathbf{CBPV}^{-,\mathrm{fo}}_{\mathrm{oplx}}$ morphism $(g, G, \alpha, \beta)$ the following comma object exists, i.e. CBPV models admit *glueing*:

$$\begin{array}{ccc} \underline{\mathcal{G}} & \longrightarrow & \underline{\mathcal{C}} \\ \downarrow & \underset{\lambda}{\Longleftarrow} & \| \\ \underline{\mathcal{B}} & \underset{(g,G,\alpha,\beta)}{\longrightarrow} & \underline{\mathcal{C}} \end{array}$$

This follows from two facts. First, for any 2-category D, if $\mathcal{C}$ has comma objects then $[\mathrm{D}, \mathcal{C}]_{\mathrm{oplx}}$ also has such comma objects, computed component-wise (cf. [75, Proposition 4.6]). Since $\mathbf{LInd}$ has all comma objects, so does $\mathrm{Adj}(\mathbf{LInd})_{\mathrm{oplx}}$. Second, a small adaptation of the classical proof (e.g. [17]) shows this restricts to $\mathbf{CBPV}^{\times}_{\mathrm{oplx}}$: when $\underline{\mathcal{C}}$ is a CBPV model with locally indexed pullbacks, $\underline{\mathcal{G}}$ is also a CBPV model.

The rest of the argument is as in the classical case (see e.g. [17, §4.10] or [70, §3.2]), observing that composition in $\mathbf{CBPV}^{-,\mathrm{fo}}_{\mathrm{oplx}}$ reduces to (1) composition of the functors in $\mathbf{DistCat}$ on the first component, and (2) on the second component, composition in $\mathbf{Cat}$ at each index.

## VIII. Lifting theorems for arbitrary shapes

In this final technical section we outline how Theorem V.1 and Corollary V.2 are special cases of a general result which applies to any "shape" of model, including those for $\lambda_{\mathrm{ml}}$ and CBPV. The key idea is that, since $\mathbf{CBPV}^{\mathrm{fo}}_{\mathrm{lx}}$ is a sub-2-category of the functor 2-category $\mathrm{Adj}(\mathbf{LInd})_{\mathrm{lx}} = [\mathbf{Adj}, \mathbf{LInd}]_{\mathrm{lx}}$, Theorem V.1 is a special case of a result about pullbacks in 2-categories of the form $[\mathrm{D}, \mathcal{C}]_{\mathrm{lx}}$ for some 2-category D of "diagram shapes". We state the general version in Theorem VIII.1.

For the theorem we need to isolate a class of 1-cells which will play the role of fibrations for logical relations, i.e. fibrations which strictly preserve structure. We do this in a style inspired by [76]. Fix a 2-category $\mathcal{C}$ and a wide, locally-full sub-2-category $\mathcal{C}_t$; we call the 1-cells in $\mathcal{C}_t$ *tight*. Further assume every fibration is tight. The statement is then as follows.

**Theorem VIII.1.** *In the situation just sketched, consider a cospan $(G \xrightarrow{\phi} H \xleftarrow{\kappa} \dot{H})$ in $[\mathrm{D}, \mathcal{C}]_{\mathrm{lx}}$ such that*

1) *$\kappa$ is strict and each 1-cell component $\kappa_d$ is tight;*
2) *For each $d \in \mathrm{D}$ the pullback $(\phi_d)^*(\kappa_d)$ of $\kappa_d$ along $\phi_d$ exists in $\mathcal{C}$ and is tight.*

*Then the 1-cells $(\phi_d)^*(\kappa_d)$ form a strict, componentwise-tight transformation, which is the pullback $\phi^*(\kappa)$ in $[\mathrm{D}, \mathcal{C}]_{\mathrm{lx}}$.*

Essentially, this says that to construct pullbacks of models with additional structure encoded by a 2-functor it suffices to construct pullbacks of the underlying 'base' model. The proof is similar to that sketched in Section V.

**Example VIII.2.** We recover Theorem V.1 as follows. Take $\mathrm{D} := \mathbf{Adj}$ and $\mathcal{C}$ to be the sub-2-category of $\mathbf{LInd}$ with objects $(\mathbb{C}, \mathcal{C})$ given by a distributive category $\mathbb{C}$ and a locally $\mathbb{C}$-indexed category $\mathcal{C}$ with finite products and $\mathbb{C}$-powers, such that $\mathbb{C}$'s coproducts are distributive. The 1-cells are $\mathbf{LInd}$-maps which preserve the products and coproducts. Say a 1-cell $(p, P)$ is tight if $p$ is a bifibration—so pullbacks of $p$ exist in $\mathbf{DistCat}$—and $(p, P)$ is a locally indexed fibration which strictly preserves the products, coproducts, and powers. $\mathcal{C}$ satisfies the conditions of Theorem VIII.1, with the required pullbacks computed as in $\mathbf{LInd}$. Since $\mathsf{self}$ also preserves pullbacks (Lemma IV.15), the pullback of a CBPV model is still a CBPV model. So we obtain Theorem V.1. Corollary V.2 follows by instantiating the theorem with $\mathbf{LInd}^{\mathrm{co}}$.

We obtain a version for $\lambda_{\mathrm{ml}}$ by varying the "shapes". By [29, §3.6], applying the 2-Grothendieck construction to the 2-functor $\mathbf{CartCat}^{\mathrm{op}} \to 2\text{-}\mathbf{CAT}$ sending $\mathbb{V}$ to the 2-category $\mathbb{V}$-$\mathbf{Act}_{\mathrm{lx}}$ of $\mathbb{V}$-actions and lax maps yields a 2-category $\mathbf{Act}$ of actions. The objects are triples of a cartesian category $\mathbb{V}$, a category $\mathbb{C}$, and a left action of $\mathbb{V}$ on $\mathbb{C}$. 1-cells $(\mathbb{V}, \mathbb{C}, \bullet) \to (\mathbb{W}, \mathbb{D}, \star)$ are triples $(f, F, \phi)$ where $f : \mathbb{V} \to \mathbb{W}$ is cartesian, $F : \mathbb{C} \to \mathbb{D}$, and $\phi_{V,C} : f(V) \star F(C) \to F(V \bullet C)$ is a natural transformation compatible with the two actions. $\mathbf{Act}$ plays the role for $\lambda_{\mathrm{ml}}$ that $\mathbf{LInd}$ did for CBPV above. First, a monad internal to $\mathbf{Act}$ (see [68]) is exactly a left action together with a monad that is strong for the action (e.g. [25, §3]). Second, using [57, Theorem 2.7], one sees the fibrations in $\mathbf{Act}$ are 1-cells $(f, F, \phi)$ such that both $f$ and $F$ are fibrations. Finally, there is a 2-functor $\mathsf{self} : \mathbf{CartCat} \to \mathbf{Act}$ sending a cartesian category $\mathbb{V}$ to the canonical $\mathbb{V}$-action on itself. This picks out the underlying structure of a $\lambda_{\mathrm{ml}}$ model, and preserves both fibrations and pullbacks (cf. Lemma IV.15).

**Example VIII.3.** Take $\mathrm{D} := \mathbf{Mnd}$ to be the walking monad, i.e. the 2-category freely generated by the data of a monad (e.g. [48], [49]), and let $\mathcal{C}$ be the sub-2-category of $\mathbf{Act}$ with objects $(\mathbb{V}, \mathbb{C}, \bullet)$ such that $\mathbb{V}$ is cartesian closed. Say a 1-cell is tight if it is an $\mathbf{Act}$-fibration which strictly preserves both the cartesian closed structure and the action. Then consider a cospan $(\mathsf{self}\,\mathbb{C} \xrightarrow{\mathsf{self}\,F} \mathsf{self}\,\mathbb{B} \xleftarrow{\mathsf{self}\,p} \mathsf{self}\,\mathbb{E})$ where $p$ is a fibration which strictly preserves the cartesian closed structure. Since $p$ is tight and the pullback exists in $\mathbf{Act}$, by Theorem VIII.1 this becomes a pullback of $\lambda_{\mathrm{ml}}$ models, which yields Lemma I.5.

## Acknowledgments

We are grateful to the Programming Languages Group in Oxford for many constructive conversations, and to Nathanael Arkor for important technical discussions. We also thank the LICS 2025 reviewers for their useful suggestions.

## References

[1] P. B. Levy, "Call-by-push-value," Ph.D. dissertation, Queen Mary and Westfield College, University of London, 2001. [Online]. Available: https://qmro.qmul.ac.uk/xmlui/handle/123456789/4742

[2] ——, *Call-By-Push-Value*. Springer Netherlands, 2003. [Online]. Available: https://doi.org/10.1007/978-94-007-0954-6

[3] C. A. Hermida, "Fibrations, logical predicates and indeterminates," Ph.D. dissertation, University of Edinburgh, 1993. [Online]. Available: http://hdl.handle.net/1842/14057

[4] G. D. Plotkin, "LCF considered as a programming language," *Theoretical Computer Science*, vol. 5, no. 3, pp. 223–255, Dec. 1977. [Online]. Available: https://doi.org/10.1016/0304-3975(77)90044-5

[5] K. Sieber, "Reasoning about sequential functions via logical relations," in *Applications of Categories in Computer Science*. Cambridge University Press, Jun. 1992, pp. 258–269. [Online]. Available: https://doi.org/10.1017/cbo9780511525902.015

[6] A. W. Appel and D. McAllester, "An indexed model of recursive types for foundational proof-carrying code," *ACM Trans. Program. Lang. Syst.*, vol. 23, no. 5, p. 657–683, Sep. 2001. [Online]. Available: https://doi.org/10.1145/504709.504712

[7] A. M. Pitts, "Step-Indexed Biorthogonality: a Tutorial Example," in *Modelling, Controlling and Reasoning About State*, ser. Dagstuhl Seminar Proceedings (DagSemProc), A. Ahmed, N. Benton, L. Birkedal, and M. Hofmann, Eds., vol. 10351. Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2010, pp. 1–10. [Online]. Available: https://drops.dagstuhl.de/entities/document/10.4230/DagSemProc.10351.6

[8] D. Dreyer, A. Ahmed, and L. Birkedal, "Logical step-indexed logical relations," *Logical Methods in Computer Science*, vol. Volume 7, Issue 2, Jun. 2011. [Online]. Available: https://doi.org/10.2168/lmcs-7(2:16)2011

[9] Q. M. Ma and J. C. Reynolds, "Types, abstraction, and parametric polymorphism, part 2," in *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 1992, pp. 1–40. [Online]. Available: https://doi.org/10.1007/3-540-55511-0_1

[10] J. C. Mitchell and A. Scedrov, "Notes on sconing and relators," in *Computer Science Logic*. Springer Berlin Heidelberg, 1993, pp. 352–378. [Online]. Available: https://doi.org/10.1007/3-540-56992-8_21

[11] A. Jung and J. Tiuryn, "A new characterization of lambda definability," in *Typed Lambda Calculi and Applications*, M. Bezem and J. F. Groote, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1993, pp. 245–257. [Online]. Available: https://doi.org/10.1007/BFb0037110

[12] M. Alimohamed, "A characterization of lambda definability in categorical models of implicit polymorphism," *Theor. Comput. Sci.*, vol. 146, no. 1-2, pp. 5–23, Jul. 1995. [Online]. Available: http://dx.doi.org/10.1016/0304-3975(94)00283-O

[13] B. Jacobs, *Categorical Logic and Type Theory*, ser. Studies in Logic and the Foundations of Mathematics. Amsterdam: North Holland, 1999, no. 141.

[14] S. Katsumata, "A characterisation of lambda definability with sums via $\top\top$-closure operators," in *Computer Science Logic*, M. Kaminski and S. Martini, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 278–292. [Online]. Available: https://doi.org/10.1007/978-3-540-87531-4_21

[15] ——, "Relating computational effects by $\top\top$-lifting," *Information and Computation*, vol. 222, pp. 228 – 246, 2013, 38th International Colloquium on Automata, Languages and Programming (ICALP 2011). [Online]. Available: https://doi.org/10.1016/j.ic.2012.10.014

[16] E. Moggi, "Notions of computation and monads," *Inf. Comput.*, vol. 93, no. 1, pp. 55–92, 1991. [Online]. Available: https://doi.org/10.1016/0890-5401(91)90052-4

[17] R. L. Crole, *Categories for Types*. Cambridge University Press, Jan. 1994. [Online]. Available: https://doi.org/10.1017/CBO9781139172707

[18] J. Goubault-Larrecq, S. Lasota, and D. Nowak, "Logical relations for monadic types," in *Computer Science Logic*. Springer Berlin Heidelberg, 2002, pp. 553–568. [Online]. Available: https://doi.org/10.1007/3-540-45793-3_37

[19] ——, "Logical relations for monadic types," *Mathematical Structures in Computer Science*, vol. 18, no. 06, p. 1169, Oct. 2008. [Online]. Available: https://doi.org/10.1017/S0960129508007172

[20] S. Katsumata, "A semantic formulation of $\top\top$-lifting and logical predicates for computational metalanguage," in *Computer Science Logic*. Springer Berlin Heidelberg, 2005, pp. 87–102. [Online]. Available: https://doi.org/10.1007/11538363_8

[21] O. Kammar and D. McDermott, "Factorisation systems for logical relations and monadic lifting in type-and-effect system semantics," *Electronic Notes in Theoretical Computer Science*, vol. 341, pp. 239 – 260, 2018, proceedings of the Thirty-Fourth Conference on the Mathematical Foundations of Programming Semantics (MFPS XXXIV). [Online]. Available: https://doi.org/10.1016/j.entcs.2018.11.012

[22] S.-y. Katsumata, T. Sato, and T. Uustalu, "Codensity lifting of monads and its dual," *Logical Methods in Computer Science*, vol. 14, no. 4, 2018. [Online]. Available: https://doi.org/10.23638/LMCS-14(4:6)2018

[23] O. Kammar, S. Katsumata, and P. Saville, "Fully abstract models for effectful $\lambda$-calculi via category-theoretic logical relations," *Proceedings of the ACM on Programming Languages*, vol. 6, no. POPL, pp. 1–28, Jan. 2022. [Online]. Available: https://doi.org/10.1145/3498705

[24] A. Kock, "Strong functors and monoidal monads," *Archiv der Mathematik*, vol. 23, no. 1, pp. 113–120, dec 1972. [Online]. Available: https://doi.org/10.1007/BF01304852

[25] D. McDermott and T. Uustalu, "What makes a strong monad?" *Electronic Proceedings in Theoretical Computer Science*, vol. 360, pp. 113–133, Jun. 2022. [Online]. Available: https://doi.org/10.4204/EPTCS.360.6

[26] P. J. Freyd and A. Ščedrov, *Categories, allegories*, transferred to digital print. ed., ser. North Holland mathematical library. Amsterdam [u.a.]: North-Holland, 2006, no. 39.

[27] G. Wraith, "Artin glueing," *Journal of Pure and Applied Algebra*, vol. 4, no. 3, pp. 345–348, Jun. 1974. [Online]. Available: https://doi.org/10.1016/0022-4049(74)90014-0

[28] A. Carboni and P. Johnstone, "Connected limits, familial representability and artin glueing," *Mathematical Structures in Computer Science*, vol. 5, no. 4, pp. 441–459, Dec. 1995, See also the corrigenda *Corrigenda for 'Connected limits, familial representability and Artin glueing'* at DOI 10.1017/S0960129503004080.

[29] M. Capucci and B. Gavranović, "Actegories for the working amthematician," 2022. [Online]. Available: https://doi.org/10.48550/ARXIV.2203.16351

[30] R. Milner, "The formal semantics of computer languages and their interpretations," University of Oxford, Tech. Rep., 1974, available online at cs.ox.ac.uk/files/3286/PRGX13.pdf.

[31] M. Shulman, "Framed bicategories and monoidal fibrations," *Theory and Applications of Categories*, vol. 20, no. 18, 2008. [Online]. Available: tac.mta.ca/tac/volumes/20/18/20-18.pdf

[32] C. Vasilakopoulou, "On enriched fibrations," *Cahiers de topologie et géométrie différentielle catégoriques*, vol. LIX, no. 4, 2018. [Online]. Available: https://cahierstgdc.com/wp-content/uploads/2018/10/Vasilakopoulou-LIX-4.pdf

[33] J. Moeller and C. Vasilakopoulou, "Monoidal Grothendieck construction," *Theory and Applications of Categories*, vol. 35, no. 31, 2020. [Online]. Available: www.tac.mta.ca/tac/volumes/35/31/35-31.pdf

[34] Y. Lafont, "Logiques, catégories et machines," Ph.D. dissertation, Université Paris VII, 1987.

[35] M. Fiore, "Semantic analysis of normalisation by evaluation for typed lambda calculus," in *Proceedings of the 4th ACM SIGPLAN International Conference on Principles and Practice of Declarative Programming*, ser. PPDP '02. New York, NY, USA: ACM, 2002, pp. 26–37. [Online]. Available: http://doi.acm.org/10.1145/571157.571161

[36] D. Turi and G. Plotkin, "Towards a mathematical operational semantics," in *Proceedings of Twelfth Annual IEEE Symposium on Logic in Computer Science*, 1997, pp. 280–291. [Online]. Available: https://doi.org/10.1109/LICS.1997.614955

[37] S. Goncharov, A. Santamaria, L. Schröder, S. Tsampas, and H. Urbat, *Logical Predicates in Higher-Order Mathematical Operational Semantics*. Springer Nature Switzerland, 2024, pp. 47–69. [Online]. Available: https://doi.org/10.1007/978-3-031-57231-9_3

[38] S. Goncharov, S. Milius, S. Tsampas, and H. Urbat, "Bialgebraic reasoning on higher-order program equivalence," in *Proceedings of the 39th Annual ACM/IEEE Symposium on Logic in Computer Science*, ser. LICS '24. New York, NY, USA: Association for Computing Machinery, 2024. [Online]. Available: https://doi.org/10.1145/3661814.3662099

[39] S. Goncharov, S. Tsampas, and H. Urbat, "Abstract operational methods for call-by-push-value," *Proceedings of the ACM on Programming Languages*, vol. 9, no. POPL, pp. 1013–1039, Jan. 2025.

[40] O. Kammar, "An algebraic theory of type-and-effect systems," Ph.D. dissertation, University of Edinburgh, 2014. [Online]. Available: http://hdl.handle.net/1842/8910

[41] D. McDermott, "Reasoning about effectful programs and evaluation order," Ph.D. dissertation, University of Cambridge, 2020. [Online]. Available: https://doi.org/10.48456/tr-948

[42] P. H. Azevedo de Amorim, "Denotational foundations for expected cost analysis," *Proc. ACM Program. Lang.*, vol. 9, no. OOPSLA1, Apr. 2025. [Online]. Available: https://doi.org/10.1145/3720424

[43] F. Borceux, *Handbook of Categorical Algebra, volume 2*. Cambridge University Press, Aug. 1994. [Online]. Available: https://doi.org/10.1017/CBO9780511525865.008

[44] R. B. B. Lucyshyn-Wright, "V-graded categories and V-W-bigraded categories: Functor categories and bifunctors over non-symmetric bases," 2025. [Online]. Available: https://doi.org/10.48550/ARXIV.2502.18557

[45] T. Leinster, *Higher Operads, Higher Categories*. Cambridge University Press, Jul. 2004, preprint available online at https://doi.org/10.48550/arXiv.math/0305049.

[46] N. Johnson and D. Yau, *2-Dimensional Categories*. Oxford University Press, 2021, preprint available online at 10.48550/arXiv.2002.06055.

[47] D. Pumplün, "Eine Bemerkung über Monaden und adjungierte Funktoren," *Mathematische Annalen*, vol. 185, pp. 329–337, 1970. [Online]. Available: http://eudml.org/doc/161964

[48] C. Auderset, "Adjonctions et monades au niveau des 2-catégories," *Cahiers de Topologie et Géométrie Différentielle Catégoriques*, vol. 15, no. 1, pp. 3–20, 1974. [Online]. Available: http://eudml.org/doc/91131

[49] S. Schanuel and R. Street, "The free adjunction," *Cahiers de Topologie et Géométrie Différentielle Catégoriques*, vol. 27, no. 1, pp. 81–83, 1986. [Online]. Available: https://www.numdam.org/item/CTGDC_1986__27_1_81_0/

[50] R. Street, "Conspectus of variable categories," *Journal of Pure and Applied Algebra*, vol. 21, no. 3, pp. 307–338, Jun. 1981. [Online]. Available: https://doi.org/10.1016/0022-4049(81)90021-9

[51] ——, *Fibrations and Yoneda's lemma in a 2-category*. Springer Berlin Heidelberg, 1974, pp. 104–133. [Online]. Available: https://doi.org/10.1007/bfb0063102

[52] F. Loregian and E. Riehl, "Categorical notions of fibration," *Expositiones Mathematicae*, vol. 38, no. 4, pp. 496–514, Dec. 2020. [Online]. Available: https://doi.org/10.1016/j.exmath.2019.02.004

[53] J. W. Gray, *Fibred and Cofibred Categories*. Springer Berlin Heidelberg, 1966, pp. 21–83. [Online]. Available: https://doi.org/10.1007/978-3-642-99902-4_2

[54] S. Lack, *A 2-Categories Companion*. Springer New York, Sep. 2009, pp. 105–191. [Online]. Available: https://doi.org/10.1007/978-1-4419-1524-5_4

[55] A. Power, "A general coherence result," *Journal of Pure and Applied Algebra*, vol. 57, no. 2, pp. 165–173, Mar. 1989. [Online]. Available: https://doi.org/10.1016/0022-4049(89)90113-8

[56] R. Blackwell, G. Kelly, and A. Power, "Two-dimensional monad theory," *Journal of Pure and Applied Algebra*, vol. 59, no. 1, pp. 1–41, Jul. 1989. [Online]. Available: https://doi.org/10.1016/0022-4049(89)90160-6

[57] M. Weber, "Yoneda structures from 2-toposes," *Applied Categorical Structures*, vol. 15, no. 3, pp. 259–323, May 2007. [Online]. Available: https://doi.org/10.1007/s10485-007-9079-2

[58] P. B. Levy, "Adjunction models for call-by-push-value with stacks," *Electronic Notes in Theoretical Computer Science*, vol. 69, pp. 248–271, 2003, cTCS'02, Category Theory and Computer Science. [Online]. Available: https://doi.org/10.1016/S1571-0661(04)80568-1

[59] ——, "Call-by-push-value: Decomposing call-by-value and call-by-name," *Higher-Order and Symbolic Computation*, vol. 19, no. 4, pp. 377–414, Dec. 2006. [Online]. Available: https://doi.org/10.1007/s10990-006-0480-6

[60] G. M. Kelly, *Basic Concepts of Enriched Category Theory*. Reprints in Theory and Applications of Categories, 2005. [Online]. Available: http://www.tac.mta.ca/tac/reprints/articles/10/tr10.pdf

[61] E. Moggi, "Computational lambda-calculus and monads," in *Proceedings of the Fourth Annual Symposium on Logic in Computer Science*. IEEE Press, 1989, p. 14–23. [Online]. Available: https://doi.org/10.1109/LICS.1989.39155

[62] S. Eilenberg and G. M. Kelly, *Closed Categories*. Springer Berlin Heidelberg, 1966, pp. 421–562. [Online]. Available: https://doi.org/10.1007/978-3-642-99902-4_22

[63] G. S. H. Cruttwell, "Normed spaces and the change of base for enriched categories," Ph.D. dissertation, Dalhousie University, 2008, available online at https://www.reluctantm.com/gcruttw/publications/thesis4.pdf.

[64] I. Baković, "Grothendieck construction for bicategories," 2010. [Online]. Available: https://www2.irb.hr/korisnici/ibakovic/sgc.pdf

[65] M. Buckley, "Fibred 2-categories and bicategories," *Journal of Pure and Applied Algebra*, vol. 218, no. 6, pp. 1034–1074, Jun. 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0022404913002065?via=ihub

[66] J. Hughes and B. Jacobs, "Factorization systems and fibrations," *Electronic Notes in Theoretical Computer Science*, vol. 69, pp. 156–182, Feb. 2003. [Online]. Available: https://doi.org/10.1016/S1571-0661(04)80564-4

[67] F. E. J. Linton, *Coequalizers in categories of algebras*. Springer Berlin Heidelberg, 1969, pp. 75–90. [Online]. Available: https://doi.org/10.1007/BFb0083082

[68] R. Street, "The formal theory of monads," *Journal of Pure and Applied Algebra*, vol. 2, no. 2, pp. 149 –168, 1972. [Online]. Available: https://doi.org/10.1016/0022-4049(72)90019-9

[69] M. Hasegawa, "Categorical glueing and logical predicates for models of linear logic," *Kyoto University. Research Institute for Mathematical Sciences [RIMS]*, 1999. [Online]. Available: https://www.kurims.kyoto-u.ac.jp/~hassei/papers/full.pdf

[70] M. Fiore, R. Di Cosmo, and V. Balat, "Remarks on isomorphisms in typed lambda calculi with empty and sum types," in *Proceedings 17th Annual IEEE Symposium on Logic in Computer Science*, ser. LICS-02. IEEE Comput. Soc, 2002, pp. 147–156. [Online]. Available: https://doi.org/10.1109/LICS.2002.1029824

[71] M. Fiore and P. Saville, *Relative Full Completeness for Bicategorical Cartesian Closed Structure*. Springer International Publishing, 2020, pp. 277–298. [Online]. Available: https://doi.org/10.1007/978-3-030-45231-5_15

[72] P. Fu, K. Kishida, N. J. Ross, and P. Selinger, "On the Lambek embedding and the category of product-preserving presheaves," *arXiv*, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2205.06068

[73] A. Abel and C. Sattler, "Normalization by evaluation for call-by-push-value and polarized lambda calculus," in *Proceedings of the 21st International Symposium on Principles and Practice of Declarative Programming*, ser. PPDP '19. New York, NY, USA: Association for Computing Machinery, 2019. [Online]. Available: https://doi.org/10.1145/3354166.3354168

[74] M. Fiore, N. Gambino, M. Hyland, and G. Winskel, "Relative pseudomonads, Kleisli bicategories, and substitution monoidal structures," *Selecta Mathematica*, vol. 24, no. 3, pp. 2791–2830, Nov. 2017. [Online]. Available: https://doi.org/10.1007/s00029-017-0361-3

[75] S. Lack, "Limits for lax morphisms," *Applied Categorical Structures*, vol. 13, no. 3, pp. 189–203, Jun. 2005. [Online]. Available: https://doi.org/10.1007/s10485-005-2958-5

[76] N. Arkor, J. Bourke, and J. Ko, "Enhanced 2-categorical structures, two-dimensional limit sketches and the symmetry of internalisation," *arXiv*, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2412.07475

[77] S. Mac Lane, *Categories for the Working Mathematician*, 2nd ed., ser. Graduate Texts in Mathematics. Springer-Verlag New York, 1998, vol. 5.