

# Categorical Models for Abadi-Plotkin's Logic for Parametricity

LARS BIRKEDAL<sup>1†</sup> and RASMUS E. MØGELBERG<sup>2</sup>

<sup>1</sup> *IT University of Copenhagen, Denmark,*

<sup>2</sup> *IT University of Copenhagen, Denmark.*

*Received 17 February 2005*

We propose a new category-theoretic formulation of relational parametricity based on a logic for reasoning about parametricity given by Abadi and Plotkin (Plotkin and Abadi, 1993). The logic can be used to reason about parametric models, such that we may prove consequences of parametricity that to our knowledge have not been proved before for existing category-theoretic notions of relational parametricity. We provide examples of parametric models and we describe a way of constructing parametric models from given models of the second-order lambda calculus.

## Contents

1	Introduction	3
2	Abadi & Plotkin's logic	5
2.1	Second-order $\lambda$ -calculus	5
2.2	The logic	7
2.3	Definable relations	7
2.4	The axioms	9
3	APL-structures	12
3.1	Soundness	17
3.2	Completeness	21
4	Parametric APL-structures	23
5	Consequences of parametricity	25
5.1	Dinaturality	25
5.2	Products	27
5.3	Coproducts	29
5.4	Initial algebras	31
5.5	Final coalgebras	35
5.6	Generalizing to strong fibred functors	38

<sup>†</sup> Corresponding Author. Full address: IT University of Copenhagen, Rued Langgaards Vej 7, DK-2300 Copenhagen S, Denmark. Email: [birkedal@itu.dk](mailto:birkedal@itu.dk). This work was partly supported by the Danish Technical Research Council under grant no.: 56-00-0309

<i>L. Birkedal and R.E. Møgelberg</i>	2
6 Concrete APL-structures	41
6.1 A parametric non-well-pointed APL-structure	43
7 Comparing with Ma & Reynolds notion of parametricity	44
8 A parametric completion process	45
8.1 Internal models for $\lambda_2$	46
8.2 Input for the parametric completion process	47
8.3 The completion process	48
8.4 The APL-structure	53
9 Parametric Internal Models	58
10 Conclusion	59
Appendix A Composable Fibrations	60
References	64

## 1. Introduction

The notion of parametricity for models of polymorphic type theories intuitively states that a function of polymorphic type behaves the same way on all type instances. Reynolds (Reynolds, 1983) discovered that parametricity is central for modeling data abstraction and proving representation independence results. The idea is that a client of an abstract data type is modeled as a polymorphic function; parametricity then guarantees that the client cannot distinguish between different implementations of the abstract data type. Reynolds also observed that parametricity can be used for encoding (inductive and coinductive) data types. See (Wadler, 1989; Mitchell, 1996) for expository introductions.

In 1983 Reynolds gave a precise formulation of parametricity called relational parametricity for set-theoretic models (Reynolds, 1983). It basically states that a term of polymorphic type preserves relations between types: if term  $u$  has type  $\prod \alpha : \text{Type}. \sigma$  and  $R : \text{Rel}(\tau, \tau')$  is a relation between  $\tau$  and  $\tau'$ , then

$$u(\tau)(\sigma[R])u(\tau'),$$

where  $\sigma[R]$  is a relational interpretation of the type  $\sigma$  defined inductively over the structure of  $\sigma$ . Equivalently, parametricity could be defined as the identity extension property: for all terms  $u, v$  of type  $\sigma(\vec{\alpha})$ ,

$$u(\sigma[e\vec{q}_\alpha])v \iff u = v.$$

However, Reynolds himself later proved that set-theoretic models do not exist (Reynolds, 1984) in classical set-theory (it was later discovered that set theoretic models do exist in some models of intuitionistic set theory (Pitts, 1987; Pitts, 1989)). In 1992 Ma and Reynolds (Ma and Reynolds, 1992) then gave a new formulation of parametricity phrased in terms of more general models (PL-categories of Seely (Seely, 1987)). One may formulate Ma and Reynolds' notion in the language of  $\lambda_2$ -fibrations<sup>†</sup> as follows. The fibration  $E \rightarrow B$  is parametric with respect to a given logic on  $E$  if there exist a reflexive graph of  $\lambda_2$ -fibrations, whose restriction to the fibres over the terminal object is the reflexive graph

$$E_1 \begin{array}{c} \longleftarrow \\ \rightleftarrows \\ \longrightarrow \end{array} \mathbf{LR}(E_1)$$

of logical relations with domain, codomain maps and the middle map mapping a type to the identity on that type. (See (Ma and Reynolds, 1992; Jacobs, 1999) for more details.)

In recent work by Birkedal and Rosolini on parametric domain-theoretic models it became clear that this is not the right categorical formulation of parametricity: it appears that the definition does not allow one to prove the expected consequences of parametricity such as data abstraction and the encoding of data types. Indeed, these consequences have only been proved for specific models, see, e.g., (Wadler, 1989; Hasegawa, 1994), using specific properties of the models.

In this article we propose a new category-theoretic formulation of parametricity, called a *parametric APL-structure*, which *does* allow one to prove the expected properties of

<sup>†</sup> A  $\lambda_2$ -fibration is a fibration with enough properties to model second-order lambda calculus, see, e.g., (Jacobs, 1999).

parametricity in general. We build upon a logic for reasoning about parametricity given by Abadi and Plotkin (Plotkin and Abadi, 1993). In this logic one can formulate parametricity as a schema and prove the expected consequences of parametricity. An APL-structure is a category-theoretic model of Abadi and Plotkin’s logic, for which we prove soundness and completeness, thereby answering a question posed in (Plotkin and Abadi, 1993, Page 5). Each APL-structure contains a model of the second-order lambda calculus, which we may reason about using the logic.

We also provide a completion process that given an internal model of  $\lambda_2$  (see (Hyland et al., 1990; Robinson and Rosolini, 1994)) produces a parametric APL-structure. In special cases, the  $\lambda_2$ -fibration of this APL-structure is the one obtained in (Robinson and Rosolini, 1994) and thus we prove that the models obtained in (Robinson and Rosolini, 1994) in fact satisfy the consequences of parametricity (as expected, but not shown in the literature before).

The consequences of parametricity proved earlier for specific models (Hasegawa, 1994; Wadler, 1989; Dunphy, 2004) all seem to use well-pointedness, i.e., the property, that morphisms  $f : A \rightarrow B$  are determined by their values on global elements  $a : 1 \rightarrow A$ . For parametric APL-structures, we do not need to use well-pointedness to prove the expected consequences of parametricity. Loosely speaking, the point is that our notion of parametric APL-structure includes an appropriate extensional logic to reason with. In *loc. cit.*, the ambient world of set theory is used as the logic and thus extensionality there amounts to asking for well-pointedness. We provide a family of concrete parametric APL-structures, including non-well pointed ones. Thus parametricity *is* useful for proving consequences also for non-well-pointed models.

In subsequent papers we will show how to modify the parametric completion process to produce domain-theoretic parametric models and how to extend the notion of APL-structure to include models of polymorphic *linear* lambda calculus (Plotkin, 1993).

The remainder of the paper is organized as follows. In Section 2, we recall Abadi and Plotkin’s logic. The reader is warned that our version of the logic is slightly different from the one described in (Plotkin and Abadi, 1993). In Section 3 we define the notion of an APL-structure. We prove soundness and completeness with respect to Abadi and Plotkin’s logic in sections 3.1 and 3.2. Section 4 defines the internal language of an APL-structure and we define the notion of a *parametric* APL-structure. We also demonstrate in Section 5 how to use the internal language to show consequences of parametricity in parametric APL-structures. Section 5 mainly contain proofs of well-known results in Abadi & Plotkin’s logic. However, since these proofs are by no means trivial, and to our knowledge do not appear in the literature, and since we think they are of general interest, we include them here.

Section 6 contains a definition of a concrete parametric APL-structure, and we also mention a non-well-pointed parametric APL-structure. Section 7 contains a comparison of our notion of parametricity with the one defined by Ma & Reynolds (Ma and Reynolds, 1992). The parametric completion process is described in Section 8. Since an internal model of  $\lambda_2$  in a quasitopos has ambient logic corresponding to most of the constructions in Abadi & Plotkin’s logic, there exists a natural APL-structure incorporating it, so we may formulate the question if this model is parametric. This is done in Section 9.

Appendix A contains definitions and theory concerning composable fibrations, i.e., pairs of fibrations such that the codomain of the first is the domain of the second. In particular, we study the case of fibrations  $\mathbb{F} \rightarrow \mathbb{E} \rightarrow \mathbb{B}$  where  $\mathbb{F} \rightarrow \mathbb{E}$  is a logic fibration, and we study what is needed for it to model quantification along vertical maps in  $\mathbb{E}$  and quantification along maps in  $\mathbb{B}$ . The definitions of this appendix are used in the definition of an APL-structure.

**Acknowledgments.** We would like to acknowledge helpful discussions with Alex Simpson and Martin Hyland and the constructive comments of the two anonymous referees.

## 2. Abadi & Plotkin's logic

We first recall Abadi & Plotkin's logic for reasoning about parametricity, originally defined in (Plotkin and Abadi, 1993). We will use a slightly modified version of the logic.

Abadi & Plotkin's logic is basically a second-order logic on the second-order  $\lambda$ -calculus ( $\lambda_2$ ). Thus we begin by calling to mind the second order  $\lambda$ -calculus (a more formal presentation can be found in e.g. (Jacobs, 1999)).

### 2.1. Second-order $\lambda$ -calculus

Well-formed type expressions in second-order  $\lambda$ -calculus are expressions of the form:

$$\alpha_1 : \text{Type}, \dots, \alpha_n : \text{Type} \vdash \sigma : \text{Type}$$

where  $\sigma$  is built up from the  $\alpha_i$ 's using products ( $1, \sigma \times \tau$ ), arrows ( $\sigma \rightarrow \tau$ ) and quantification over types. The latter means that if we have a type

$$\alpha_1 : \text{Type}, \dots, \alpha_n : \text{Type} \vdash \sigma : \text{Type},$$

then we may form the type

$$\alpha_1 : \text{Type}, \dots, \alpha_{i-1} : \text{Type}, \alpha_{i+1} : \text{Type}, \dots, \alpha_n : \text{Type} \vdash \prod \alpha_i : \text{Type}. \sigma : \text{Type}$$

We do not allow repetitions in the list of  $\alpha$ 's, and we call this list the kind context. It is often denoted simply  $\Xi$  or  $\vec{\alpha}$ . We use  $\sigma, \tau, \omega$  to range over the set of types.

The terms in  $\lambda_2$  are of the form:

$$\Xi \mid x_1 : \sigma_1, \dots, x_n : \sigma_n \vdash t : \tau$$

where the  $\sigma_i$  and  $\tau$  are well-formed types in the kind context  $\Xi$ . The list of  $x$ 's is called the type context and is often denoted  $\Gamma$ . As for kind contexts we do not accept repetition in type contexts.

The grammar for raw terms is:

$$t ::= x \mid \lambda x : \sigma. t \mid t(t) \mid \star \mid \langle t, t \rangle \mid \pi t \mid \pi' t \mid \Lambda \alpha : \text{Type}. t \mid t(\sigma)$$

corresponding to variables,  $\lambda$ -abstraction, function applications, an element of unit type, pairing and projections on product types and second-order  $\lambda$ -abstractions and type applications. We use  $s, t, u$  to range over the set of terms, and as usual we consider  $\alpha$ -equivalent

$$\begin{array}{c}
\frac{\Xi | \Gamma, x: \sigma \vdash t: \tau \quad \Xi | \Gamma \vdash u: \sigma}{\Xi | \Gamma \vdash (\lambda x: \sigma. t)u = t[u/x]} \beta\text{-reduction} \\
\frac{\Xi, \alpha | \Gamma \vdash t: \tau \quad \Xi \vdash \sigma: \mathbf{Type} \quad \Xi | \Gamma \text{ well-formed}}{\Xi | \Gamma \vdash (\Lambda \alpha: \mathbf{Type}. t)\sigma = t[\sigma/\alpha]} \beta\text{-reduction} \\
\frac{\Xi | \Gamma \vdash t: \sigma \rightarrow \tau}{\Xi | \Gamma \vdash \lambda x: \sigma. (tx) = t} \eta\text{-reduction} \\
\frac{\Xi | \Gamma \vdash t: \prod \alpha: \mathbf{Type}. \sigma}{\Xi | \Gamma \vdash \Lambda \alpha: \mathbf{Type}. (t\alpha) = t} \eta\text{-reduction} \\
\frac{\Xi | \Gamma \vdash t: \sigma \quad \Xi | \Gamma \vdash u: \tau \quad \Xi | \Gamma \vdash t: \sigma \quad \Xi | \Gamma \vdash u: \tau}{\Xi | \Gamma \vdash \pi(t, u) = t \quad \Xi | \Gamma \vdash \pi' \langle t, u \rangle = u} \\
\frac{\Xi | \Gamma \vdash t: \sigma \times \tau \quad \Xi | \Gamma \vdash t: 1}{\Xi | \Gamma \vdash \langle \pi t, \pi' t \rangle = t \quad \Xi | \Gamma \vdash t = \star} \\
\frac{\Xi | \Gamma \vdash t = t': \sigma \quad \Xi | \Gamma, x: \sigma \vdash u: \tau}{\Xi | \Gamma \vdash u[t/x] = u[t'/x]} \text{replacement} \\
\frac{\Xi | \Gamma, x: \sigma \vdash t = s: \tau \quad \Xi, \alpha | \Gamma \vdash t = s \quad \Xi | \Gamma \text{ well-formed}}{\Xi | \Gamma \vdash \lambda x: \sigma. t = \lambda x: \sigma. s \quad \Xi | \Gamma \vdash \Lambda \alpha. t = \Lambda \alpha. s}
\end{array}$$

Fig. 1. Rules for external equality

terms equal. Most of the formation rules are well known from the simply-typed  $\lambda$ -calculus; here we just recall the two additional rules for type abstraction and type application:

$$\frac{\Xi, \alpha: \mathbf{Type} | \Gamma \vdash t: \sigma}{\Xi | \Gamma \vdash \Lambda \alpha: \mathbf{Type}. t: \prod \alpha: \mathbf{Type}. \sigma} \Xi | \Gamma \text{ is well-formed}$$

$$\frac{\Xi | \Gamma \vdash t: \prod \alpha: \mathbf{Type}. \sigma \quad \Xi \vdash \tau: \mathbf{Type}}{\Xi | \Gamma \vdash t(\tau): \sigma[\tau/\alpha]}$$

What we have described above is called the *pure* second-order  $\lambda$ -calculus. In general we will consider second-order  $\lambda$ -calculi based on polymorphic signatures (Jacobs, 1999, 8.1.1). Informally one may think of such a calculus as the pure second-order  $\lambda$ -calculus with added type-constants and term-constants. For instance one may have a constant type for integers or a constant type for lists  $\alpha \vdash \text{lists}(\alpha): \mathbf{Type}$ . We will be particularly interested in the internal language of a  $\lambda_2$ -fibration (see Section 3) which in general will be a non-pure calculus.

2.1.1. *Equality* We consider an equality theory on second-order  $\lambda$ -calculus called *external* equality. It is the least equivalence relation given by the rules in Figure 1.

## 2.2. The logic

Abadi & Plotkin's logic can be built on top of any second-order lambda calculus (based on any polymorphic signature), so in the following we will assume that we are given one such.

Formulas of Abadi & Plotkin's logic live in contexts of elements of  $\lambda_2$  and relations on types of  $\lambda_2$ . The contexts look like

$$\Xi \mid \Gamma \mid R_1 : \text{Rel}(\tau_1, \tau'_1), \dots, R_n : \text{Rel}(\tau_n, \tau'_n),$$

where  $\Xi \mid \Gamma$  is a context of second-order  $\lambda$ -calculus and the  $\tau_i$  and  $\tau'_i$  are well-formed types in context  $\Xi$ , for all  $i$ . The list of  $R$ 's is called the relational context and is often denoted  $\Theta$ . In this context as in the other contexts we do not accept repetitions of variable names. It is important to notice that the relational and type contexts are independent of each other in the sense that one does not affect whether the other is well-formed.

Formulas are given by the syntax:

$$\begin{aligned} \phi ::= & (t =_{\sigma} u) \mid \rho(t, u) \mid \phi \supset \psi \mid \perp \mid \top \mid \phi \wedge \psi \mid \phi \vee \psi \mid \forall \alpha : \text{Type}. \phi \mid \\ & \forall x : \sigma. \phi \mid \forall R : \text{Rel}(\sigma, \tau). \phi \mid \exists \alpha : \text{Type}. \phi \mid \exists x : \sigma. \phi \mid \exists R : \text{Rel}(\sigma, \tau). \phi, \end{aligned}$$

where  $\rho$  is a definable relation (to be discussed below).

In the following we give formation rules for the above. First we have internal equality

$$\frac{\Xi \mid \Gamma \vdash t : \sigma \quad \Xi \mid \Gamma \vdash u : \sigma}{\Xi \mid \Gamma \mid \Theta \vdash (t =_{\sigma} u) : \text{Prop}}$$

Notice here the notational difference between  $t = u$  and  $t =_{\sigma} u$ . The former denotes *external* equality and the latter is a formula in the logic. The rules for  $\supset$ ,  $\vee$  and  $\wedge$  are the usual ones.  $\top$ ,  $\perp$  are formulas in any context.

We have the formation rules for universal quantification:

$$\frac{\Xi \mid \Gamma, x : \sigma, \Gamma' \mid \Theta \vdash \phi : \text{Prop}}{\Xi \mid \Gamma, \Gamma' \mid \Theta \vdash \forall x : \sigma. \phi : \text{Prop}}$$

$$\frac{\Xi \mid \Gamma \mid \Theta, R : \text{Rel}(\sigma, \tau), \Theta' \vdash \phi : \text{Prop}}{\Xi \mid \Gamma \mid \Theta, \Theta' \vdash \forall R : \text{Rel}(\sigma, \tau). \phi : \text{Prop}}$$

$$\frac{\Xi, \alpha, \Xi' \mid \Gamma \mid \Theta \vdash \phi : \text{Prop}}{\Xi, \Xi' \mid \Gamma \mid \Theta \vdash \forall \alpha : \text{Type}. \phi : \text{Prop}} \quad \Xi, \Xi' \mid \Gamma \mid \Theta \text{ is well-formed}$$

The same formation rules apply to the existential quantifier.

## 2.3. Definable relations

Definable relations are given by the grammar:

$$\rho ::= R \mid (x : \sigma, y : \tau). \phi \mid \sigma[\vec{\rho}].$$

A definable relation  $\rho$  always has a domain and a codomain, and we write  $\rho : \text{Rel}(\sigma, \tau)$  to denote that  $\rho$  has domain  $\sigma$  and codomain  $\tau$ . There are 3 rules for this judgement. The

first two are

$$\frac{\frac{\Xi \mid \Gamma \mid \Theta, R: \text{Rel}(\sigma, \tau), \Theta' \vdash R: \text{Rel}(\sigma, \tau)}{\Xi \mid \Gamma, x: \sigma, y: \tau \mid \Theta \vdash \phi: \text{Prop}}}{\Xi \mid \Gamma \mid \Theta \vdash (x: \sigma, y: \tau). \phi: \text{Rel}(\sigma, \tau)}.$$

In the second rule above the variables  $x, y$  become bound in  $\phi$ . For example, we have the equality relation  $eq_\sigma$  defined as  $(x: \sigma, y: \sigma). x =_\sigma y$  and the graph relation of a function  $\langle f \rangle = (x: \sigma, y: \tau). fx =_\tau y$  if  $f: \sigma \rightarrow \tau$ .

The last rule for definable relations is

$$\frac{\alpha_1, \dots, \alpha_n \vdash \sigma: \text{Type} \quad \Xi \mid \Gamma \mid \Theta \vdash \rho_1: \text{Rel}(\tau_1, \tau'_1), \dots, \rho_n: \text{Rel}(\tau_n, \tau'_n)}{\Xi \mid \Gamma \mid \Theta \vdash \sigma[\vec{\rho}]: \text{Rel}(\sigma(\vec{\tau}), \sigma(\vec{\tau}'))}.$$

The notation is a bit ambiguous, since by  $\sigma[\vec{\rho}]$  we mean to substitute each  $\rho_i$  for  $\alpha_i$  in  $\sigma$ , and so the order of the  $\alpha$ 's and the  $\rho$ 's is important. A more precise notation would have been  $\sigma[\rho_1/\alpha_1, \dots, \rho_n/\alpha_n]$ , but we choose to use the more convenient  $\sigma[\vec{\rho}]$ .

Observe that  $\sigma[\vec{\rho}]$  is a syntactic construction and is not obtained by substitution. In (Plotkin and Abadi, 1993)  $\sigma[\vec{\rho}]$  is defined inductively from the structure of  $\sigma$ , but in our case this is not enough, since we will need to form  $\sigma[\vec{\rho}]$  for type constants  $\sigma$  in Section 4. The inductive definition of (Plotkin and Abadi, 1993) is reflected in the rules (12)-(15) below. We call  $\sigma[\vec{\rho}]$  the *relational interpretation of the type  $\sigma$* .

If  $\rho: \text{Rel}(\sigma, \tau)$  is a definable relation, we may apply it to terms of the right types. This gives the last formation rule for formulas

$$\frac{\Xi \mid \Gamma \mid \Theta \vdash \rho: \text{Rel}(\sigma, \tau) \quad \Xi \mid \Gamma \vdash t: \sigma, u: \tau}{\Xi \mid \Gamma \mid \Theta \vdash \rho(t, u): \text{Prop}}.$$

We will also write  $t\rho u$  for  $\rho(t, u)$ .

**Lemma 2.1.** Suppose  $\Xi \mid \Gamma \vdash \Theta, R: \text{Rel}(\sigma, \tau) \vdash \phi: \text{Prop}$  and  $\Xi \mid \Gamma \mid \Theta \vdash \rho: \text{Rel}(\sigma, \tau)$  are well-formed. Then

$$\Xi \mid \Gamma \mid \Theta \vdash \phi[\rho/R]: \text{Prop}$$

is well-formed.

*Proof.* Easy induction on the structure of  $\phi$ . □

**Remark 2.2.** Abadi & Plotkin's logic is designed for reasoning about binary relational parametricity. For reasoning about other arities of parametricity (such as unary parametricity), one can easily replace binary relations in the logic by relations of other arities. In the case of unary parametricity, for example, one would then have an interpretation of types as predicates. See also (Takeuti, 1998; Wadler, 2004)

We introduce the short notation  $\rho \equiv \rho'$  for definable relations  $\rho: \text{Rel}(\sigma, \tau)$ ,  $\rho': \text{Rel}(\sigma, \tau)$  as  $\forall x: \sigma, y: \tau. \rho(x, y) \supseteq \rho'(x, y)$ . Notice that we use  $\supseteq$  for biimplication.

We can take exponents, products and universal quantification of relations. These constructions will turn out to define categorical exponents, products and quantification in a



$$\begin{array}{l}
\Xi: \text{Ctx} \quad \Xi \vdash \sigma: \text{Type} \quad \Xi \mid \Gamma: \text{Ctx} \\
\Xi \mid \Theta: \text{Ctx} \quad \Xi \mid \Gamma \vdash t: \sigma \quad \Xi \mid \Gamma \vdash t = u \\
\Xi \mid \Gamma \mid \Theta \vdash \phi: \text{Prop} \quad \Xi \mid \Gamma \mid \Theta \vdash \rho: \text{Rel}(\sigma, \tau) \quad \Xi \mid \Gamma \mid \Theta \mid \phi_1, \dots, \phi_n \vdash \psi
\end{array}$$

Fig. 2. Types of judgements

category of relations (see Lemma 3.7). For now, the reader should just consider the next three definitions as shorthand notation.

If  $\rho: \text{Rel}(\sigma, \tau)$  and  $\rho': \text{Rel}(\sigma', \tau')$  we may define a definable relation:

$$(\rho \rightarrow \rho'): \text{Rel}((\sigma \rightarrow \sigma'), (\tau \rightarrow \tau'))$$

as

$$\rho \rightarrow \rho' = (f: \sigma \rightarrow \sigma', g: \tau \rightarrow \tau'). \forall x: \sigma. \forall y: \tau. (x\rho y \supset (fx)\rho'(gy))$$

We may also take the product of  $\rho$  and  $\rho'$ :

$$\rho \times \rho': \text{Rel}((\sigma \times \sigma'), (\tau \times \tau'))$$

as

$$\rho \times \rho' = (x: \sigma \times \sigma', y: \tau \times \tau'). (\pi x)\rho(\pi y) \wedge (\pi' x)\rho'(\pi' y)$$

If

$$\Xi, \alpha, \beta \mid \Gamma \mid \Theta, R: \text{Rel}(\alpha, \beta) \vdash \rho: \text{Rel}(\sigma, \tau)$$

is well-formed and  $\Xi \mid \Gamma \mid \Theta$  and  $\Xi, \alpha \vdash \sigma: \text{Type}$  and  $\Xi, \beta \vdash \tau: \text{Type}$  we may define:

$$\Xi \mid \Gamma \mid \Theta \vdash \forall(\alpha, \beta, R: \text{Rel}(\alpha, \beta)). \rho: \text{Rel}((\prod \alpha: \text{Type}. \sigma), (\prod \beta: \text{Type}. \tau))$$

as

$$(t: \prod \alpha: \text{Type}. \sigma, u: \prod \beta: \text{Type}. \tau). \forall \alpha, \beta: \text{Type}. \forall R: \text{Rel}(\alpha, \beta). (t\alpha)\rho(u\beta).$$

#### 2.4. The axioms

Figure 2 sums up the types of judgements we have in the logic. The last judgement in the figure says that in the given context, the conjunction of the formulas  $\phi_1, \dots, \phi_n$  implies  $\psi$ .

Having specified the language of Abadi & Plotkin's logic, it is time to specify the axioms and the rules of the logic. We have all the axioms of propositional logic plus the rules specified below.

We have rules for  $\forall$ -quantification:

$$\frac{\Xi, \alpha \mid \Gamma \mid \Theta \mid \Phi \vdash \psi}{\Xi \mid \Gamma \mid \Theta \mid \Phi \vdash \forall \alpha: \text{Type}. \psi} \Xi \mid \Gamma \mid \Theta \vdash \Phi \quad (1)$$

$$\frac{\Xi \mid \Gamma, x: \sigma \mid \Theta \mid \Phi \vdash \psi}{\Xi \mid \Gamma \mid \Theta \mid \Phi \vdash \forall x: \sigma. \psi} \Xi \mid \Gamma \mid \Theta \vdash \Phi \quad (2)$$

$$\frac{\Xi \mid \Gamma \mid \Theta, R: \text{Rel}(\tau, \tau') \mid \Phi \vdash \psi}{\Xi \mid \Gamma \mid \Theta \mid \Phi \vdash \forall R: \text{Rel}(\tau, \tau').\psi} \Xi \mid \Gamma \mid \Theta \vdash \Phi \quad (3)$$

The double bars mean that these are double rules, i.e., the condition on the bottom implies the one on top and vice versa.

Rules for  $\exists$ -quantification:

$$\frac{\Xi, \alpha \mid \Gamma \mid \Theta \mid \phi \vdash \psi}{\Xi \mid \Gamma \mid \Theta \mid \exists \alpha: \text{Type}.\phi \vdash \psi} \Xi \mid \Gamma \mid \Theta \vdash \psi \quad (4)$$

$$\frac{\Xi \mid \Gamma, x: \sigma \mid \Theta \mid \phi \vdash \psi}{\Xi \mid \Gamma \mid \Theta \mid \exists x: \sigma.\phi \vdash \psi} \Xi \mid \Gamma \mid \Theta \vdash \psi \quad (5)$$

$$\frac{\Xi \mid \Gamma \mid \Theta, R: \text{Rel}(\tau, \tau') \mid \phi \vdash \psi}{\Xi \mid \Gamma \mid \Theta \mid \exists R: \text{Rel}(\tau, \tau').\phi \vdash \psi} \Xi \mid \Gamma \mid \Theta \vdash \psi \quad (6)$$

We have substitution rules

$$\frac{\Xi, \alpha \mid \Gamma \mid \Theta \mid \Psi \vdash \phi \quad \Xi \vdash \sigma: \text{Type}}{\Xi \mid \Gamma[\sigma/\alpha] \mid \Theta[\sigma/\alpha] \mid \Psi[\sigma/\alpha] \vdash \phi[\sigma/\alpha]} \quad (7)$$

$$\frac{\Xi \mid \Gamma, x: \sigma \mid \Theta \mid \Psi \vdash \phi \quad \Xi \mid \Gamma \vdash t: \sigma}{\Xi \mid \Gamma \mid \Theta \mid \Psi[t/x] \vdash \phi[t/x]} \quad (8)$$

$$\frac{\Xi \mid \Gamma \mid \Theta, R: \text{Rel}(\sigma, \tau) \mid \Psi \vdash \phi \quad \Xi \mid \Gamma \mid \Theta \vdash \rho: \text{Rel}(\sigma, \tau)}{\Xi \mid \Gamma \mid \Theta \mid \Psi[\rho/R] \vdash \phi[\rho/R]} \quad (9)$$

The *substitution* axiom:

$$\frac{\Xi \mid \Gamma \mid \Theta \mid \top \vdash \forall \alpha, \beta: \text{Type}.\forall x, x': \alpha.\forall y, y': \beta.\forall R: \text{Rel}(\alpha, \beta). R(x, y) \wedge x =_{\alpha} x' \wedge y =_{\beta} y' \supset R(x', y')}{\Xi \mid \Gamma \mid \Theta \mid \top \vdash \forall \alpha, \beta: \text{Type}.\forall x, x': \alpha.\forall y, y': \beta.\forall R: \text{Rel}(\alpha, \beta). R(x, y) \wedge x =_{\alpha} x' \wedge y =_{\beta} y' \supset R(x', y')} \quad (10)$$

External equality implies internal equality:

$$\frac{\Xi \mid \Gamma \vdash t = u: \sigma}{\Xi \mid \Gamma \mid \Theta \mid \top \vdash t =_{\sigma} u} \quad (11)$$

We omit the obvious rules stating that internal equality is an equivalence relation. The following rules concern the interpretation of types as relations.

$$\frac{}{\Xi \mid \Gamma \mid \Theta \mid \top \vdash \forall x, y: 1. x1y} \quad (12)$$

$$\frac{\vec{\alpha} \vdash \alpha_i \quad \Xi \mid \Gamma \mid \Theta \vdash \vec{\rho}: \text{Rel}(\vec{\tau}, \vec{\tau}')}{\Xi \mid \Gamma \mid \Theta \mid \top \vdash \alpha_i[\vec{\rho}] \equiv \rho_i} \quad (13)$$

$$\frac{\vec{\alpha} \vdash \sigma \rightarrow \sigma' \quad \Xi \mid \Theta \vdash \vec{\rho}: \text{Rel}(\vec{\tau}, \vec{\tau}')}{\Xi \mid \Gamma \mid \Theta \mid \top \vdash (\sigma \rightarrow \sigma')[\vec{\rho}] \equiv (\sigma[\vec{\rho}] \rightarrow \sigma'[\vec{\rho}])} \quad (14)$$

$$\frac{\bar{\alpha} \vdash \prod \beta. \sigma(\bar{\alpha}, \beta) \quad \Xi \mid \Theta \vdash \bar{\rho}: \text{Rel}(\bar{\tau}, \bar{\tau}')}{\Xi \mid \Gamma \mid \Theta \mid \top \vdash (\forall \beta. \sigma(\bar{\alpha}, \beta))[\bar{\rho}] \equiv \forall(\beta, \beta', R: \text{Rel}(\beta, \beta')). \sigma[\bar{\rho}, R]} \quad (15)$$

Finally we have

$$\frac{\Xi \mid \Gamma, x: \sigma, y: \tau \mid \Theta \vdash \phi: \text{Prop} \quad \Xi \mid \Gamma \vdash t: \sigma, u: \tau}{\Xi \mid \Gamma \vdash ((x: \sigma, y: \tau). \phi)(t, u) \preceq \phi[t, u/x, y]}. \quad (16)$$

Using this rule, we may prove a bijective correspondence between definable relations and propositions with two free variables considered up to provable equivalence. The bijection maps a definable relation  $\rho$  to the formula  $\rho(x, y)$  with free variables  $x, y$  and a formula  $\phi$  with free variables  $x, y$  to the definable relation  $(x, y). \phi$ .

**Lemma 2.3.** Suppose  $\Xi \mid \Gamma \mid \Theta \vdash \rho: \text{Rel}(\sigma, \tau)$  and  $\Xi \mid \Gamma, x: \sigma, y: \tau \mid \Theta \vdash \phi: \text{Prop}$ . Then

$$\Xi \mid \Gamma, x: \sigma, y: \tau \mid \Theta \mid \top \vdash \phi \preceq ((x: \sigma, y: \tau). \phi)(x, y)$$

and

$$\Xi \mid \Gamma \mid \Theta \mid \top \vdash \rho \equiv (x: \sigma, y: \tau). \rho(x, y).$$

*Proof.* The first statement above is just a reformulation of (16), and for the second we need to prove that

$$\forall x: \sigma, y: \tau. ((x: \sigma, y: \tau). \rho(x, y))(x, y) \preceq \rho(x, y)$$

which is also an easy consequence of (16).  $\square$

We would also like to mention the extensionality schemes:

$$\begin{aligned} (\forall x: \sigma. t x =_{\tau} u x) &\supset t =_{\sigma \rightarrow \tau} u \\ (\forall \alpha: \text{Type}. t \alpha =_{\tau} u \alpha) &\supset t =_{\prod \alpha: \text{Type}. \tau} u. \end{aligned}$$

These are taken as axioms in (Plotkin and Abadi, 1993), but we shall not take these as axioms as we would like to be able to talk about models that are not necessarily extensional.

**Lemma 2.4.** The substitution axiom above implies the *replacement* rule:

$$\frac{\Xi \mid \Gamma \mid \Theta \mid \Phi \vdash t =_{\sigma} t' \quad \Xi \mid \Gamma, x: \sigma \vdash u: \tau}{\Xi \mid \Gamma \mid \Theta \mid \Phi \vdash u[t/x] =_{\tau} u[t'/x]}$$

*Proof.* Instantiate the substitution axiom with the definable relation

$$\rho = (y: \sigma, z: \sigma). u[y/x] =_{\tau} u[z/x].$$

Clearly  $\Phi \vdash \rho(t, t)$ , so since  $t =_{\sigma} t'$ , we have  $\Phi \vdash \rho(t, t')$  as desired.  $\square$

**Lemma 2.5 (Weakening, Exchange).** If  $\Xi \mid \Gamma \mid \Theta \mid \Psi \vdash \phi$  is provable in the logic, and if further  $\Xi' \mid \Gamma' \mid \Theta'$  is a context obtained from  $\Xi \mid \Gamma \mid \Theta$  by permuting the order of the variables in the contexts, and possibly adding variables, then

$$\Xi' \mid \Gamma' \mid \Theta' \mid \Psi \vdash \phi$$

is also provable in the logic.

### 3. APL-structures

In this section we define the notion of an APL-structure, which is basically a category-theoretic formulation of a model of Abadi & Plotkin's logic. We also show how to interpret the logic in an APL-structure. We use the definitions and results of Appendix A.

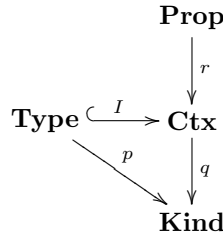
But first we recall the notion of a  $\lambda_2$ -fibration, which is basically a model of  $\lambda_2$ .

**Definition 3.1.** A fibration  $\mathbf{Type} \rightarrow \mathbf{Kind}$  is a  $\lambda_2$ -fibration if it is fibred cartesian closed, has a generic object  $\Omega \in \mathbf{Kind}$ , products in  $\mathbf{Kind}$ , and simple  $\Omega$ -products, i.e., right adjoints  $\prod_{\pi}$  to the reindexing functors  $\pi^*$  for projections  $\pi: \Xi \times \Omega \rightarrow \Xi$ .

**Remark 3.2.** In a  $\lambda_2$  fibration, for a map  $f: \Xi \rightarrow \Omega$  in  $\mathbf{Kind}$ , we will use the notation  $\hat{f}$  to denote the object of  $\mathbf{Type}_{\Xi}$  corresponding to  $f$ , and likewise for  $\sigma \in \mathbf{Type}_{\Xi}$  we write  $\hat{\sigma}: \Xi \rightarrow \Omega$  for the map corresponding to  $\sigma$ .

**Definition 3.3.** A pre-APL-structure consists of

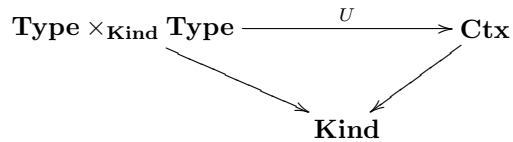
- 1 Fibrations:



where

- $p$  is a  $\lambda_2$ -fibration.
- $q$  is a fibration with fibred products
- $(r, q)$  is an indexed first-order logic fibration (Definition A.4) which has products and coproducts with respect to  $\Xi \times \Omega \rightarrow \Xi$  in  $\mathbf{Kind}$  (Definition A.5) where  $\Omega$  is the generic object of  $p$ .
- $I$  is a faithful product preserving map of fibrations.

- 2 a contravariant morphism of fibrations:



- 3 a family of bijections

$$\Psi_{\Xi} : \mathrm{Hom}_{\mathbf{Ctx}_{\Xi}}(\xi, U(\sigma, \tau)) \rightarrow \mathrm{Obj}(\mathbf{Prop}_{\xi \times I(\sigma \times \tau)})$$

for  $\sigma$  and  $\tau$  in  $\mathbf{Type}_{\Xi}$  and  $\xi$  in  $\mathbf{Ctx}_{\Xi}$ , which

- is natural in the  $\xi, \sigma, \tau$
- commutes with reindexing functors; that is, if  $\rho: \Xi' \rightarrow \Xi$  is a morphism in  $\mathbf{Kind}$

and  $u : \xi \rightarrow U(\sigma, \tau)$  is a morphism in  $\mathbf{Ctx}_\Xi$ , then

$$\Psi_{\Xi'}(\rho^*(u)) = (\bar{\rho})^*(\Psi_\Xi(u))$$

where  $\bar{\rho}$  is the cartesian lift of  $\rho$ .

Notice that  $\Psi$  is only defined on vertical morphisms.

By a contravariant functor of fibrations, we mean a functor of fibrations, which is contravariant in each fibre.

**Remark 3.4.** Item 3 implies that  $(U(1_\Xi, 1_\Xi))_{\Xi \in \mathbf{Kind}}$  is an indexed family of generic objects. If, on the other hand, we have an indexed family of generic objects  $(\Sigma_\Xi)_{\Xi \in \mathbf{Kind}}$  and  $\mathbf{Ctx}$  is cartesian closed, then we may define  $U$  to be  $\Sigma^{-\times-}$  and thereby get items 2 and 3 for free. In general, however,  $\mathbf{Ctx}$  will not be cartesian closed. In particular, in the syntactic model described below in the proof of completeness  $\mathbf{Ctx}$  is not cartesian closed.

**Remark 3.5.** Below we will describe how the  $U(\sigma, \tau)$  is used to model the object of relations from  $\sigma$  to  $\tau$ . To model a version of Abadi & Plotkin's logic for unary or any other arity of parametricity as in Remark 2.2, the functor  $U$  should have corresponding arity and the domain and codomain of the bijection  $\Psi$  should be changed accordingly.

We now explain how to interpret all of Abadi & Plotkin's logic, except for the relational interpretation of types, in a pre-APL-structure. First we recall the interpretation of  $\lambda_2$  in a  $\lambda_2$ -fibration.

A type  $\alpha_1 \dots \alpha_n \vdash \alpha_i$  is interpreted as the object of **Type** over  $\Omega^n$  corresponding to the  $i$ 'th projection  $\Omega^n \rightarrow \Omega$ . For a type  $\alpha_1 \dots \alpha_n \vdash \sigma$ , we have  $\llbracket \prod \alpha_i. \sigma \rrbracket = \prod_{\pi} \llbracket \vec{\alpha} \vdash \sigma \rrbracket$ , where  $\pi$  is the projection forgetting the  $i$ 'th coordinate. Since each fibre of the  $\lambda_2$ -fibration is cartesian closed, we may interpret the constructions of the simply typed  $\lambda$ -calculus using fibrewise constructions.

If  $\Xi, \alpha \mid \Gamma \vdash t : \tau$  is a term and  $\Xi \vdash \Gamma$  is well-formed, then we may interpret the term  $\Xi \mid \Gamma \vdash \Lambda \alpha. t : \prod \alpha. \tau$  as the morphism corresponding to  $\llbracket \Xi, \alpha \mid \Gamma \vdash t : \tau \rrbracket$  under the adjunction  $\pi^* \dashv \prod_{\pi}$ .

To interpret  $\Xi \mid \Gamma \vdash t \sigma$ , notice that  $\llbracket \Xi \vdash \sigma \rrbracket$  corresponds to a map

$$\widehat{\llbracket \Xi \vdash \sigma \rrbracket} : \llbracket \Xi \rrbracket \rightarrow \Omega.$$

The morphism  $\llbracket \Xi \mid \Gamma \vdash t : \prod \alpha. \tau \rrbracket$  corresponds by the adjunction  $\pi^* \dashv \prod_{\pi}$  to a morphism in the fibre over  $\llbracket \Xi \rrbracket \times \Omega$ . We reindex this morphism along

$$\langle id_{\llbracket \Xi \rrbracket}, \widehat{\llbracket \Xi \vdash \sigma \rrbracket} \rangle : \llbracket \Xi \rrbracket \rightarrow \llbracket \Xi \rrbracket \times \Omega$$

to get  $\llbracket \Xi \mid \Gamma \vdash t \sigma \rrbracket$ .

Relational contexts are interpreted in **Ctx** as:

$$\llbracket \Xi \mid R_1 : \mathbf{Rel}(\sigma_1, \tau_1), \dots, R_n : \mathbf{Rel}(\sigma_n, \tau_n) \rrbracket = U(\llbracket \sigma_1 \rrbracket, \llbracket \tau_1 \rrbracket) \times \dots \times U(\llbracket \sigma_n \rrbracket, \llbracket \tau_n \rrbracket),$$

where  $\llbracket \sigma_i \rrbracket, \llbracket \tau_i \rrbracket$  are the interpretations of the types in **Type** as described above.

We aim to define  $\llbracket \Xi \mid \Gamma \mid \Theta \vdash \phi \rrbracket$  as an object of **Prop** over  $\llbracket \Xi \mid \Gamma \mid \Theta \rrbracket$ , which we

define to be  $I(\llbracket \Xi \mid \Gamma \rrbracket) \times \llbracket \Xi \mid \Theta \rrbracket$ . We proceed by induction on the structure of  $\phi$ . We use the short notation  $\llbracket \Xi \mid \Gamma \mid \Theta \vdash t : \tau \rrbracket$  for the composition

$$\llbracket \Xi \mid \Gamma \mid \Theta \rrbracket \xrightarrow{\pi} I(\llbracket \Xi \mid \Gamma \rrbracket) \xrightarrow{I(\llbracket \Xi \mid \Gamma \vdash t : \tau \rrbracket)} I(\llbracket \Xi \vdash \tau \rrbracket),$$

and we will in the following leave obvious isomorphisms involving products implicit.

If we define  $\Delta_X : X \rightarrow X \times X$  to be the diagonal map, then

$$\llbracket \Xi \mid x : \sigma, y : \sigma \mid - \vdash x =_\sigma y : \mathbf{Prop} \rrbracket = \coprod_{\Delta_{I(\llbracket \sigma \rrbracket)}} (\top)$$

and

$$\begin{aligned} & \llbracket \Xi \mid \Gamma \mid \Theta \mid t =_\sigma u \rrbracket = \\ & \langle \llbracket \Xi \mid \Gamma \mid \Theta \vdash t \rrbracket, \llbracket \Xi \mid \Gamma \mid \Theta \vdash u \rrbracket \rangle^* \llbracket \Xi \mid x : \sigma, y : \sigma \mid - \vdash x =_\sigma y : \mathbf{Prop} \rrbracket. \end{aligned}$$

$\forall x : A.\phi$  and  $\forall R : \mathbf{Rel}(\sigma, \tau).\phi$  are interpreted using right adjoints to reindexing functors related to the appropriate projections in **Ctx**. Likewise  $\exists x : A.\phi$  and  $\exists R : \mathbf{Rel}(\sigma, \tau).\phi$  are interpreted using left adjoints to the same reindexing functors.

$\forall \alpha.\phi$  and  $\exists \alpha.\phi$  are interpreted using respectively right and left adjoints to  $\bar{\pi}^*$  where  $\bar{\pi}$  is the lift of the projection  $\pi : \llbracket \Xi, \alpha : \mathbf{Type} \rrbracket \rightarrow \llbracket \Xi \rrbracket$  in **Kind** to **Ctx**.

Definable relations are interpreted as maps in **Ctx**. To be more precise, a definable relation  $\Xi \mid \Gamma \mid \Theta \vdash \rho : \mathbf{Rel}(\sigma, \tau)$  is interpreted as a morphism from  $\llbracket \Xi \mid \Gamma \mid \Theta \rrbracket$  to  $U(\llbracket \sigma \rrbracket, \llbracket \tau \rrbracket)$ . The definable relation  $\Xi \mid \Gamma \mid \Theta, R : \mathbf{Rel}(\sigma, \tau), \Theta' \vdash R : \mathbf{Rel}(\sigma, \tau)$  is interpreted as the projection. We define

$$\llbracket \Xi \mid \Gamma \mid \Theta \vdash (x : \sigma, y : \tau).\phi : \mathbf{Rel}(\sigma, \tau) \rrbracket = \Psi^{-1} \llbracket \Xi \mid \Gamma, x : \sigma, y : \tau \mid \Theta \vdash \phi \rrbracket.$$

We define the interpretation of application of definable relations to terms as follows:

$$\llbracket \Xi \mid \Gamma, x : \sigma, y : \tau \mid \Theta \vdash \rho(x, y) \rrbracket = \Psi(\llbracket \Xi \mid \Gamma \mid \Theta \vdash \rho : \mathbf{Rel}(\sigma, \tau) \rrbracket).$$

Finally

$$\begin{aligned} & \llbracket \Xi \mid \Gamma \mid \Theta \vdash \rho(t, u) \rrbracket = \\ & \langle \pi, id, \llbracket \Xi \mid \Gamma \mid \Theta \vdash t \rrbracket, \llbracket \Xi \mid \Gamma \mid \Theta \vdash u \rrbracket, \pi' \rangle^* \llbracket \Xi \mid \Gamma, x : \sigma, y : \tau \mid \Theta \vdash \rho(x, y) \rrbracket \end{aligned}$$

where  $\pi : \llbracket \Xi \mid \Gamma \mid \Theta \rrbracket \rightarrow I(\llbracket \Xi \mid \Gamma \rrbracket)$  and  $\pi' : \llbracket \Xi \mid \Gamma \mid \Theta \rrbracket \rightarrow \llbracket \Xi \mid - \mid \Theta \rrbracket$  are the projections. As usual, we have left out some obvious isomorphisms here.

To interpret the relational interpretation of types we need a little more structure. First we consider a fibration

$$\mathbf{Relations} \rightarrow \mathbf{RelCtx},$$

that can be defined for every pre-APL-structure. **RelCtx** is defined as the pullback

$$\begin{array}{ccc} \mathbf{RelCtx} & \longrightarrow & \mathbf{Ctx} \\ \downarrow \lrcorner & & \downarrow \\ \mathbf{Kind} \times \mathbf{Kind} & \xrightarrow{\times} & \mathbf{Kind} \end{array}$$

If  $\Theta$  is an object of **RelCtx** projecting to  $(\Xi, \Xi') \in \mathbf{Kind} \times \mathbf{Kind}$ , we will write it as  $\Xi, \Xi' \mid \Theta$ . The fibre of **Relations** over  $\Xi, \Xi' \mid \Theta$  is

**objects**      Triples  $(\sigma, \tau, \rho)$ , where  $\sigma$  is an object in  $\mathbf{Type}_{\Xi}$ ,  $\tau$  is an object in  $\mathbf{Type}_{\Xi'}$

and  $\rho$  is a map  $\rho: \Theta \rightarrow U(\pi^*\sigma, (\pi')^*\tau)$ , where  $\pi, \pi'$  are the projections out of  $\Xi \times \Xi'$ .

**morphisms** A morphism from  $(\sigma, \tau, \rho)$  to  $(\sigma', \tau', \rho')$  is a pair of morphisms  $(s, t)$ , such that  $s: \sigma \rightarrow \sigma'$  and  $t: \tau \rightarrow \tau'$ , and

$$\Psi(U(\pi^*t, (\pi')^*s) \circ \rho') \leq \Psi(\rho)$$

where the ordering refers to the fibrewise ordering on **Prop**.

Reindexing  $(\sigma, \tau, \rho)$  along a vertical map  $\Theta' \rightarrow \Theta$  in **RelCtx** (vertical with respect to **Kind**  $\times$  **Kind**) is given by composition. Reindexing with respect to lifts of maps  $(\omega, \omega'): (\Xi_1, \Xi'_1) \rightarrow (\Xi_2, \Xi'_2)$  is given by reindexing in **Ctx**  $\rightarrow$  **Kind**.

**Remark 3.6.** In the internal language, objects of **Relations** are simply relations

$$\Xi, \Xi' \mid \Theta \vdash \rho: \text{Rel}(\sigma(\Xi), \tau(\Xi')),$$

and a morphism from  $\rho: \text{Rel}(\sigma(\Xi), \tau(\Xi'))$  to  $\rho': \text{Rel}(\sigma'(\Xi), \tau'(\Xi'))$  is simply a pair of morphisms  $t: \sigma \rightarrow \sigma'$  in **Type** $_{\Xi}$  and  $s: \tau \rightarrow \tau'$  in **Type** $_{\Xi'}$  such that

$$\forall x, y. \rho(x, y) \supset \rho'(t x, s y).$$

We clearly have two functors **RelCtx**  $\rightarrow$  **Kind** defined by mapping  $(\Xi, \Xi', \Theta)$  to  $\Xi$  and  $\Xi'$  respectively, and we also have two functors **Relations**  $\rightarrow$  **Type** defined by mapping  $(\phi, \sigma, \tau)$  to  $\sigma$  and  $\tau$  respectively.

**Lemma 3.7.** The fibration **Relations**  $\rightarrow$  **RelCtx** is a  $\lambda_2$ -fibration, and the maps mentioned above define a pair of maps of  $\lambda_2$  fibrations

$$\begin{array}{ccc} \mathbf{Type} & \begin{array}{c} \xleftarrow{\partial_0} \\ \xleftarrow{\partial_1} \end{array} & \mathbf{Relations} \\ \downarrow & & \downarrow \\ \mathbf{Kind} & \begin{array}{c} \xleftarrow{\partial_0} \\ \xleftarrow{\partial_1} \end{array} & \mathbf{RelCtx}. \end{array}$$

*Proof.* The category **RelCtx** has products:

$$\begin{aligned} & (\Xi_1, \Xi'_1, \Theta) \times (\Xi_2, \Xi'_2, \Theta') = \\ & (\Xi_1 \times \Xi_2, \Xi'_1 \times \Xi'_2, (\pi, \pi)^*\Theta \times (\pi', \pi')^*\Theta'). \end{aligned}$$

where  $(\pi, \pi): (\Xi_1 \times \Xi_2, \Xi'_1 \times \Xi'_2) \rightarrow (\Xi_1, \Xi'_1)$  is the projection, and  $(\pi', \pi')$  is the other evident projection.

The fibration has a generic object  $(\Omega, \Omega, U(\widehat{id_{\Omega}}, \widehat{id_{\Omega}}))$ , since morphism into this from  $(\Xi, \Xi', \Theta)$  in **RelCtx** consists of pairs of types  $(f: \Xi \rightarrow \Omega, g: \Xi' \rightarrow \Omega)$  and vertical morphisms from  $\Theta$  to  $U(\hat{f}, \hat{g})$ . These are exactly the objects of **Relations**.

The constructions for fibred products, fibred exponents and simple  $\Omega$ -products are simply the rules for products, exponents and universal quantification of relations in Abadi & Plotkin's logic formulated in the internal language of the model, which we will describe in Section 4. One can either interpret these constructions in the pre-APL-structure, and prove directly that these constructions have the desired properties, or one can use the

fact that pre-APL-structures interpret these constructions soundly (Theorem 3.10) and reason in the internal logic.

Here we give the rest of the proof reasoning in the internal logic. Suppose  $\rho: \mathbf{Rel}(\sigma, \tau)$  and  $\rho': \mathbf{Rel}(\sigma', \tau')$  and  $\omega: \mathbf{Rel}(\sigma'', \tau'')$  are objects in some fibre of **Relations**. Then a vertical morphism from  $\omega$  to

$$\rho \times \rho': \mathbf{Rel}((\sigma \times \sigma'), (\tau \times \tau')),$$

defined as

$$(x, x')\rho \times \rho'(y, y') = x\rho y \wedge x'\rho' y',$$

is a pair of maps  $t: \sigma'' \rightarrow \sigma \times \sigma'$  and  $u: \tau'' \rightarrow \tau \times \tau'$  such that

$$\forall x, y. x\omega y \supset \pi(tx)\rho\pi(uy) \wedge \pi'(tx)\rho'\pi'(uy),$$

which is the same as a pair of maps from  $\omega$  into  $\rho$  and  $\rho'$  respectively.

Likewise maps from  $\omega$  into

$$(\rho \rightarrow \rho'): \mathbf{Rel}((\sigma \rightarrow \sigma'), (\tau \rightarrow \tau')),$$

defined as

$$f(\rho \rightarrow \rho')g = \forall x: \sigma \forall y: \tau (x\rho y \supset (fx)\rho'(gy)),$$

are in one-to-one correspondence with maps from  $\omega \times \rho$  to  $\rho'$ .

Given new relations  $\Xi, \Xi' \mid \Theta \vdash \omega: \mathbf{Rel}(\sigma, \sigma')$  and

$$\Xi, \alpha; \Xi', \beta \mid \Theta, R: \mathbf{Rel}(\alpha, \beta) \vdash \rho: \mathbf{Rel}(\tau, \tau'),$$

we have defined

$$\Xi, \Xi' \mid \Theta \vdash \forall(\alpha, \beta, R: \mathbf{Rel}(\alpha, \beta)). \rho: \mathbf{Rel}((\prod \alpha: \mathbf{Type}. \tau), (\prod \beta: \mathbf{Type}. \tau'))$$

as

$$(t: \prod \alpha. \tau, \prod \beta. \tau'). \forall \alpha, \beta: \mathbf{Type}. \forall R: \mathbf{Rel}(\alpha, \beta). (t\alpha)\rho(u\beta).$$

We need to show that this defines a right adjoint to weakening. The idea is that the correspondence between maps will be the same as in **Type**  $\rightarrow$  **Kind**. In this fibration, the correspondence is given as follows, a map  $\Xi, \alpha \mid - \vdash t: \sigma \rightarrow \tau$  with  $\Xi \vdash \sigma: \mathbf{Type}$  corresponds to  $\Xi \mid - \vdash \hat{t}: \sigma \rightarrow \prod \alpha. \tau$  where  $\hat{t} = \lambda x: \sigma. \Lambda \alpha. (tx)$ . We will show, that  $(t, u)$  preserves relations iff  $(\hat{t}, \hat{u})$  does. It is clear that

$$\Xi, \alpha; \Xi', \beta \mid x: \sigma, y: \sigma' \mid \Theta, R: \mathbf{Rel}(\alpha, \beta) \mid x\omega y \vdash (tx)\rho(uy)$$

iff

$$\Xi, \Xi' \mid x: \sigma, y: \sigma' \mid \Theta \mid x\omega y \vdash \forall \alpha, \beta: \mathbf{Type}. \forall R: \mathbf{Rel}(\alpha, \beta). (\hat{t}x\alpha)\rho(\hat{u}y\beta),$$

which establishes the bijective correspondence.  $\square$

**Definition 3.8.** An **APL-structure** is a pre-APL-structure for which the graph of 3.7



can be extended to a reflexive graph of  $\lambda_2$ -fibrations

$$\begin{array}{ccc}
 \mathbf{Type} & \begin{array}{c} \xleftarrow{\partial_0} \\ \xrightarrow{J} \\ \xleftarrow{\partial_1} \end{array} & \mathbf{Relations} \\
 \downarrow & & \downarrow \\
 \mathbf{Kind} & \begin{array}{c} \xleftarrow{\partial_0} \\ \xrightarrow{J} \\ \xleftarrow{\partial_1} \end{array} & \mathbf{RelCtx},
 \end{array}$$

i.e., there exists a map  $J$  of  $\lambda_2$ -fibrations such that  $\partial_0 J = id = \partial_1 J$ .

**Remark 3.9.** There is a functor from **Relations** to **Prop** mapping an object  $(\sigma, \tau, \rho)$  to  $\Psi(\rho)$ . In the following we often use that functor implicitly.

We need to show how to interpret the rule

$$\frac{\alpha_1, \dots, \alpha_n \vdash \sigma(\vec{\alpha}) : \mathbf{Type} \quad \Xi \mid \Gamma \mid \Theta \vdash \rho_1 : \mathbf{Rel}(\tau_1, \tau'_1), \dots, \rho_n : \mathbf{Rel}(\tau_n, \tau'_n)}{\Xi \mid \Gamma \mid \Theta \vdash \sigma[\vec{\rho}] : \mathbf{Rel}(\sigma(\vec{\tau}), \sigma(\vec{\tau}'))}$$

in an APL-structure.

Since  $J$  preserves products and generic objects,  $J(\llbracket \vec{\alpha} \vdash \sigma(\vec{\alpha}) \rrbracket)$  is a definable relation of the form  $\llbracket \vec{\alpha}; \vec{\beta} \mid - \mid \vec{R} : \mathbf{Rel}(\vec{\alpha}, \vec{\beta}) \vdash J(\sigma) : \mathbf{Rel}(\sigma(\vec{\alpha}), \sigma(\vec{\beta})) \rrbracket$ . It thus makes sense to define

$$\llbracket \vec{\alpha}, \vec{\beta} \mid - \mid \vec{R} : \mathbf{Rel}(\vec{\alpha}, \vec{\beta}) \vdash \sigma[\vec{R}] : \mathbf{Rel}(\sigma(\vec{\alpha}), \sigma(\vec{\beta})) \rrbracket$$

to be  $J(\llbracket \vec{\alpha} \vdash \sigma(\vec{\alpha}) : \mathbf{Type} \rrbracket)$ , so now all we need to do is reindex this object. Given types  $\Xi \vdash \vec{\tau}, \vec{\tau}' : \mathbf{Type}$ , we define

$$\llbracket \Xi \mid - \mid \vec{R} : \mathbf{Rel}(\vec{\tau}, \vec{\tau}') \vdash \sigma[\vec{R}] : \mathbf{Rel}(\sigma(\vec{\tau}), \sigma(\vec{\tau}')) \rrbracket$$

to be

$$\langle \llbracket \Xi \vdash \vec{\tau} \rrbracket, \llbracket \Xi \vdash \vec{\tau}' \rrbracket \rangle * \llbracket \vec{\alpha}; \vec{\beta} \mid - \mid \vec{R} : \mathbf{Rel}(\vec{\alpha}, \vec{\beta}) \vdash \sigma[\vec{R}] : \mathbf{Rel}(\sigma(\vec{\alpha}), \sigma(\vec{\beta})) \rrbracket.$$

Finally, given definable relations  $\Xi \mid \Gamma \mid \Theta \vdash \vec{\rho} : \mathbf{Rel}(\vec{\tau}, \vec{\tau}')$  we define

$$\begin{aligned}
 & \llbracket \Xi \mid \Gamma \mid \Theta \vdash \sigma[\vec{\rho}] : \mathbf{Rel}(\sigma(\vec{\tau}), \sigma(\vec{\tau}')) \rrbracket = \\
 & \llbracket \Xi \mid - \mid \vec{R} : \mathbf{Rel}(\vec{\tau}, \vec{\tau}') \vdash \sigma[\vec{R}] : \mathbf{Rel}(\sigma(\vec{\tau}), \sigma(\vec{\tau}')) \rrbracket \circ \llbracket \Xi \mid \Gamma \mid \Theta \vdash \vec{\rho} : \mathbf{Rel}(\vec{\tau}, \vec{\tau}') \rrbracket.
 \end{aligned}$$

### 3.1. Soundness

We have now completed showing how to interpret all constructions of the language of Abadi and Plotkin's logic in APL-structures. We consider an implication  $\Xi \mid \Gamma \mid \Theta \mid \phi_1, \dots, \phi_n \vdash \psi$  to hold in the model if

$$\bigwedge_i \llbracket \Xi \mid \Gamma \mid \Theta \vdash \phi_i \rrbracket \vdash \llbracket \Xi \mid \Gamma \mid \Theta \vdash \psi \rrbracket,$$

where  $\vdash$  above refers to the fibrewise ordering in **Prop**.

**Theorem 3.10 (Soundness).** In any APL-structure the interpretation defined above is sound with respect to the axioms and rules specified in Section 2.4, i.e., all axioms hold in the model, and for all rules, if the hypothesis holds in the model, then so does the

conclusion. In any pre-APL structure the interpretation of the part of the logic excluding the relational interpretation of terms is sound.

We will only prove the first part of Theorem 3.10, i.e., soundness for APL-structures. The proof of soundness for pre-APL structures is basically the same. For the proof we need the following lemmas:

**Lemma 3.11.** If  $\Xi \mid \Gamma \vdash t : \sigma$  then

$$\llbracket \Xi \mid \Gamma \mid \Theta \vdash \phi[t/x] \rrbracket = (I \langle id_{\llbracket \Xi \mid \Gamma \rrbracket}, \llbracket t \rrbracket \rangle \times id_{\llbracket \Xi \mid \Theta \rrbracket})^* \llbracket \Xi \mid \Gamma, x : \sigma \mid \Theta \vdash \phi \rrbracket$$

*Proof.* We will prove the statement of the lemma and the statement

$$\begin{aligned} & \llbracket \Xi \mid \Gamma \mid \Theta \vdash \rho[t/x] : \text{Rel}(\tau, \tau') \rrbracket = \\ & \llbracket \Xi \mid \Gamma, x : \sigma \mid \Theta \vdash \rho : \text{Rel}(\tau, \tau') \rrbracket \circ (I \langle id_{\llbracket \Xi \mid \Gamma \rrbracket}, \llbracket t \rrbracket \rangle \times id_{\llbracket \Xi \mid \Theta \rrbracket}), \end{aligned}$$

for all definable relations  $\rho$ , by simultaneous induction on the structure of  $\phi$  and  $\rho$ . We only do a few cases and leave the rest to the reader.

**Case**  $\rho = \sigma[\vec{\rho}']$ :

$$\begin{aligned} \llbracket \Xi \mid \Gamma \mid \Theta \vdash \sigma[\vec{\rho}'][t/x] \rrbracket &= \llbracket \Xi \mid \Gamma \mid \Theta \vdash \sigma[\vec{\rho}'][t/x] \rrbracket = \\ & \llbracket \Xi \mid - \mid \vec{R} \vdash \sigma[\vec{R}] \rrbracket \circ \llbracket \vec{\rho}'[t/x] \rrbracket \end{aligned}$$

Since by induction  $\llbracket \vec{\rho}'[t/x] \rrbracket = \llbracket \vec{\rho}' \rrbracket \circ (I \langle id_{\llbracket \Xi \mid \Gamma \rrbracket}, \llbracket t \rrbracket \rangle \times id_{\llbracket \Xi \mid \Theta \rrbracket})$ , we are done.

**Case**  $\rho = (y : \tau, z : \tau') . \phi$ :

$$\llbracket \Xi \mid \Gamma \mid \Theta \vdash \rho[t/x] \rrbracket = \Psi^{-1}(\llbracket \Xi \mid \Gamma, y : \tau, z : \tau' \mid \Theta \vdash \phi[t/x] \rrbracket),$$

which by induction is equal to

$$\Psi^{-1}(\langle \pi_{\llbracket \Gamma \rrbracket}, \llbracket t \rrbracket, \pi_{\llbracket y : \tau, z : \tau' \mid \Theta \rrbracket} \rangle^* \llbracket \Xi \mid \Gamma, x : \sigma, y : \tau, z : \tau' \mid \Theta \vdash \phi \rrbracket).$$

By naturality of  $\Psi$  this is equal to

$$\begin{aligned} \Psi^{-1}(\llbracket \Xi \mid \Gamma, x : \sigma, y : \tau, z : \tau' \mid \Theta \vdash \phi \rrbracket) \circ \langle \pi_{\llbracket \Gamma \rrbracket}, \llbracket t \rrbracket, \pi_{\llbracket \Theta \rrbracket} \rangle = \\ \llbracket \Xi \mid \Gamma, x : \sigma \mid \Theta \vdash \rho \rrbracket \circ \langle \pi_{\llbracket \Gamma \rrbracket}, \llbracket t \rrbracket, \pi_{\llbracket \Theta \rrbracket} \rangle \end{aligned}$$

as desired.

**Case**  $\phi = \rho(u, s)$

Using naturality of  $\Psi$  as before, one can prove that

$$\begin{aligned} & \llbracket \Xi \mid \Gamma, y : \tau, z : \tau' \mid \Theta \vdash \rho(y, z)[t/x] \rrbracket = \\ & (I \langle id_{\llbracket \Xi \mid \Gamma, y : \tau, z : \tau' \rrbracket}, \llbracket t \rrbracket \rangle \times id_{\llbracket \Xi \mid \Theta \rrbracket})^* \llbracket \Xi \mid \Gamma, y : \tau, z : \tau', x : \sigma \mid \Theta \vdash \rho(y, z) \rrbracket. \end{aligned}$$

The general case follows from the fact that in a  $\lambda_2$ -fibration

$$\llbracket \Xi \mid \Gamma \vdash u[t/x] \rrbracket = \llbracket \Xi \mid \Gamma \vdash u \rrbracket \circ \langle id, \llbracket \Xi \mid \Gamma \vdash t \rrbracket \rangle.$$

**Case**  $\phi = \forall \alpha : \text{Type}. \psi$ :

We need to show that

$$\begin{aligned} & \llbracket \Xi \mid \Gamma \mid \Theta \vdash \forall \alpha : \text{Type}. \psi[t/x] \rrbracket = \\ & (I \langle id_{\llbracket \Xi \mid \Gamma \rrbracket}, \llbracket t \rrbracket \rangle \times id_{\llbracket \Xi \mid \Theta \rrbracket})^* \llbracket \Xi \mid \Gamma, x : \sigma \mid \Theta \mid \forall \alpha : \text{Type}. \psi \rrbracket. \end{aligned}$$

Let  $\bar{\pi}$  denote the cartesian lift of the projection  $[[\Xi, \alpha]] \rightarrow [[\Xi]]$ . Then by induction we have that the left hand side of the equation is

$$\prod_{\bar{\pi}} (I \langle id_{\Gamma}, [t] \rangle \times id_{\Theta})^* [[\Xi, \alpha \mid \Gamma, x: \sigma \mid \Theta \vdash \psi]].$$

Consider the square

$$\begin{array}{ccc} [[\Xi, \alpha \mid \Gamma \mid \Theta]] & \xrightarrow{\bar{\pi}} & [[\Xi \mid \Gamma \mid \Theta]] \\ I \langle id_{\Gamma}, [t] \rangle \times id_{\Theta} \downarrow & & \downarrow I \langle id_{\Gamma}, [t] \rangle \times id_{\Theta} \\ [[\Xi, \alpha \mid \Gamma, x: \sigma \mid \Theta]] & \xrightarrow{\bar{\pi}} & [[\Xi \mid \Gamma, x: \sigma \mid \Theta]]. \end{array}$$

This square commutes since  $\bar{\pi}$  is a natural transformation from  $\pi^*$  to  $id$ , and it is a pullback by (Jacobs, 1999, Exercise 1.4.4). The Beck-Chevalley condition relative to this square gives the desired result.  $\square$

**Lemma 3.12.** If  $\Xi \mid \Gamma \mid \Theta \vdash \phi: \mathbf{Prop}$ , then

$$[[\Xi \mid \Gamma, x: \sigma \mid \Theta \vdash \phi]] = \pi^* [[\Xi \mid \Gamma \mid \Theta \vdash \phi]],$$

where  $\pi: [[\Xi \mid \Gamma, x: \sigma \mid \Theta]] \rightarrow [[\Xi \mid \Gamma \mid \Theta]]$  is the projection.

**Lemma 3.13.** If  $\Xi \vdash \sigma: \mathbf{Type}$  then

$$[[\Xi \mid \Gamma[\sigma/\alpha] \mid \Theta[\sigma/\alpha] \vdash \phi[\sigma/\alpha]]] = \overline{\langle id_{[[\Xi]]}, [[\sigma]] \rangle}^* [[\Xi, \alpha: \mathbf{Type} \mid \Gamma \mid \Theta \vdash \phi]],$$

where the vertical line in  $\overline{\langle id_{[[\Xi]]}, [[\sigma]] \rangle}$  denotes the cartesian lift.

*Proof.* Notice first that a corresponding reindexing lemma for interpretation of  $\lambda_2$  in  $\lambda_2$ -fibrations tells us that

$$\langle id_{[[\Xi]]}, [[\sigma]] \rangle^* [[\Xi, \alpha \mid \Gamma \mid \Theta]] = [[\Xi \mid \Gamma[\sigma/\alpha] \mid \Theta[\sigma/\alpha]]].$$

The rest of the proof is by induction over the structure of  $\phi$ , and since it resembles the proof of Lemma 3.11 closely we leave it to the reader.  $\square$

**Lemma 3.14.** If  $\Xi \mid \Gamma \mid \Theta \vdash \phi$  then

$$[[\Xi \mid \Gamma \mid \Theta \vdash \phi]] = \bar{\pi}_{\Xi, \alpha \rightarrow \Xi}^* [[\Xi, \alpha \mid \Gamma \mid \Theta \vdash \phi]]$$

*Proof.* The proof is almost the same as for Lemma 3.13.  $\square$

**Lemma 3.15.** If  $\Xi \mid \Gamma \mid \Theta \vdash \rho: \mathbf{Rel}(\tau, \tau')$  is a definable relation, then

$$[[\Xi \mid \Gamma \mid \Theta \vdash \phi[\rho/R]]] = \langle \langle id_{[[\Xi \mid \Gamma \mid \Theta]]}, [[\rho]] \rangle \rangle^* [[\Xi \mid \Gamma \mid \Theta, R: \mathbf{Rel}(\tau, \tau') \vdash \phi]]$$

*Proof.* The lemma should be proved simultaneously with the statement

$$\begin{aligned} & [[\Xi \mid \Gamma \mid \Theta \vdash \rho'[\rho/R]]] = \\ & [[\Xi \mid \Gamma \mid \Theta, R: \mathbf{Rel}(\tau, \tau') \vdash \rho']] \circ \langle \langle id_{[[\Xi \mid \Gamma \mid \Theta]]}, [[\rho]] \rangle \rangle \end{aligned}$$

for all definable relations  $\rho'$ , by structural induction on  $\phi$  and  $\rho'$ . We leave the proof to the reader, as it closely resembles the proof of (3.11).  $\square$

**Lemma 3.16.** If  $\Xi \mid \Gamma \mid \Theta \vdash \phi : \text{Prop}$ , then

$$\llbracket \Xi \mid \Gamma \mid \Theta, R : \text{Rel}(\sigma, \tau) \vdash \phi \rrbracket = \pi^* \llbracket \Xi \mid \Gamma \mid \Theta \vdash \phi \rrbracket,$$

where  $\pi : \llbracket \Xi \mid \Gamma \mid \Theta, R : \text{Rel}(\sigma, \tau) \rrbracket \rightarrow \llbracket \Xi \mid \Gamma \mid \Theta \rrbracket$  is the projection.

We are now ready to prove soundness.

*Proof of Theorem 3.10* The rules for quantification (1)- (6) follow directly from the fact that the interpretation of  $\forall$  and  $\exists$  are given by right, respectively left adjoints to weakening functors. The substitution rules (7) - (9) are sound by Lemmas 3.11, 3.13 and 3.15.

For the *substitution* axiom (10) we will only prove

$$\begin{aligned} & \llbracket \alpha, \beta \mid x, x' : \alpha, y : \beta \mid R : \text{Rel}(\alpha, \beta) \vdash x =_\alpha x' \rrbracket \leq \\ & \llbracket \alpha, \beta \mid x, x' : \alpha, y : \beta \mid R : \text{Rel}(\alpha, \beta) \vdash R(x, y) \supset R(x', y) \rrbracket. \end{aligned}$$

Once this is done, the rest of the proof amounts to doing the same thing in the second variable. We will for readability write simply  $\llbracket \alpha \rrbracket, \llbracket \beta \rrbracket, \llbracket R \rrbracket$  for  $\llbracket \alpha, \beta \vdash \alpha \rrbracket, \llbracket \alpha, \beta \vdash \beta \rrbracket, \llbracket \alpha, \beta \mid - \mid R : \text{Rel}(\alpha, \beta) \rrbracket$ .

If we let  $\pi_1, \pi_2, \pi_3, \pi_4$  denote the projections out of

$$\begin{aligned} & \llbracket \alpha, \beta \mid x, x' : \alpha, y : \beta \mid R : \text{Rel}(\alpha, \beta) \rrbracket = \\ & \llbracket \alpha, \beta \vdash \alpha \rrbracket^2 \times \llbracket \alpha, \beta \vdash \beta \rrbracket \times U(\llbracket \alpha, \beta \vdash \alpha \rrbracket, \llbracket \alpha, \beta \vdash \beta \rrbracket) \end{aligned}$$

we can formulate what we aim to prove as

$$\langle \pi_1, \pi_2 \rangle^* (\coprod_{\Delta_{\llbracket \alpha \rrbracket}} (\top)) \leq \langle \pi_1, \pi_3 \rangle^* \Psi(id_{\llbracket R \rrbracket}) \supset \langle \pi_2, \pi_3 \rangle^* \Psi(id_{\llbracket R \rrbracket}),$$

where  $\Delta$  denotes the diagonal map.

Using the Beck-Chevalley condition on the square

$$\begin{array}{ccc} \llbracket \alpha \rrbracket \times \llbracket \beta \rrbracket \times \llbracket R \rrbracket & \xrightarrow{\Delta_{\llbracket \alpha \rrbracket} \times id} & \llbracket \alpha \rrbracket^2 \times \llbracket \beta \rrbracket \times \llbracket R \rrbracket \\ \pi_1 \downarrow \lrcorner & & \downarrow \langle \pi_1, \pi_2 \rangle \\ \llbracket \alpha \rrbracket & \xrightarrow{\Delta_{\llbracket \alpha \rrbracket}} & \llbracket \alpha \rrbracket^2 \end{array}$$

we get

$$\langle \pi_1, \pi_2 \rangle^* (\coprod_{\Delta_{\llbracket \alpha \rrbracket}} (\top)) = \coprod_{\Delta_{\llbracket \alpha \rrbracket} \times id_{\llbracket \beta \rrbracket} \times \llbracket R \rrbracket}} (\top).$$

Now the result follows from using the adjunction and the fact that

$$\langle \pi_1, \pi_3 \rangle \circ (\Delta_{\llbracket \alpha \rrbracket} \times id_{\llbracket \beta \rrbracket} \times \llbracket R \rrbracket}) = \langle \pi_2, \pi_3 \rangle \circ (\Delta_{\llbracket \alpha \rrbracket} \times id_{\llbracket \beta \rrbracket} \times \llbracket R \rrbracket}).$$

External equality implies internal equality (11) since the model of  $\lambda_2$  included in the model is sound. Internal equality is clearly an equivalence relation.

The axioms concerning types as relations (12) - (15) follow from the fact that  $J$  is required to be a morphism of  $\lambda_2$  fibrations and that the  $\lambda_2$  structure in **Relations**  $\rightarrow$  **RelCtx** is given by the interpretation of products and quantification of relations. For

instance soundness of the (15) is proved as follows:

$$\begin{aligned} \llbracket \vec{\alpha}, \vec{\alpha}' \mid - \mid \vec{R} : \text{Rel}(\vec{\alpha}, \vec{\alpha}') \vdash (\prod \beta. \sigma)[\vec{R}] \rrbracket &= \\ J(\llbracket \vec{\alpha} \vdash \prod \beta. \sigma \rrbracket) &= \\ \llbracket \vec{\alpha}, \vec{\alpha}' \mid \vec{R} : \text{Rel}(\vec{\alpha}, \vec{\alpha}') \vdash (\forall \gamma, \gamma', S : \text{Rel}(\gamma, \gamma')). \sigma[\vec{R}, S] \rrbracket \end{aligned}$$

where the second equality holds since  $J$  preserves simple  $\Omega$ -products.

Finally, to prove soundness of rule (16), it suffices to prove soundness of

$$\Xi \mid \Gamma, x : \sigma, y : \tau \mid \Theta \mid \top \vdash ((x : \sigma, y : \tau). \phi)(x, y) \vDash \phi,$$

but

$$\begin{aligned} \llbracket \Xi \mid \Gamma, x : \sigma, y : \tau \mid \Theta \vdash ((x : \sigma, y : \tau). \phi)(x, y) \rrbracket &= \\ \Psi(\llbracket \Xi \mid \Gamma \mid \Theta \vdash (x : \sigma, y : \tau). \phi \rrbracket) &= \\ \Psi \circ \Psi^{-1}(\llbracket \Xi \mid \Gamma, x : \sigma, y : \tau \mid \Theta \vdash \phi \rrbracket) &= \llbracket \Xi \mid \Gamma, x : \sigma, y : \tau \mid \Theta \vdash \phi \rrbracket. \end{aligned}$$

□

### 3.2. Completeness

The Soundness Theorem (3.10) allows us to reason about APL-structures using Abadi & Plotkin's logic. The Completeness Theorem below states that any formula that holds in all APL-structures, is provable in the logic. This allows us to reason about the logic using the class of APL-structures. However, since the APL-structure below is constructed from the logic, this does not say much. Instead, one should view the Completeness Theorem as stating that the class of APL-structures is not too restrictive; it completely describes the logic.

**Theorem 3.17 (Completeness).** There exists an APL-structure with the property that any formula of Abadi & Plotkin's logic based on pure  $\lambda_2$  that holds in the structure may be proved in the logic.

*Proof.* We construct the APL-structure syntactically, giving the categories in question the same names as in the diagram of item 1 in Definition 3.3.

- The category **Kind** has sequences of the form  $\alpha_1 : \text{Type}, \dots, \alpha_n : \text{Type}$  as objects, where we identify these contexts up to renaming (in other words, we may think of objects as natural numbers). A morphism from  $\Xi$  into  $\alpha_1 : \text{Type}, \dots, \alpha_n : \text{Type}$  is a sequence of types  $(\sigma_1, \dots, \sigma_n)$  such that all  $\sigma_i$  are well-formed in context  $\Xi$ .
- Objects in the fibre of **Type** over  $\Xi$  are well-formed types in this context, where we identify types up to renaming of free type variables. Morphisms in this fibre from  $\sigma$  to  $\tau$  are equivalence classes of terms  $t$  such that  $\Xi \mid - \vdash t : \sigma \rightarrow \tau$  where we identify terms up to external equality. Reindexing with respect to morphisms in **Kind** is by substitution.
- The category **Ctx** has as objects in the fibre over  $\Xi$  well-formed contexts of Abadi & Plotkin's logic:  $\Xi \mid \Gamma \mid \Theta$ , where we again identify such contexts up to renaming of free type-variables. A vertical morphism from  $\Xi \mid \Gamma \mid \Theta$  to  $\Xi \mid \Gamma' \mid R_1 : \text{Rel}(\sigma_1, \tau_1), \dots, R_n : \text{Rel}(\sigma_n, \tau_n)$  is a pair, consisting of a morphism  $\Xi \mid \Gamma \rightarrow \Xi \mid \Gamma'$

in the sense of morphisms in **Type** and a sequence of definable relations  $(\rho_1, \dots, \rho_n)$  such that  $\Xi \mid \Gamma \mid \Theta \vdash \rho_i : \text{Rel}(\sigma_i, \tau_i)$ . We identify two such morphisms represented by the same type morphism and the definable relations  $(\rho_1, \dots, \rho_n)$  and  $(\rho'_1, \dots, \rho'_n)$  if, for each  $i$ ,  $\rho_i \equiv \rho'_i$  is provable in the logic. one. Reindexing is by substitution.

- The fibre of the category **Prop** over a context  $\Xi \mid \Gamma \mid \Theta$  has as objects formulas in that context, where we identify two formulas if they are provably equivalent. These are ordered by entailment in the logic. Reindexing is done by substitution, that is, reindexing with respect to lifts of morphisms from **Kind** is done by substitution in Kind-variables, whereas reindexing with respect to vertical maps in **Ctx** is by substitution in type variables and relational variables.

It is straightforward to verify that this structure satisfies item 1 of Definition 3.3. The only non-obvious thing to verify here is existence of products and coproducts in **Prop** with respect to vertical maps in **Ctx**.

Suppose  $(\vec{t}, \vec{\rho})$  represents a morphism from  $\Xi \mid \vec{x} : \vec{\sigma} \mid \vec{R}$  to  $\Xi \mid \vec{y} : \vec{\tau} \mid \vec{S}$ . Then we can define the product functor in **Prop** by:

$$\prod_{(\vec{t}, \vec{\rho})} (\Xi \mid \vec{x} : \vec{\sigma} \mid \vec{R} \vdash \phi(\vec{x}, \vec{R})) = \Xi \mid \vec{y} : \vec{\tau} \mid \vec{S} \vdash \forall \vec{x}. \forall \vec{R}. (\vec{t}\vec{x} = \vec{y} \wedge (\vec{\rho}(\vec{x}, \vec{R}) \equiv \vec{S}) \supset \phi(\vec{x}, \vec{R})).$$

We define coproduct as:

$$\coprod_{(\vec{t}, \vec{\rho})} (\Xi \mid \vec{x} : \vec{\sigma} \mid \vec{R} \vdash \phi(\vec{x}, \vec{R})) = \Xi \mid \vec{y} : \vec{\tau} \mid \vec{S} \vdash \exists \vec{x}. \exists \vec{R}. \vec{t}\vec{x} = \vec{y} \wedge \vec{\rho}(\vec{x}, \vec{R}) \equiv \vec{S} \wedge \phi(\vec{x}, \vec{R}).$$

The functor  $U$  of item 2 is defined as

$$U(\sigma, \tau) = R : \text{Rel}(\sigma, \tau)$$

and

$$U(t : \sigma \rightarrow \sigma', u : \tau \rightarrow \tau') = \Xi \mid R : \text{Rel}(\sigma', \tau') \vdash (x : \sigma, y : \tau). R(tx, uy)$$

The map  $\Psi$  maps a definable relation  $\Xi \mid \Gamma \mid \Theta \vdash \rho : \text{Rel}(\sigma, \tau)$  to the proposition  $\Xi \mid \Gamma, x : \sigma, y : \tau \mid \Theta \vdash \rho(x, y) : \text{Prop}$ , which is a bijection by Lemma 2.3.

We have defined a pre-APL-structure. The category **RelCtx** obtained from this pre-APL structure has as objects  $\vec{\alpha}, \vec{\beta} \mid \Gamma \mid \Theta$ . The fibre of **Relations** over an object  $\vec{\alpha}, \vec{\beta} \mid \Gamma \mid \Theta$  in **RelCtx** is:

**Objects**      Equivalence classes of definable relations

$$\vec{\alpha}, \vec{\beta} \mid \Gamma \mid \Theta \vdash \rho : \text{Rel}(\sigma(\vec{\alpha}), \tau(\vec{\beta})).$$

**Morphisms** A morphism from  $\rho : \text{Rel}(\sigma(\vec{\alpha}), \tau(\vec{\beta}))$  to  $\rho' : \text{Rel}(\sigma'(\vec{\alpha}), \tau'(\vec{\beta}))$  is a pair of morphisms  $t : \sigma \rightarrow \sigma', u : \tau \rightarrow \tau'$  such that it is provable that

$$\forall x : \sigma. \forall y : \tau. \rho(x, y) \supset \rho'(tx, uy).$$

In the reflexive graph of Lemma 3.7, the functor from **Kind** to **RelCtx** acts on objects as

$$\alpha_1, \dots, \alpha_n \mapsto \alpha_1, \dots, \alpha_n; \beta_1, \dots, \beta_n \mid R_1 : \text{Rel}(\alpha_1, \beta_1), \dots, R_n : \text{Rel}(\alpha_n, \beta_n)$$

and it takes a morphism  $\vec{\sigma} : \vec{\alpha} \rightarrow \vec{\alpha}'$  to the triple  $(\vec{\sigma}(\alpha), \vec{\sigma}(\beta), \vec{\sigma}[\vec{R}])$ . Notice that this defines a morphism since

$$\vec{\alpha}, \vec{\beta} \mid \vec{R} : \text{Rel}(\vec{\alpha}, \vec{\beta}) \vdash \sigma_i[\vec{R}] : \text{Rel}(\sigma_i(\vec{\alpha}), \sigma_i(\vec{\beta}))$$

This really defines the object part of the functor from **Type** to **Relations** since it must preserve  $\lambda 2$ -structure. So this functor takes a type  $\vec{\alpha} \vdash \sigma$  to

$$\vec{\alpha}; \vec{\beta} \mid \vec{R} : \text{Rel}(\vec{\alpha}, \vec{\beta}) \vdash \sigma[\vec{R}] : \text{Rel}(\sigma(\vec{\alpha}), \sigma(\vec{\beta})).$$

The functor maps a morphism  $\vec{\alpha} \mid x : \sigma \vdash t : \tau$  to the pair  $(\lambda x : \sigma. t, \lambda x : \sigma. t)$ . This defines a morphism in **Relations** since the Logical Relations Lemma (Plotkin and Abadi, 1993, Lemma 2) implies that

$$\vec{\alpha}; \vec{\beta} \mid \vec{R} : \text{Rel}(\vec{\alpha}, \vec{\beta}) \mid x : \sigma(\vec{\alpha}), y : \sigma(\vec{\beta}) \vdash \sigma[\vec{R}](x, y) \supset \tau[\vec{R}](t, t[\beta/\alpha][y/x]).$$

One may easily verify that the functors above define a reflexive graph of  $\lambda 2$ -fibrations.

Now, by definition, a formula holds in this APL-structure iff it is provable in Abadi & Plotkin's logic.  $\square$

**Remark 3.18.** The Completeness Theorem only states completeness for Abadi & Plotkin's logic based on the *pure*  $\lambda 2$ . The reason for this is that the proof uses the Logical Relations Lemma, which is proved in (Plotkin and Abadi, 1993) by structural induction on terms. In the case of general calculi, one must know that the Logical Relations Lemma holds for term-constants in the language to be able to prove completeness.

#### 4. Parametric APL-structures

Given an APL-structure, we may consider the internal logic of the model (to be defined precisely below), and formulate parametricity as a schema in this logic. For technical reasons we will define parametric APL-structures as APL-structures not only satisfying the parametricity schema, but also extensionality and very strong equality (A.7). For parametric APL-structures, we can derive consequences of parametricity using Abadi & Plotkin's logic, as in (Plotkin and Abadi, 1993). For many of these proofs extensionality is needed, and we need very strong equality to deduce from theorems in Abadi & Plotkin's logic to category theoretic theorems, as we will see in Section 5. This is the reason why we propose parametric APL-structures as a category-theoretic definition of parametricity.

The internal language of an APL-structure is simply Abadi & Plotkin's logic on the internal language of the  $\lambda 2$ -fibration (see (Jacobs, 1999)), with the ordering relation in a fibre of **Prop** defined as  $\phi \vdash \psi$  iff  $\llbracket \phi \rrbracket \vdash \llbracket \psi \rrbracket$  holds in the model. Using the internal language we may express properties of the APL-structure, and ask whether these properties hold in the logic.

**Definition 4.1.** The extensionality schemes in the internal language of an APL-structure are the schemes

$$- \mid - \mid - \vdash \forall \alpha, \beta : \mathbf{Type}. \forall t, u : \alpha \rightarrow \beta. (\forall x : \alpha. tx =_{\beta} ux) \supset t =_{\alpha \rightarrow \beta} u, \quad (17)$$

$$\exists \mid - \mid - \vdash \forall f, g : (\Pi \alpha : \mathbf{Type}. \sigma). (\forall \alpha : \mathbf{Type}. f\alpha =_{\sigma} g\alpha) \supset f =_{\Pi \alpha : \mathbf{Type}. \sigma} g, \quad (18)$$

where in (18)  $\sigma$  ranges over all types such that  $\Xi, \alpha \vdash \sigma : \text{Type}$ .

**Lemma 4.2.** For any APL-structure, very strong equality (Definition A.7) implies extensionality.

*Proof.* We can formulate extensionality equivalently as the rules

$$\frac{\Xi \mid \Gamma, x : \sigma \mid \Theta \vdash t =_{\tau} u}{\Xi \mid \Gamma \mid \Theta \vdash \lambda x : \sigma. t =_{\sigma \rightarrow \tau} \lambda x : \sigma. u}$$

$$\frac{\Xi, \alpha : \text{Type} \mid \Gamma \mid \Theta \vdash f =_{\sigma} g}{\Xi \mid \Gamma \mid \Theta \vdash \Lambda \alpha. \text{Type}. f =_{\Pi \alpha : \text{Type}. \sigma} \Lambda \alpha. \text{Type}. g}$$

If internal equality is the same as external equality then these rules hold by the rules for external equality in Figure 1.  $\square$

**Definition 4.3.** The schema

$$\forall \vec{\alpha} : \text{Type}. \forall u, v : \sigma. (u(\sigma[eq_{\vec{\alpha}}])v \text{ } \mathcal{D} \text{ } u =_{\sigma} v)$$

is called the *Identity Extension Schema*. Here  $\sigma$  ranges over all types such that  $\vec{\alpha} \vdash \sigma : \text{Type}$ .

**Definition 4.4.** A **parametric APL-structure** is an APL-structure with very strong equality – and hence extensionality – satisfying the Identity Extension Schema.

**Remark 4.5.** If we write out the interpretation of the Identity Extension Schema, we get a category-theoretical formulation of the notion of parametric APL-structure. It is an APL-structure with very strong equality, extensionality and in which for all types  $\vec{\alpha} \vdash \sigma : \text{Type}$ ,

$$(id_{[\vec{\alpha} \vdash \sigma]^2} \times \llbracket \vec{\alpha} \mid - \mid - \vdash eq_{\vec{\alpha}} \rrbracket) * J(\llbracket \vec{\alpha} \vdash \sigma \rrbracket) = \llbracket \vec{\alpha} \mid x : \sigma, y : \sigma \mid - \vdash x =_{\sigma} y \rrbracket.$$

**Definition 4.6.** For any type  $\beta$ ,  $\vec{\alpha} \vdash \sigma(\beta, \vec{\alpha})$  we can form the parametricity schema:

$$\forall \vec{\alpha} : \text{Type}. \forall u : (\prod \beta. \sigma). \forall \beta, \beta' : \text{Type}. \forall R : \text{Rel}(\beta, \beta'). (u \beta) \sigma[R, eq_{\vec{\alpha}}](u \beta')$$

in the empty context.

**Proposition 4.7.** The Identity Extension Schema implies the parametricity schema. Thus the parametricity schema holds in any parametric APL-structure.

*Proof.* Since

$$\vec{\alpha} \mid u : \prod \beta : \text{Type}. \sigma(\beta, \vec{\alpha}) \mid - \vdash u =_{\prod \beta : \text{Type}. \sigma} u$$

always holds in the model, by the Identity Extension Schema, we know that

$$\vec{\alpha} \mid u : \prod \beta : \text{Type}. \sigma(\beta, \vec{\alpha}) \mid - \vdash u(\prod \beta : \text{Type}. \sigma)[eq_{\vec{\alpha}}]u$$

holds, but by the Axiom (15) this means that

$$\vec{\alpha} \mid u : \prod \beta : \text{Type}. \sigma(\beta, \vec{\alpha}) \vdash \forall \beta, \beta' \forall R : \text{Rel}(\beta, \beta'). (u \beta) (\sigma[R, eq_{\vec{\alpha}}])(u \beta')$$

holds as desired.  $\square$



Without assuming parametricity we can prove the logical relations lemma:

**Lemma 4.8 (Logical Relations Lemma).** For any APL-structure the Logical Relations Schema

$$- \mid - \mid - \vdash t\sigma t$$

holds, where  $t$  ranges over all *closed* terms of closed type, i.e.,  $- \mid - \vdash t: \sigma$ .

*Proof.* The lemma is really just a restatement of the requirement that

$$J : \mathbf{Type} \rightarrow \mathbf{Relations}$$

is a functor. Let us write out the details.

A closed term  $t$  of closed type  $\sigma$  corresponds in the model to a map  $t: 1 \rightarrow \sigma$  in  $\mathbf{Type}_1$ , and by definition of the interpretation

$$\llbracket - \mid x: \sigma, y: \sigma \mid - \vdash x\sigma y \rrbracket = J(\sigma).$$

The fact that  $J$  is required to be a functor, means exactly that the pair  $(t, t)$  should define a map in  $\mathbf{Relations}$ , i.e., the formula

$$- \mid - \mid - \vdash \forall x, y: 1. x1y \supset t\sigma t$$

should hold in the model. Since the relational interpretation of  $1$  is simply the constantly true relation, we get the statement of the lemma.  $\square$

**Remark 4.9.** The Logical Relations Lemma suspiciously resembles the Identity Extension Schema. For a closed term of open type:  $\vec{\alpha} \mid - \vdash t: \sigma$ , the Logical Relations Lemma implies  $(\Lambda\alpha. t) \prod \vec{\alpha}. \sigma(\Lambda\alpha. t)$ , so that  $t\sigma[eq_{\vec{\alpha}}]t$ . However, since this only holds for *closed* terms  $t$ , we do not have the formula

$$\forall t: \sigma. t\sigma[eq_{\vec{\alpha}}]t,$$

which is the formula that we will need to prove consequences of parametricity.

## 5. Consequences of parametricity

As mentioned in the introduction to Section 4 we may use Abadi & Plotkin's logic to derive consequences of parametricity in parametric APL-structures. In this section we exemplify how to do so. Through our examples, it should become apparent how extensionality and very strong equality play important roles in the proofs of the consequences.

The proofs of the consequences are based on theorems about Abadi & Plotkin's logic stated in (Plotkin and Abadi, 1993). For completeness, we have written out proofs of these theorems, often inspired by (Hasegawa, 1994). What is new here, is just that we show how to conclude from the logic to the APL-structures.

### 5.1. Dinaturality

We shall use the following definition very often.

**Definition 5.1.** We say that  $\vec{\alpha} \vdash \sigma$ : **Type** is an inductively constructed type, if it can be constructed from free variables  $\vec{\alpha}$  and closed types using the type constructors of  $\lambda_2$ , i.e.,  $\times$ ,  $\rightarrow$  and  $\prod \alpha$ .

For example, if  $\sigma$  is a closed type then  $\prod \alpha. \sigma \times \alpha$  is an inductively constructed type. However, some models may contain types that are not inductively constructed! For example, in syntactical models, any basic open type, such as the type  $\alpha \vdash \text{lists}(\alpha)$  is not inductively constructed.

We define the notion of positive and negative occurrences of a type variable  $\alpha$  in an inductively constructed type  $\sigma$  inductively over the structure of  $\sigma$  as follows. The type variable  $\alpha$  occurs positively in  $\alpha$ . The positive occurrences of  $\alpha$  in  $\sigma \times \tau$  are the positive occurrences of  $\alpha$  in  $\sigma$  and the positive occurrences of  $\alpha$  in  $\tau$ . Likewise for negative occurrences. The positive occurrences of  $\alpha$  in  $\sigma \rightarrow \tau$  are the positive occurrences of  $\alpha$  in  $\tau$  and the negative occurrences of  $\alpha$  in  $\sigma$ . The negative occurrences are the negative in  $\tau$  and the positive in  $\sigma$ . The positive and negative occurrences of  $\alpha$  in  $\prod \beta. \sigma$  are the same as for  $\sigma$ , if  $\alpha \neq \beta$ . There are no positive or negative occurrences of  $\alpha$  in  $\prod \alpha. \sigma$  since we only consider free occurrences of a type variable.

Suppose  $\sigma(\alpha, \beta)$  is an inductively constructed type with all free variables in  $\alpha, \beta$  such that  $\alpha$  occurs only negatively and  $\beta$  occurs only positively in  $\sigma$ . We may then for  $f: \alpha \rightarrow \alpha'$  and  $g: \beta \rightarrow \beta'$  define a morphism

$$\sigma(f, g) : \sigma(\alpha', \beta) \rightarrow \sigma(\alpha, \beta')$$

inductively over the structure of  $\sigma$  as in (Plotkin and Abadi, 1993).

It is well-known that Dinaturality is a consequence of parametricity, but we include the proof for completeness.

**Lemma 5.2 (Dinaturality).** In a parametric APL-structure, the dinaturality schema

$$\forall \alpha, \beta. \forall f: \alpha \rightarrow \beta. \sigma(\text{id}_\alpha, f) \circ (\cdot)_\alpha = \prod_{\alpha. (\sigma(\alpha, \alpha)) \rightarrow \sigma(\alpha, \beta)} \sigma(f, \text{id}_\beta) \circ (\cdot)_\beta$$

holds. Here  $(\cdot)_\alpha$  denotes the term  $\lambda u: (\prod \alpha. \sigma(\alpha, \alpha)). u(\alpha)$ .

*Proof.* Suppose  $f: \alpha \rightarrow \beta$ . By extensionality it suffices to prove that, for any  $u: \prod \alpha. \sigma(\alpha, \alpha)$ ,

$$\sigma(\text{id}_\alpha, f)u(\alpha) =_{\sigma(\alpha, \beta)} \sigma(f, \text{id}_\beta)u(\beta).$$

Instantiating the Logical Relations Lemma with the types

$$\begin{aligned} \alpha, \beta, \gamma, \delta \vdash (\alpha \rightarrow \beta) \times (\gamma \rightarrow \delta) \\ \alpha, \beta, \gamma, \delta \vdash \sigma(\beta, \gamma) \rightarrow \sigma(\alpha, \delta) \end{aligned}$$

and

$$\begin{aligned} t = \Lambda \alpha, \beta, \gamma, \delta. \lambda \omega: (\alpha \rightarrow \beta) \times (\gamma \rightarrow \delta). \sigma(\pi \omega, \pi' \omega) : \\ \prod \alpha, \beta, \gamma, \delta. (\alpha \rightarrow \beta) \times (\gamma \rightarrow \delta) \rightarrow \sigma(\beta, \gamma) \rightarrow \sigma(\alpha, \delta) \end{aligned}$$

we get

$$\begin{aligned} \alpha, \beta, \gamma, \delta, \alpha', \beta', \gamma', \delta' \mid x: (\alpha \rightarrow \beta) \times (\gamma \rightarrow \delta), y: (\alpha' \rightarrow \beta') \times (\gamma' \rightarrow \delta') \mid \\ R_1: \text{Rel}(\alpha, \alpha'), R_2: \text{Rel}(\beta, \beta'), R_3: \text{Rel}(\gamma, \gamma'), R_4: \text{Rel}(\delta, \delta') \mid \\ x(R_1 \rightarrow R_2) \times (R_3 \rightarrow R_4)y \vdash \sigma(\pi x, \pi' x)(\sigma[R_2, R_3] \rightarrow \sigma[R_1, R_4])\sigma(\pi y, \pi' y). \end{aligned}$$

Recall the notation  $\langle f \rangle$  for the graph of the function  $f$  defined as  $(x: \alpha, y: \beta). f(x) =_\beta y$ . If we set  $\alpha, \beta, \gamma, \alpha'$  to  $\alpha$  and set  $\delta, \beta', \gamma', \delta'$  to  $\beta$  and let  $R_1 = eq_\alpha$ ,  $R_2 = R_3 = \langle f \rangle$  and  $R_4 = eq_\beta$ , then we get

$$x(eq_\alpha \rightarrow \langle f \rangle) \times (\langle f \rangle \rightarrow eq_\beta)y \vdash \sigma(\pi x, \pi' x)(\sigma[\langle f \rangle, \langle f \rangle]) \rightarrow \sigma[eq_\alpha, eq_\beta]\sigma(\pi y, \pi' y).$$

If we set  $x = \langle id_\alpha, f \rangle$  and  $y = \langle f, id_\beta \rangle$  then since  $id_\alpha(eq_\alpha \rightarrow \langle f \rangle)f$  and  $f(\langle f \rangle \rightarrow eq_\beta)id_\beta$  we obtain

$$\sigma(id_\alpha, f)(\sigma[\langle f \rangle, \langle f \rangle]) \rightarrow \sigma[eq_\alpha, eq_\beta]\sigma(f, id_\beta).$$

Since the parametricity schema tells us that

$$u(\alpha)\sigma[\langle f \rangle, \langle f \rangle]u(\beta),$$

it follows that

$$\sigma(id_\alpha, f)(u(\alpha))(\sigma[eq_\alpha, eq_\beta])\sigma(f, id_\beta)u(\beta),$$

but by the Identity Extension Schema this is just

$$\sigma(id_\alpha, f)(u(\alpha)) =_{\sigma(\alpha, \beta)} \sigma(f, id_\beta)u(\beta).$$

□

## 5.2. Products

Consider the type  $T = \prod \alpha. \alpha \rightarrow \alpha$ . The term  $\Lambda \alpha. \lambda x: \alpha. x$  inhabits  $T$ . Thus

**Proposition 5.3.** In any model of  $\lambda_2$  the type  $T$  defines a fibred weak terminal object.

**Theorem 5.4.** In a parametric APL-structure, the proposition

$$\forall u: T. (u =_T \Lambda \alpha. \lambda x: \alpha. x)$$

holds in the internal logic.

*Proof.* By extensionality it suffices to prove that

$$\alpha: \mathbf{Type} \mid u: T, x: \alpha \vdash (u\alpha)x =_\alpha x.$$

Consider the relation

$$\alpha: \mathbf{Type} \mid u: T, x: \alpha \vdash \rho = (y: \alpha, z: \alpha). y =_\alpha x: \mathbf{Rel}(\alpha, \alpha).$$

By parametricity we have

$$\alpha: \mathbf{Type} \mid u: T, x: \alpha \vdash (u\alpha)(\rho \rightarrow \rho)(u\alpha),$$

but this means that

$$\alpha: \mathbf{Type} \mid u: T, x: \alpha \vdash y =_\alpha x \supset (u\alpha)y =_\alpha x.$$

□

**Theorem 5.5.** In a parametric APL-structure,  $T$  defines a fibred terminal object of  $\mathbf{Type} \rightarrow \mathbf{Kind}$ .

*Proof.* Suppose  $u: \sigma \rightarrow T$  is a morphism in the fibre. By the above theorem and extensionality,  $u$  is internally equal to  $\lambda y: \sigma. \Lambda \alpha. \lambda x: \alpha. x$ . By very strong equality we have external equality between  $u$  and  $\lambda y: \sigma. \Lambda \alpha. \lambda x: \alpha. x$ . So  $T$  is a terminal object.  $\square$

For two types  $\sigma$  and  $\tau$  in the same fibre, consider

$$\sigma \hat{\times} \tau = \prod \alpha. ((\sigma \rightarrow \tau \rightarrow \alpha) \rightarrow \alpha).$$

We use  $\hat{\times}$  to distinguish this definition from the usual fibrewise product denoted  $\times$ . We will show that  $\hat{\times}$  defines a weak product in the fibre, and that in parametric APL-structures it defines a genuine product.

Let projections  $\pi: \sigma \hat{\times} \tau \rightarrow \sigma$  and  $\pi': \sigma \hat{\times} \tau \rightarrow \tau$  be defined by

$$\begin{aligned} \pi x &= x \sigma (\lambda x: \sigma. \lambda y: \tau. x) \\ \pi' x &= x \tau (\lambda x: \sigma. \lambda y: \tau. y) \end{aligned}$$

and let  $pair: \sigma \rightarrow \tau \rightarrow \sigma \hat{\times} \tau$  be defined by

$$pair \ x \ y = \Lambda \alpha. \lambda f: \sigma \rightarrow \tau \rightarrow \alpha. f \ x \ y$$

If  $f: \alpha \rightarrow \sigma$  and  $g: \alpha \rightarrow \tau$ , we will write  $\langle f, g \rangle$  for  $\lambda x: \alpha. pair (f \ x) (g \ x)$ . Then

$$\pi \circ \langle f, g \rangle = \lambda x: \alpha. (pair (f \ x) (g \ x)) \sigma (\lambda x: \sigma. \lambda y: \tau. x) = \lambda x: \alpha. f \ x = f$$

and likewise

$$\pi' \circ \langle f, g \rangle = g$$

This proves:

**Proposition 5.6.** In any model of  $\lambda_2$  the construction  $\hat{\times}$  defines a fibrewise weak product.

**Theorem 5.7.** For any parametric APL-structure the proposition

$$\forall \sigma, \tau. \langle \pi, \pi' \rangle =_{\sigma \hat{\times} \tau} id_{\sigma \hat{\times} \tau}$$

holds in the internal logic.

*Proof.* For any  $f: \sigma \rightarrow \tau \rightarrow \alpha$  define  $f^*: \sigma \hat{\times} \tau \rightarrow \alpha$  as

$$f^* \ x = x \ \alpha \ f.$$

Suppose  $z: \sigma \hat{\times} \tau$ . By parametricity, for any relation  $R: \text{Rel}(\alpha, \beta)$ ,

$$(z \ \alpha)((eq_\sigma \rightarrow eq_\tau \rightarrow R) \rightarrow R)(z \ \beta).$$

Now, for any  $f: \sigma \rightarrow \tau \rightarrow \alpha$ ,

$$f^*(pair \ x \ y) = pair \ x \ y \ \alpha \ f = f \ x \ y,$$

i.e.,

$$pair(eq_\sigma \rightarrow eq_\tau \rightarrow \langle f^* \rangle) f,$$

which means that

$$(z \ \sigma \hat{\times} \tau \ pair) \langle f^* \rangle (z \ \alpha \ f).$$

In other words,

$$f^*(z \sigma \hat{\times} \tau \text{ pair}) =_{\alpha} z \alpha f.$$

Since the left hand side of this equation simply is

$$(z \sigma \hat{\times} \tau \text{ pair}) \alpha f,$$

we get by extensionality since  $\alpha, f$  were arbitrary,

$$z \sigma \hat{\times} \tau \text{ pair} =_{\sigma \hat{\times} \tau} z.$$

Suppose now that we are given  $f: \sigma \rightarrow \tau \rightarrow \alpha$ . We construct  $g: \sigma \hat{\times} \tau \rightarrow \alpha$  by

$$g z = f (\pi z) (\pi' z)$$

Then  $\text{pair}(eq_{\sigma} \rightarrow eq_{\tau} \rightarrow \langle g \rangle) f$  since

$$g (\text{pair } x y) = f (\pi \circ \text{pair } x y) (\pi' \circ \text{pair } x y) = f x y$$

Parametricity now states that for any  $z: \sigma \hat{\times} \tau$

$$(z \sigma \hat{\times} \tau)((eq_{\sigma} \rightarrow eq_{\tau} \rightarrow \langle g \rangle) \rightarrow \langle g \rangle)(z \alpha).$$

Thus  $(z \sigma \hat{\times} \tau \text{ pair}) \langle g \rangle (z \alpha f)$  and since  $(z \sigma \hat{\times} \tau \text{ pair}) =_{\sigma \hat{\times} \tau} z$  we have

$$f (\pi z) (\pi' z) = g z =_{\alpha} z \alpha f.$$

By extensionality

$$\lambda z: \sigma \hat{\times} \tau. \Lambda \alpha. \lambda f: \sigma \rightarrow \tau \rightarrow \alpha. f (\pi z) (\pi' z) =_{\sigma \hat{\times} \tau \rightarrow \sigma \hat{\times} \tau}$$

$$\lambda z: \sigma \hat{\times} \tau. \Lambda \alpha. \lambda f: \sigma \rightarrow \tau \rightarrow \alpha. z \alpha f = id_{\sigma \hat{\times} \tau}.$$

But the left hand side of this equation is just  $\langle \pi, \pi' \rangle$ .  $\square$

**Theorem 5.8.** In any parametric APL-structure,  $\hat{\times}$  defines a fibrewise product in **Type**  $\rightarrow$  **Kind**.

*Proof.* Since clearly  $\langle \pi \circ f, \pi' \circ f \rangle = \langle \pi, \pi' \rangle \circ f$  any map into  $\sigma \hat{\times} \tau$  is uniquely determined by its composition with  $\pi$  and  $\pi'$  by Theorem 5.7 and very strong equality.  $\square$

### 5.3. Coproducts

For the empty sum we define

$$I = \coprod \alpha. \alpha.$$

**Proposition 5.9.** In any model of  $\lambda_2$ ,  $I$  defines a fibred weak initial object.

*Proof.* Suppose  $\sigma$  is a type over some **Kind** object  $\Xi$ . The interpretation of the term  $x: I \vdash x\sigma$  is a morphism from  $I$  to  $\sigma$  in the fibre over  $\Xi$ .  $\square$

**Theorem 5.10.** In a parametric APL-structure, the proposition

$$\forall u: I. \perp$$

holds in the internal logic of the model.

*Proof.* Parametricity says

$$\forall u: \prod \alpha. \alpha. \forall \alpha, \beta: \mathbf{Type}. \forall R: \mathbf{Rel}(\alpha, \beta). u(\alpha) R u(\beta)$$

Instantiate this with the definable relation

$$(x: 1, y: 1). \perp : \mathbf{Rel}(1, 1)$$

□

**Theorem 5.11.** In a parametric APL-structure,  $I$  defines a fibred initial object of  $\mathbf{Type} \rightarrow \mathbf{Kind}$ .

*Proof.* Given two morphisms  $u, v: I \rightarrow \sigma$  we have

$$(\forall x: I. \perp) \vdash (\forall x: I. ux =_{\sigma} vx) \vdash (u =_{I \rightarrow \sigma} v),$$

so, by very strong equality, we have  $u = v$ . □

Given two types  $\sigma$  and  $\tau$  we define

$$\sigma + \tau = \prod \alpha. (\sigma \rightarrow \alpha) \rightarrow (\tau \rightarrow \alpha) \rightarrow \alpha$$

and introduce combinators  $inl_{\sigma, \tau}: \sigma \rightarrow \sigma + \tau$ ,  $inr_{\sigma, \tau}: \tau \rightarrow \sigma + \tau$  and

$$cases_{\sigma, \tau}: \prod \alpha. ((\sigma \rightarrow \alpha) \rightarrow (\tau \rightarrow \alpha) \rightarrow (\sigma + \tau) \rightarrow \alpha)$$

by

$$\begin{aligned} inl_{\sigma + \tau}(a) &= \Lambda \alpha. \lambda f: \sigma \rightarrow \alpha. \lambda g: \tau \rightarrow \alpha. f(a), \\ inr_{\sigma + \tau}(a) &= \Lambda \alpha. \lambda f: \sigma \rightarrow \alpha. \lambda g: \tau \rightarrow \alpha. g(a), \\ cases_{\sigma + \tau} \alpha f g \omega &= \omega \alpha f g. \end{aligned}$$

Now, suppose we are given two morphisms  $t: \sigma \rightarrow \alpha$  and  $u: \tau \rightarrow \alpha$ . Then we may define  $[u, t] = cases_{\sigma, \tau} \alpha t u: \sigma + \tau \rightarrow \alpha$  and we then have

$$[u, t] \circ inl_{\sigma, \tau}(x) = inl_{\sigma, \tau} x \alpha t u = t(x)$$

and likewise

$$[u, t] \circ inr_{\sigma, \tau}(y) = inr_{\sigma, \tau} y \alpha t u = u(y)$$

so we have proved the following proposition.

**Proposition 5.12.** For any model of  $\lambda_2$ , the operation  $+$  defines a fibred weak coproduct.

We will prove that in a parametric APL-structure,  $\sigma + \tau$  is in fact a coproduct.

**Theorem 5.13.** In a parametric APL-structure, the proposition

$$\forall \alpha, \sigma, \tau: \mathbf{Type}. \forall h: \sigma + \tau \rightarrow \alpha. h =_{\sigma + \tau \rightarrow \alpha} [h \circ inl_{\sigma + \tau}, h \circ inr_{\sigma + \tau}]$$

holds.

*Proof.* We will first prove that

$$[inl_{\sigma + \tau}, inr_{\sigma + \tau}] =_{\sigma + \tau} id_{\sigma + \tau}.$$

Instantiating the parametricity schema for  $\omega : \sigma + \tau$  with the relation  $\langle f \rangle$  we get that, for any  $f : \alpha \rightarrow \beta$  and all  $a : \sigma \rightarrow \alpha$  and  $b : \tau \rightarrow \alpha$ ,

$$f(\omega \alpha a b) =_{\beta} \omega \beta (f \circ a) (f \circ b).$$

Now consider any  $a' : \sigma \rightarrow \alpha$  and  $b' : \tau \rightarrow \alpha$  and set  $f : \sigma + \tau \rightarrow \alpha$  to

$$f(u) = u \alpha a' b'.$$

If we set  $a$  above to  $inl$  and  $b$  to  $inr$  we get

$$(\omega(\sigma + \tau) inl inr) \alpha a' b' =_{\beta} \omega \alpha (f \circ inl) (f \circ inr). \quad (19)$$

Since

$$f \circ inl(x) = inl(x) \alpha a' b' = a'(x),$$

for all  $x : \sigma$ , and likewise  $f \circ inr(y) = b'(y)$ , for  $y : \tau$ , (19) reduces to

$$(\omega(\sigma + \tau) inl inr) \alpha a' b' =_{\beta} \omega \alpha a' b'.$$

By extensionality this implies

$$(\omega(\sigma + \tau) inl inr) =_{\sigma + \tau} \omega,$$

and using extensionality again we obtain

$$[inl_{\sigma + \tau}, inr_{\sigma + \tau}] =_{\sigma + \tau \rightarrow \sigma + \tau} id_{\sigma + \tau}. \quad (20)$$

Finally, by the parametricity condition on *cases*, we have for any  $h : \sigma + \tau \rightarrow \alpha$  that

$$h(\text{cases}(\sigma + \tau) inl inr \omega) =_{\alpha} \text{cases} \alpha (h \circ inl) (h \circ inr) \omega,$$

so by extensionality and (20),

$$h =_{\sigma + \tau \rightarrow \alpha} [h \circ inl, h \circ inr].$$

□

**Theorem 5.14.** In any parametric APL-structure,  $+$  defines a fibred coproduct of **Type**  $\rightarrow$  **Kind**.

*Proof.* Using very strong equality, Theorem 5.13 tells us that maps out of  $\sigma + \tau$  are uniquely determined by their compositions with  $inl$  and  $inr$ . □

#### 5.4. Initial algebras

**Definition 5.15.** Consider a fibred functor

$$\begin{array}{ccc} \mathbb{E} & \xrightarrow{T} & \mathbb{E} \\ & \searrow & \swarrow \\ & \mathbb{B} & \end{array}$$

An indexed family of initial algebras for the functor  $T$  is a family

$$(in_{\Xi} : T(\sigma_{\Xi}) \rightarrow \sigma_{\Xi})_{\Xi \in \text{Obj } \mathbb{B}}$$

such that each  $in_{\Xi}$  is an initial algebra for the restriction of  $T$  to the fibre over  $\Xi$  and the family is closed under reindexing. If each  $in_{\Xi}$  is only a weak initial algebra we call it a family of weak initial algebras.

Suppose  $\alpha \vdash \sigma : \mathbf{Type}$  is an inductively constructed type (see Definition 5.1) in which  $\alpha$  occurs only positively. Then  $\sigma(\alpha)$  can be considered a functor in each fibre (Plotkin and Abadi, 1993). Actually, in (Plotkin and Abadi, 1993) Abadi & Plotkin construct a term

$$t : \prod \alpha, \beta : \mathbf{Type}. (\alpha \rightarrow \beta) \rightarrow \sigma(\alpha) \rightarrow \sigma(\beta),$$

which internalizes the morphism part of the functor  $\sigma$ . This term is an example of what we later on call a *polymorphic strength* (see Definition 5.29).

The type  $\sigma$  induces a fibred functor

$$\begin{array}{ccc} \mathbf{Type} & \xrightarrow{\quad} & \mathbf{Type} \\ & \searrow & \swarrow \\ & \mathbf{Kind} & \end{array}$$

mapping  $\Xi \vdash \tau$  to  $\Xi \vdash \sigma(\tau)$ . In this section we study families of initial algebras for such functors.

First we prove the graph lemma:

**Lemma 5.16.** If  $\alpha \vdash \sigma$  is an inductively constructed type in a parametric APL-structure in which  $\alpha$  occurs only positively, interpreted as a fibred functor as in (Plotkin and Abadi, 1993), then the formula

$$\forall \alpha, \beta : \mathbf{Type}. \forall f : \alpha \rightarrow \beta. \sigma[\langle f \rangle] \equiv \langle \sigma(f) \rangle$$

holds in the internal language of the model, where, as usual,  $\rho \equiv \rho'$  is short for

$$\forall x, y. \rho(x, y) \text{ } \vDash \text{ } \rho'(x, y).$$

*Proof.* Since the polymorphic strength  $t$  mentioned above is parametric, we have, for any pair of relations  $\rho : \mathbf{Rel}(\alpha, \alpha')$  and  $\rho' : \mathbf{Rel}(\beta, \beta')$ ,

$$t \alpha \beta ((\rho \rightarrow \rho') \rightarrow (\sigma[\rho] \rightarrow \sigma[\rho'])) t \alpha' \beta'. \quad (21)$$

If we instantiate this with  $\rho = eq_{\alpha}$ ,  $\rho' = \langle f \rangle$  for some map  $f : \alpha \rightarrow \beta$ , we get

$$t \alpha \alpha ((eq_{\alpha} \rightarrow \langle f \rangle) \rightarrow (eq_{\sigma(\alpha)} \rightarrow \sigma[\langle f \rangle])) t \alpha \beta,$$

using the Identity Extension Schema. Since  $id_{\alpha}(eq_{\alpha} \rightarrow \langle f \rangle)f$ , and since  $t \alpha \beta f = \sigma(f)$  and  $t \alpha \alpha id_{\alpha} = \sigma(id_{\alpha}) = id_{\sigma(\alpha)}$  we get

$$id_{\sigma(\alpha)}(eq_{\sigma(\alpha)} \rightarrow \sigma[\langle f \rangle])\sigma(f),$$

that is,

$$\forall x : \sigma(\alpha). x(\sigma[\langle f \rangle])\sigma(f)x.$$

Thus we have proved  $\langle \sigma(f) \rangle$  implies  $\sigma[\langle f \rangle]$ .

To prove the other direction, instantiate (21) with the relations  $\rho = \langle f \rangle$  and  $\rho' = eq_{\beta}$



for  $f: \alpha \rightarrow \beta$ . Since  $f(\langle f \rangle \rightarrow eq_\beta) id_\beta$ ,

$$\sigma(f)(\sigma[\langle f \rangle]) \rightarrow eq_{\sigma(\beta)} id_{\sigma(\beta)}.$$

So for any  $x: \sigma(\alpha)$  and  $y: \sigma(\beta)$  we have  $x(\sigma[\langle f \rangle])y$  implies  $\sigma(f)x = y$ . In other words,  $\sigma[\langle f \rangle]$  implies  $\langle \sigma(f) \rangle$ .  $\square$

We shall now define a family of initial algebras for the functor induced by  $\sigma$ . In each fibre  $\mathbf{Type}_\Xi$  we may define the type

$$\mu\alpha. \sigma(\alpha) = \prod \alpha. ((\sigma(\alpha) \rightarrow \alpha) \rightarrow \alpha)$$

with combinators

$$fold: \prod \alpha. ((\sigma(\alpha) \rightarrow \alpha) \rightarrow \mu\beta. \sigma(\beta) \rightarrow \alpha)$$

and

$$in: \sigma(\mu\alpha. \sigma(\alpha)) \rightarrow \mu\alpha. \sigma(\alpha)$$

given by

$$fold \alpha f z = z \alpha f$$

and

$$in z = \Lambda\alpha. \lambda f: \sigma(\alpha) \rightarrow \alpha. f(\sigma(fold \alpha f) z).$$

**Theorem 5.17.** In any model of second-order  $\lambda$ -calculus the family

$$(\Xi \vdash in: \sigma(\mu\alpha. \sigma(\alpha)) \rightarrow \mu\alpha. \sigma(\alpha))_\Xi$$

is a family of weak initial algebras for  $\sigma$ .

*Proof.* Given any algebra  $f: \sigma(\alpha) \rightarrow \alpha$  in any fibre, the diagram

$$\begin{array}{ccc} \sigma(\mu\alpha. \sigma(\alpha)) & \xrightarrow{in} & \mu\alpha. \sigma(\alpha) \\ \sigma(fold \alpha f) \downarrow & & \downarrow fold \alpha f \\ \sigma(\alpha) & \xrightarrow{f} & \alpha \end{array}$$

is commutative since

$$(fold \alpha f) \circ in z = in z \alpha f = f(\sigma(fold \alpha f) z)$$

and

$$f \circ \sigma(fold \alpha f) z = f(\sigma(fold \alpha f) z).$$

$\square$

We will show that in a parametric APL-structure,  $(\Xi \vdash in)_\Xi$  actually is a family of initial algebras. First we prove a lemma.

**Lemma 5.18.** In a parametric APL-structure, the formula

$$fold \mu\alpha. \sigma(\alpha) in =_{\mu\alpha. \sigma(\alpha) \rightarrow \mu\alpha. \sigma(\alpha)} id_{\mu\alpha. \sigma(\alpha)}$$

holds in the internal logic.

*Proof.* Consider an arbitrary element  $\omega: \mu\alpha.\sigma(\alpha)$  and a map  $f: \alpha \rightarrow \beta$ . The parametricity condition then gives

$$(\omega \alpha)((\sigma[\langle f \rangle] \rightarrow \langle f \rangle) \rightarrow \langle f \rangle)(\omega \beta).$$

Since Lemma 5.16 tells us that  $\sigma[\langle f \rangle] \equiv \langle \sigma(f) \rangle$ , this means that, if  $a: \sigma(\alpha) \rightarrow \alpha$  and  $b: \sigma(\beta) \rightarrow \beta$  have the property that

$$\forall x: \sigma(\alpha). f(a x) =_{\beta} b(\sigma(f) x)$$

(that is, if  $f$  is a morphism of algebras), then

$$f(\omega \alpha a) =_{\beta} \omega \beta b.$$

Consider now an arbitrary algebra  $k: \sigma(\alpha) \rightarrow \alpha$  and instantiate the above with the algebra morphism  $fold \alpha k$  from  $in$  to  $k$ , to get

$$fold \alpha k(\omega \mu\alpha.\sigma(\alpha) in) =_{\alpha} \omega \alpha k.$$

Since the left hand side of this equation is  $(\omega \mu\alpha.\sigma(\alpha) in) \alpha k$ , we get by extensionality that

$$\omega \mu\alpha.\sigma(\alpha) in =_{\mu\alpha.\sigma(\alpha)} \omega$$

and therefore, using extensionality again,

$$fold \mu\alpha.\sigma(\alpha) in =_{\mu\alpha.\sigma(\alpha) \rightarrow \mu\alpha.\sigma(\alpha)} id_{\mu\alpha.\sigma(\alpha)},$$

as required.  $\square$

**Theorem 5.19.** Suppose  $g: \mu\alpha.\sigma(\alpha) \rightarrow \alpha$  induces a map between algebras from  $in$  to  $f: \sigma(\alpha) \rightarrow \alpha$  in a parametric APL-structure. Then

$$g =_{\mu\alpha.\sigma(\alpha) \rightarrow \alpha} fold \alpha f$$

holds in the internal logic.

*Proof.* Since  $g$  is a map of algebras, the parametricity condition on an arbitrary  $\omega: \mu\alpha.\sigma(\alpha)$  entails as in the proof of Lemma 5.18 that

$$g(\omega \mu\alpha.\sigma(\alpha) in) =_{\alpha} \omega \alpha f$$

and therefore the result follows from extensionality since, by Lemma 5.18,

$$\omega \mu\alpha.\sigma(\alpha) in = (fold \mu\alpha.\sigma(\alpha) in) \omega =_{\mu\alpha.\sigma(\alpha)} \omega$$

and, moreover,

$$\omega \alpha f = (fold \alpha f) \omega.$$

$\square$

**Theorem 5.20.** In a parametric APL-structure,  $(\Xi \vdash in)_{\Xi}$  is a family of initial algebras for  $\sigma$ .

*Proof.* Using very strong equality Thm 5.19 gives uniqueness of algebra morphisms out of  $in$ .  $\square$

**Remark 5.21.** Consider the case of an inductively constructed type  $\alpha, \beta \vdash \sigma(\alpha, \beta)$  in which  $\alpha$  and  $\beta$  occur only positively. For each closed type  $\tau$  we may consider the type  $\alpha \vdash \sigma(\alpha, \tau)$  and the analysis above gives us a family of initial algebras for this functor. Moreover, for each morphism  $f : \tau \rightarrow \tau'$  between closed types we get a morphism of algebras induced by initiality:

$$\begin{array}{ccc}
 \sigma(\mu\alpha. \sigma(\alpha, \tau), \tau) & \dashrightarrow & \sigma(\mu\alpha. \sigma(\alpha, \tau'), \tau) \\
 \downarrow \text{in}_\tau & & \downarrow \sigma(id, f) \\
 & & \sigma(\mu\alpha. \sigma(\alpha, \tau'), \tau') \\
 & & \downarrow \text{in}_{\tau'} \\
 \mu\alpha. \sigma(\alpha, \tau) & \dashrightarrow & \mu\alpha. \sigma(\alpha, \tau').
 \end{array}$$

For example, if we consider the type  $\alpha, \beta \vdash 1 + \alpha \times \beta$ , then for any  $\tau$ , we get  $lists(\tau) = \mu\alpha. (1 + \alpha \times \tau)$  and, for any  $f : \tau \rightarrow \tau'$ , the induced morphism is the familiar morphism  $map f : lists(\tau) \rightarrow lists(\tau')$ , which applies  $f$  to each element in a list.

### 5.5. Final coalgebras

In this section we consider the same setup as in Section 5.4, that is,  $\alpha \vdash \sigma : \mathbf{Type}$  is an inductively constructed type in which  $\alpha$  occurs only positively. As before  $\sigma$  defines a fibred endofunctor on  $\mathbf{Type} \rightarrow \mathbf{Kind}$ .

**Definition 5.22.** Consider a fibred functor

$$\begin{array}{ccc}
 \mathbb{E} & \xrightarrow{T} & \mathbb{E} \\
 & \searrow & \swarrow \\
 & \mathbb{B} &
 \end{array}$$

An indexed family of final coalgebras for the functor  $T$  is a family

$$(out_{\Xi} : \sigma_{\Xi} \rightarrow T(\sigma_{\Xi}))_{\Xi \in \text{Obj } \mathbb{B}}$$

such that each  $out_{\Xi}$  is a final coalgebra for the restriction of  $T$  to the fibre over  $\Xi$  and the family is closed under reindexing. If each  $out_{\Xi}$  is only a weak final coalgebra we call it a family of weak final coalgebras.

In this section we define a family of weak final coalgebras for  $\sigma$  and prove that for parametric APL-structures it is in fact a family of final coalgebras. First we need to define existential quantification in each fibre as

$$\coprod \alpha. \sigma(\alpha) = \prod \alpha. (\prod \beta. (\sigma(\beta) \rightarrow \alpha)) \rightarrow \alpha$$

and the combinator  $pack : \prod \alpha. (\sigma(\alpha) \rightarrow \coprod \beta. \sigma(\beta))$  by

$$pack \alpha x = \Lambda \beta \lambda f : \prod \alpha. (\sigma(\alpha) \rightarrow \beta). f \alpha x.$$

In each fibre we define the type

$$\nu\alpha. \sigma(\alpha) = \coprod \alpha. ((\alpha \rightarrow \sigma(\alpha)) \times \alpha) = \coprod \alpha. (\coprod \beta. (\beta \rightarrow \sigma(\beta)) \times \beta \rightarrow \alpha) \rightarrow \alpha$$

with combinators

$$\text{unfold}: \coprod \alpha. ((\alpha \rightarrow \sigma(\alpha)) \rightarrow \alpha \rightarrow (\nu\alpha. \sigma(\alpha)))$$

and

$$\text{out}: \nu\alpha. \sigma(\alpha) \rightarrow \sigma(\nu\alpha. \sigma(\alpha))$$

defined as

$$\text{unfold } \alpha \ f \ x = \text{pack } \alpha \ \langle f, x \rangle$$

and

$$\text{out}(x) = x \ \sigma(\nu\alpha. \sigma(\alpha)) \ (\Lambda \alpha \lambda \langle f, x \rangle : ((\alpha \rightarrow \sigma(\alpha)) \times \alpha). \sigma(\text{unfold } \alpha \ f)(f \ x)).$$

**Theorem 5.23.** In any model of second-order  $\lambda$ -calculus  $(\Xi \vdash \text{out})_{\Xi}$  is a family of weak final coalgebras for  $\sigma$ .

*Proof.* Consider a coalgebra  $f: \alpha \rightarrow \sigma(\alpha)$  in any fibre. Then

$$\begin{array}{ccc} \alpha & \xrightarrow{f} & \sigma(\alpha) \\ \text{unfold } \alpha \ f \ \downarrow & & \downarrow \sigma(\text{unfold } \alpha \ f) \\ \nu\alpha. \sigma(\alpha) & \xrightarrow{\text{out}} & \sigma(\nu\alpha. \sigma(\alpha)) \end{array}$$

commutes since

$$\begin{aligned} \text{out}(\text{unfold } \alpha \ f \ z) &= \text{out}(\text{pack } \alpha \ \langle f, z \rangle) = \\ (\text{pack } \alpha \ \langle f, z \rangle) \ (\sigma(\nu\alpha. \sigma(\alpha))) \ (\Lambda \alpha \lambda \langle f, x \rangle : ((\alpha \rightarrow \sigma(\alpha)) \times \alpha). \sigma(\text{unfold } \alpha \ f)(f \ x)) &= \\ = \sigma(\text{unfold } \alpha \ f)(f \ z) \end{aligned}$$

□

**Lemma 5.24.** In a parametric APL-structure,

$$\text{unfold } \nu\alpha. \sigma(\alpha) \ \text{out}$$

is internally equal to the identity on  $\nu\alpha. \sigma(\alpha)$ .

*Proof.* Set  $h = \text{unfold } \nu\alpha. \sigma(\alpha) \ \text{out}$  in the following.

By parametricity, for any  $k: \alpha \rightarrow \beta$ ,

$$\text{unfold } \alpha \ (\langle k \rangle \rightarrow \sigma[\langle k \rangle]) \rightarrow (\langle k \rangle \rightarrow \text{eq}_{\nu\alpha. \sigma(\alpha)}) \text{unfold } \beta.$$

Hence, since  $\sigma[\langle k \rangle] \equiv \langle \sigma(k) \rangle$  by Lemma 5.16, if

$$k: (f: \alpha \rightarrow \sigma(\alpha)) \rightarrow (g: \beta \rightarrow \sigma(\beta))$$

is a morphism of coalgebras, then

$$\text{unfold } \alpha \ f =_{\alpha \rightarrow \nu\alpha. \sigma(\alpha)} (\text{unfold } \beta \ g) \circ k.$$

So since  $h$  is a morphism of coalgebras from  $out$  to  $out$  we have  $h = h^2$ . Intuitively, all we need to prove now is that  $h$  is “surjective”.

Consider any  $g : \prod \alpha. ((\alpha \rightarrow \sigma(\alpha)) \times \alpha \rightarrow \beta)$ . By parametricity and Lemma 5.16, for any coalgebra map  $k : (f : \alpha \rightarrow \sigma(\alpha)) \rightarrow (f' : \alpha' \rightarrow \sigma(\alpha'))$ , we must have

$$\forall x : \alpha. g \alpha \langle f, x \rangle =_{\beta} g \alpha' \langle f', k(x) \rangle.$$

Using this on the coalgebra map  $unfold \alpha f$  from  $f$  to  $out$  we obtain

$$\forall x : \alpha. g \alpha \langle f, x \rangle =_{\beta} g \nu \alpha. \sigma(\alpha) \langle out, unfold \alpha f x \rangle.$$

In other words, if we define

$$k : \prod \alpha. ((\alpha \rightarrow \sigma(\alpha)) \times \alpha \rightarrow \tau),$$

where  $\tau = (\nu \alpha. \sigma(\alpha) \rightarrow \sigma(\nu \alpha. \sigma(\alpha))) \times \nu \alpha. \sigma(\alpha)$ , to be

$$k = \Lambda \alpha. \lambda \langle f, x \rangle : (\alpha \rightarrow \sigma(\alpha)) \times \alpha. \langle out, unfold \alpha f x \rangle,$$

then

$$\forall \alpha. g \alpha =_{(\alpha \rightarrow \sigma(\alpha)) \times \alpha \rightarrow \beta} (g \nu \alpha. \sigma(\alpha)) \circ (k \alpha). \quad (22)$$

Now, suppose we are given  $\alpha, \alpha', R : \text{Rel}(\alpha, \alpha')$  and terms  $f, f'$  such that

$$f((R \rightarrow \sigma[R]) \times R \rightarrow \beta) f'.$$

Then, by (22) and parametricity of  $g$

$$g \alpha f =_{\beta} g \alpha' f' =_{\beta} (g \nu \alpha. \sigma(\alpha))(k \alpha' f'),$$

from which we conclude

$$g(\forall(\alpha, \beta, R : \text{Rel}(\alpha, \beta)). ((R \rightarrow \sigma[R]) \times R \rightarrow \langle g \nu \alpha. \sigma(\alpha) \rangle)) k.$$

This implies that for any  $x : \nu \alpha. \sigma(\alpha)$  by parametricity we have

$$x \beta g =_{\beta} g \nu \alpha. \sigma(\alpha) (x \tau k).$$

Thus, since  $g$  was arbitrary, we may apply the above to  $g = k$  and get

$$x \tau k =_{\tau} k \nu \alpha. \sigma(\alpha) (x \tau k) = \langle out, unfold \nu \alpha. \sigma(\alpha) \pi(x \tau k) \pi'(x \tau k) \rangle.$$

If we write

$$l = \lambda x : \nu \alpha. \sigma(\alpha). unfold \nu \alpha. \sigma(\alpha) \pi(x \tau k) \pi'(x \tau k),$$

then since  $k$  is a closed term, so is  $l$ , and from the above calculations we conclude that we have

$$\forall \beta. \forall g : \prod \alpha. (\alpha \rightarrow \sigma(\alpha)) \times \alpha \rightarrow \beta. x \beta g =_{\beta} g \nu \alpha. \sigma(\alpha) \langle out, l x \rangle.$$

Now, finally

$$\begin{aligned} h(l x) &= unfold \nu \alpha. \sigma(\alpha) out(l x) = \\ &= pack \nu \alpha. \sigma(\alpha) \langle out, l x \rangle = \\ \Lambda \beta. \lambda g : \prod \alpha. ((\alpha \rightarrow \sigma(\alpha)) \times \alpha \rightarrow \beta). g \nu \alpha. \sigma(\alpha) \langle out, l x \rangle &=_{\nu \alpha. \sigma(\alpha)} \\ \Lambda \beta. \lambda g : \prod \alpha. ((\alpha \rightarrow \sigma(\alpha)) \times \alpha \rightarrow \beta). x \beta g &= x \end{aligned}$$

where we have used extensionality. Thus  $l$  is a right inverse to  $h$ , and we conclude

$$h x =_{\nu\alpha.\sigma(\alpha)} h^2(l x) =_{\nu\alpha.\sigma(\alpha)} h(l x) =_{\nu\alpha.\sigma(\alpha)} x.$$

□

**Theorem 5.25.** In a parametric APL-structure,  $(\Xi \vdash out)_{\Xi}$  is a family of final coalgebras for  $\sigma$ .

*Proof.* Consider a map of coalgebras into  $out$ :

$$\begin{array}{ccc} \alpha & \xrightarrow{f} & \sigma(\alpha) \\ \downarrow g & & \downarrow \sigma(g) \\ \nu\alpha.\sigma(\alpha) & \xrightarrow{out} & \sigma(\nu\alpha.\sigma(\alpha)). \end{array}$$

By parametricity of  $unfold$  we have

$$unfold \alpha f =_{\alpha \rightarrow \nu\alpha.\sigma(\alpha)} (unfold \nu\alpha.\sigma(\alpha) out) \circ g =_{\alpha \rightarrow \nu\alpha.\sigma(\alpha)} g.$$

Very strong equality then implies uniqueness of coalgebra morphisms into  $out$  as desired. □

### 5.6. Generalizing to strong fibred functors

In this section, our aim is to generalize the results of Sections 5.4 and 5.5 to initial algebras and final coalgebras for a more general class of fibred functors, than the one defined by inductively constructed types. To be able to use the internal logic of the model, however, we need the fibred functor to be “internalized” in the internal logic.

Consider a fibred functor

$$\begin{array}{ccc} \mathbf{Type} & \xrightarrow{F} & \mathbf{Type} \\ & \searrow & \swarrow \\ & \mathbf{Kind} & \end{array}$$

Since  $\mathbf{Type} \rightarrow \mathbf{Kind}$  has a generic object  $T \in \mathbf{Type}_{\Omega}$  ((Jacobs, 1999, Definition 5.2.8)), there is for every  $I \in \mathbf{Kind}$  a map

$$\phi_I : \mathbf{Type}_I \rightarrow \mathbf{Hom}_{\mathbf{Kind}}(I, \Omega)$$

such that  $\phi_I(X)$  is the unique map, such that  $\phi_I(X)^*T$  is vertically isomorphic to  $X$ . Now,

$$F(X) \cong F(\phi_I(X)^*T) \cong \phi_I(X)^*F(T).$$

Thus we have proved the following lemma.

**Lemma 5.26.** Every fibred functor  $F: \mathbf{Type} \rightarrow \mathbf{Type}$  is naturally isomorphic to a functor, whose object part is defined as  $X \mapsto \phi_I(X)^*\sigma$  for some  $\sigma \in \mathbf{Type}_{\Omega}$ . In the internal language the object part of such a functor is written as  $\tau \mapsto \sigma(\tau)$ .

In the following, we shall assume that  $F$  has this form and simply denote  $F$  by  $\sigma$ .

Thus we can always represent the object part of a functor in the internal language. To represent the morphism part, we need to use strong functors.

**Definition 5.27.** An endofunctor  $T : B \rightarrow B$  on a cartesian closed category is called **strong** if there exists a natural transformation  $t_{\sigma, \tau} : \sigma^\sigma \rightarrow T\tau^{T\sigma}$  preserving identity and composition:

$$\begin{array}{ccc}
 1 & \xrightarrow{\widehat{id}_\sigma} & \sigma^\sigma \\
 & \searrow & \downarrow t_{\sigma, \sigma} \\
 & & T\sigma^{T\sigma} \\
 \sigma_2^{\sigma_1} \times \sigma_3^{\sigma_2} & \xrightarrow{comp} & \sigma_3^{\sigma_1} \\
 \downarrow t \times t & & \downarrow t \\
 T\sigma_2^{T\sigma_1} \times T\sigma_3^{T\sigma_2} & \xrightarrow{comp} & T\sigma_3^{T\sigma_1}
 \end{array}$$

The natural transformation  $t$  is called the **strength** of the functor  $T$ .

One should note that  $t$  in the definition above represents the morphism part of the functor  $T$  in the sense that it makes the diagram

$$\begin{array}{ccc}
 1 & \xrightarrow{\widehat{f}} & \tau^\sigma \\
 & \searrow & \downarrow t_{\sigma, \tau} \\
 & & T\tau^{T\sigma} \\
 & \widehat{Tf} & 
 \end{array}$$

commute, for any morphism  $f : \sigma \rightarrow \tau$ . This follows from the commutative diagram

$$\begin{array}{ccccc}
 1 & & & & \\
 & \searrow & \widehat{id} & & \\
 & & \sigma^\sigma & \xrightarrow{t} & T\sigma^{T\sigma} \\
 & \searrow & \downarrow f^\sigma & & \downarrow Tf^{T\sigma} \\
 & & \tau^\sigma & \xrightarrow{t} & T\tau^{T\sigma} \\
 & \widehat{f} & & & 
 \end{array}$$

**Definition 5.28.** A **strong fibred functor** is a fibred endofunctor

$$\begin{array}{ccc}
 \mathbb{E} & \xrightarrow{T} & \mathbb{E} \\
 & \searrow & \swarrow \\
 & & \mathbb{B}
 \end{array}$$

on a fibred ccc, for which there exists a fibred natural transformation  $t$  from the fibred functor  $(-)^{(+)}$  to  $T(-)^{T(+)}$  satisfying commutativity of the two diagrams of Definition 5.27 in each fibre. The natural transformation  $t$  is called the **strength** of the functor  $T$ .

In this definition, one should of course check that the two functors  $(-)^{(+)}$  and  $T(-)^{T(+)}$

— a priori only defined on the fibres — in fact define fibred functors

$$\begin{array}{ccc} \mathbf{Type} \times_{\mathbf{Kind}} \mathbf{Type} & \longrightarrow & \mathbf{Type} \\ & \searrow & \swarrow \\ & \mathbf{Kind} & \end{array}$$

But this is easily seen. Notice also that  $T$  is not required to preserve the fibred ccc-structure and that the components of  $t$  are preserved under reindexing since  $t$  is a fibred natural transformation.

**Definition 5.29.** A fibred endofunctor

$$\begin{array}{ccc} \mathbf{Type} & \xrightarrow{\sigma} & \mathbf{Type} \\ & \searrow & \swarrow \\ & \mathbf{Kind} & \end{array}$$

defined on objects by  $\tau \mapsto \sigma(\tau)$  for a type  $\sigma \in \mathbf{Type}_\Omega$  as above, is **polymorphically strong** if there exists a term

$$t: \prod \alpha, \beta: \mathbf{Type}. (\alpha \rightarrow \beta) \rightarrow \sigma(\alpha) \rightarrow \sigma(\beta)$$

such that the family  $(t \alpha \beta)_{(\alpha, \beta) \in \mathbf{Type} \times_{\mathbf{Kind}} \mathbf{Type}}$  is a strength of the functor  $\sigma$  in the sense of Definition 5.28. The term  $t$  is called the **polymorphic strength** of  $\sigma$ .

**Example 5.30.** An inductively constructed type with one free variable  $\alpha \vdash \sigma: \mathbf{Type}$ , where  $\alpha$  occurs only positively, defines a polymorphically strong fibred functor: see Section 5.4.

But in many situations one may want to reason about other polymorphically strong fibred functors than the ones defined by types modeled in  $\lambda_2$ . For example, if the  $\lambda_2$ -fibration of the APL-structure models other type constructions than the ones from  $\lambda_2$  for which there are natural functorial interpretations, one may want to prove existence of initial algebras for functors induced by types in this extended language.

For polymorphically strong fibred functors we can also reason about their morphism part in the internal language. For instance, we may write

$$\alpha, \beta \mid f: \alpha \rightarrow \beta \vdash t \alpha \beta f: \sigma(\alpha) \rightarrow \sigma(\beta)$$

to express the application of a functor  $\sigma$  with strength  $t$  to a morphism  $f$ .

Furthermore, since the morphism part of the functor is represented by a polymorphic term, we can use parametricity to reason about it. For instance, we may prove the following generalization of Lemma 5.16.

**Lemma 5.31 (Graph Lemma).** For any parametric APL-structure, if  $\sigma$  is a polymorphically strong fibred endofunctor  $\mathbf{Type} \rightarrow \mathbf{Type}$ , then the formula

$$\forall \alpha, \beta: \mathbf{Type}. \forall f: \alpha \rightarrow \beta. \sigma[\langle f \rangle] \equiv \langle \sigma(f) \rangle$$



holds in the internal language of the APL-structure, where  $\rho \equiv \rho'$  is short for

$$\forall x, y. \rho(x, y) \simeq \rho'(x, y).$$

The proof of this lemma is the same as the proof of Lemma 5.16.

**Corollary 5.32.** For any parametric APL-structure, the morphism part of a polymorphically strong fibred endofunctor  $\sigma$  is uniquely determined by the object part.

*Proof.* By Lemma 5.31,  $y = \sigma(f)(x)$  iff  $x\sigma[\langle f \rangle]y$ .  $\square$

**Theorem 5.33.** In a parametric APL-structure, any polymorphically strong fibred functor  $T : \mathbf{Type} \rightarrow \mathbf{Type}$  has

- A family of initial algebras defined as in Section 5.4
- A family of final coalgebras defined as in Section 5.5

*Proof.* The proofs work exactly as in Sections 5.4 and 5.5 since we may express the functor  $T$  in the internal language, as described above.

The fact that these initial algebras and final coalgebras are preserved by reindexing follows from the fact that the strengths  $t$  are preserved.  $\square$

## 6. Concrete APL-structures

In this section we define a concrete parametric APL-structure based on a well-known variant of the per-model (see, for instance, (Jacobs, 1999, Section 8.4)).

The diagram of Definition 3.3 in the concrete model is:

$$\begin{array}{ccc}
 & & \mathbf{UFam}(\mathbf{RegSub}(\mathbf{Asm})) \\
 & & \downarrow r \\
 \mathbf{PFam}(\mathbf{Per}) & \xrightarrow{I} & \mathbf{UFam}(\mathbf{Asm}) \\
 & \searrow p & \downarrow q \\
 & & \mathbf{PPer}
 \end{array} \tag{23}$$

The fibration  $p$  is the fibration of (Jacobs, 1999, Def. 8.4.9); we repeat the definition here. In the following,  $\mathbf{Per}$  and  $\mathbf{Asm}$ , will denote the sets of partial equivalence relations and assemblies respectively on the natural numbers (see (Jacobs, 1999)).

The category  $\mathbf{PPer}$  is defined as

**Objects** Natural numbers.

**Morphisms** A morphism  $f : n \rightarrow 1$  is a pair  $(f^p, f^r)$  where  $f^p : \mathbf{Per}^n \rightarrow \mathbf{Per}$  is any map and

$$f^r \in \prod_{\vec{R}, \vec{S} \in \mathbf{Per}^n} \left[ \prod_{i \leq n} P(\mathbb{N}/R_i \times \mathbb{N}/S_i) \rightarrow P(\mathbb{N}/f^p(\vec{R})) \times \mathbb{N}/f^p(\vec{S}) \right]$$

is a map that satisfies *the identity extension condition*:  $f^r(\vec{Eq}) = Eq$ . A morphism from  $n$  to  $m$  is an  $m$ -vector of morphism from  $n$  to 1.

We can now define  $\mathbf{PFam}(\mathbf{Per})$  as the indexed category with fibre over  $n$  defined as

**Objects** morphisms,  $n \rightarrow 1$  of  $\mathbf{PPer}$ .

**Morphisms** a morphism from  $f$  to  $g$  is an indexed family of maps  $(\alpha_{\vec{R}})_{\vec{R} \in \mathbf{Per}^n}$  where

$$\alpha_{\vec{R}} : \mathbb{N}/f^p(\vec{R}) \rightarrow \mathbb{N}/g^p(\vec{R})$$

are tracked uniformly, i.e., there exists a code  $e$  such that, for all  $\vec{R}$  and  $[n] \in \mathbb{N}/f^p(\vec{R})$ ,  $\alpha_{\vec{R}}([n]) = [e \cdot n]$ . Further, the morphism  $\alpha$  should respect relations, that is, if  $A_i \subset \mathbb{N}/R_i \times \mathbb{N}/S_i$  and  $(a, b) \in f^r(\vec{A})$  then  $(\alpha_{\vec{R}}(a), \alpha_{\vec{S}}(b)) \in g^r(\vec{A})$ .

Reindexing is by composition.

Next we define the fibration  $q$ . The fibre category  $\mathbf{UFam}(\mathbf{Asm})_n$  is defined as

**Objects** all maps  $f : \mathbf{Per}^n \rightarrow \mathbf{Asm}$ .

**Morphisms** a morphism from  $f$  to  $g$  is an indexed family of maps  $(\alpha_{\vec{R}})_{\vec{R} \in \mathbf{Per}^n}$  where

$$\alpha_{\vec{R}} : f(\vec{R}) \rightarrow g(\vec{R})$$

are maps between the underlying sets of the assemblies that are tracked uniformly, i.e. there exists a code  $e$  such that for all  $\vec{R}$  and all  $i \in f(\vec{R})$  and all  $a \in E_{f(\vec{R})}(i)$  we have  $e \cdot a \in E_{g(\vec{R})}(\alpha_{\vec{R}}(i))$ .

Reindexing is again by composition.

Finally we can define the category  $\mathbf{UFam}(\mathbf{RegSub}(\mathbf{Asm}))$  as

**Objects** An object over  $f$  is any family of subsets  $(A_{\vec{R}} \subseteq f(\vec{R}))_{\vec{R}}$ , where by subset we mean subset of the underlying set of the assembly.

**Morphisms** In each fibre the morphisms are just subset inclusions.

Reindexing is defined as follows: Suppose  $\phi : f \rightarrow g$  is a morphism in  $\mathbf{UFam}(\mathbf{Asm})$  projecting to  $q\phi : n \rightarrow m$  in  $\mathbf{PPer}$ . By definition this is a map in the fibre of  $\mathbf{UFam}(\mathbf{Asm})$  over  $n$  from  $f$  to  $(q\phi)^*(g)$ . Such morphisms are given by indexed families of maps

$$\phi_{\vec{R}} : f(\vec{R}) \rightarrow g \circ (q\phi)^p(\vec{R})$$

ranging over  $\vec{R} \in \mathbf{Per}^n$  so we can define

$$\phi^*(A_{\vec{S}} \subseteq g(\vec{S}))_{\vec{S} \in \mathbf{Per}^m} = (\phi_{\vec{R}}^{-1}(A_{g \circ (q\phi)^p(\vec{R})}))_{\vec{R} \in \mathbf{Per}^n}$$

The inclusion  $I$  is obtained by projecting  $(f^p, f^r)$  to  $f^p$  using the inclusion of  $\mathbf{Per}$  into  $\mathbf{Asm}$ .

**Lemma 6.1.**  $p$  is a  $\lambda 2$ -fibration.

*Proof.* This is (Jacobs, 1999, Prop. 8.4.10). The ccc-structure is given by a pointwise construction, and 1 is clearly a generic object. For a type  $f : n + 1 \rightarrow 1$  we define  $\prod f : n \rightarrow 1$  as

$$\begin{aligned} (\prod f)^p(\vec{R}) &= \{(a, a') \mid \forall U, V \in \mathbf{Per}. \forall B \subseteq \mathbb{N}/U \times \mathbb{N}/V \\ &\quad a \in |f^p(\vec{R}, U)| \text{ and } a' \in |f^p(\vec{R}, V)| \text{ and} \\ &\quad ([a], [a']) \in f^r_{(\vec{R}, U), (\vec{R}, V)}(\vec{E}q_{\vec{R}}, B)\} \end{aligned}$$

and

$$\begin{aligned}
 (\prod f)_{\vec{R} \times \vec{S}}^r(\vec{A}) &= \{([a]_{\prod(f)^p(\vec{R})}, [a']_{\prod(f)^p(\vec{S})}) \mid \forall U, V \in \mathbf{Per}. \forall B \subseteq \mathbb{N}/U \times \mathbb{N}/V \\
 &\quad ([a]_{f^p(\vec{R}, U)}, [a']_{f^p(\vec{S}, V)}) \in f_{(\vec{R}, U), (\vec{S}, V)}^r(\vec{A}, B)\}
 \end{aligned}$$

for  $\vec{A} \subseteq \vec{R} \times \vec{S}$ . □

**Theorem 6.2.** The diagram (23) defines a parametric APL-structure.

We do not prove Lemma 6.2 directly. Instead, we will show in Remark 8.27 that (23) is a special case of the parametric completion process of Section 8.

**Remark 6.3.** In the above model we use nothing special about the PCA  $\mathbb{N}$  so the same construction applies to pers and assemblies over any PCA. All the lemmas above generalize, so that in the general case we also obtain a parametric APL-structure.

### 6.1. A parametric non-well-pointed APL-structure

We may generalize the construction above even further to the case of relative realizability. Suppose we are given a PCA  $A$  and a sub-PCA  $A_\#$ . We can then define the APL-structure as above with pers and assemblies over  $A$ , with the only exception that morphisms in  $\mathbf{PFam}(\mathbf{Per})$  and  $\mathbf{UFam}(\mathbf{Asm})$  should be uniformly tracked by codes in  $A_\#$ . All the proofs of section 6 generalize so that we obtain:

**Proposition 6.4.** For any PCA  $A$  and sub-PCA  $A_\#$  the diagram

$$\begin{array}{ccc}
 & & \mathbf{UFam}(\mathbf{RegSub}(\mathbf{Asm}(\mathbf{A}, \mathbf{A}_\#))) \\
 & & \downarrow r \\
 \mathbf{PFam}(\mathbf{Per}(\mathbf{A}, \mathbf{A}_\#)) & \xrightarrow{I} & \mathbf{UFam}(\mathbf{Asm}(\mathbf{A}, \mathbf{A}_\#)) \\
 & \searrow p & \downarrow q \\
 & & \mathbf{PPer}(\mathbf{A}, \mathbf{A}_\#)
 \end{array}$$

defines a parametric APL-structure.

However, one may also prove:

**Proposition 6.5.** The fibre  $\mathbf{PFam}(\mathbf{Per}(\mathbf{A}, \mathbf{A}_\#))_0$  is in general not well-pointed.

*Proof.* Consider a per of the form  $\{(a, a)\}$ , for  $a \in A \setminus A_\#$ . There may be several maps out of this per, but it does not have any global points. □

Proposition 6.4 tells us that all the theorems of Section 5 apply, such that the  $\lambda_2$ -fibration  $\mathbf{PFam}(\mathbf{Per}(\mathbf{A}, \mathbf{A}_\#)) \rightarrow \mathbf{PPer}(\mathbf{A}, \mathbf{A}_\#)$  has all the properties that we consider consequences of parametricity. This should be compared to (Dunphy, 2004) in which a family of parametric models is presented (with another definition of “parametric model”) and the consequences of parametricity are proved only for the *well-pointed* parametric models.

## 7. Comparing with Ma & Reynolds notion of parametricity

In this section we compare the notion of parametricity presented above with Ma & Reynolds' notion of parametricity (Ma and Reynolds, 1992) (see also (Jacobs, 1999)). This latter notion was the first proposal for a general category theoretic formulation of parametricity and is perhaps the most well-known.

To define parametricity in the sense Ma & Reynolds, consider first a situation where we are given a  $\lambda_2$ -fibration  $E \longrightarrow B$  and a logic on the types given by an indexed first-order logic fibration

$$D \longrightarrow E \longrightarrow B .$$

Consider the category of relations on closed types  $LR(E_1)$  defined as

$$\begin{array}{ccccc} LR(E_1) & \longrightarrow & D_1 & \longrightarrow & D \\ \downarrow \lrcorner & & \downarrow \lrcorner & & \downarrow \\ E_1 \times E_1 & \xrightarrow{\times} & E_1 & \hookrightarrow & E \end{array}$$

where by 1 we mean the terminal object of  $B$ . In this case we have a reflexive graph of categories

$$E_1 \begin{array}{c} \longleftarrow \\ \rightleftarrows \\ \longrightarrow \end{array} LR(E_1) ,$$

where the functor going left to right maps a type to the identity on that type. By reflexive graph we mean that the two compositions starting and ending in  $E_1$  are identities.

**Definition 7.1.** The  $\lambda_2$ -fibration

$$\begin{array}{c} E \\ \downarrow \\ B \end{array}$$

is *parametric in the sense of Ma & Reynolds with respect to  $D \rightarrow E$*  if there exists a  $\lambda_2$ -fibration  $F \rightarrow C$  and a reflexive graph of  $\lambda_2$  fibrations

$$\left( \begin{array}{c} E \\ \downarrow \\ B \end{array} \right) \begin{array}{c} \longleftarrow \\ \rightleftarrows \\ \longrightarrow \end{array} \left( \begin{array}{c} F \\ \downarrow \\ C \end{array} \right)$$

such that the restriction to the fibres over the terminal objects becomes

$$E_1 \begin{array}{c} \longleftarrow \\ \rightleftarrows \\ \longrightarrow \end{array} LR(E_1) .$$

Given an APL-structure, we have a logic over types given by the pullback of **Prop** along  $I$ . We also have a reflexive graph giving the relational interpretation of all types. It is natural to ask what kind of parametricity we obtain by requiring that the reflexive graph giving the relational interpretation of types satisfies the requirements of Definition 7.1.

First we notice that **Relations**<sub>1</sub> =  $LR(E_1)$ , and that the two maps going from

**Relations** to  $E_1$  are in fact the domain and codomain maps, as required, so the requirements of Definition 7.1 only effect the nature of the map  $J$ .

The last requirement of Definition 7.1 says exactly that, for all closed types  $\sigma$ ,

$$J(\llbracket \sigma \rrbracket) = \llbracket eq_\sigma \rrbracket.$$

Consider now an open type  $\vec{\alpha} \vdash \sigma : \mathbf{Type}$  and a vector of closed types  $\vec{\tau}$ . Then, since  $J$  is a map of fibrations, we have

$$J(\llbracket \sigma(\vec{\tau}) \rrbracket) = J(\llbracket \vec{\tau} \rrbracket^* \llbracket \vec{\alpha} \vdash \sigma \rrbracket) = J(\llbracket \vec{\alpha} \vdash \sigma \rrbracket) \circ \llbracket eq_{\vec{\tau}} \rrbracket = \llbracket \sigma[eq_{\vec{\tau}}] \rrbracket.$$

In other words, the model satisfies a weak form of Identity Extension Schema:

**Definition 7.2.** The schema

$$\forall u, v : \sigma(\vec{\tau}). (u \sigma[eq_{\vec{\tau}}] v) \sqsupseteq u =_{\sigma(\vec{\tau})} v$$

where  $\vec{\alpha} \vdash \sigma$  ranges over all types and  $\vec{\tau}$  ranges over all closed types is called the *weak identity extension schema*.

We will briefly mention which of the consequences of parametricity mentioned in Section 5 that hold under assumption of the weak Identity Extension Schema.

First we notice that the weak Identity Extension Schema implies the parametricity schema

$$\forall u : (\prod \beta : \mathbf{Type}. \sigma(\beta, \tau_2, \dots, \tau_n)). u(\forall \beta. \sigma[\beta, eq_{\tau_2}, \dots, eq_{\tau_n}])u$$

in the case where the  $\tau_i$  are closed types.

Using only this weak version of the parametricity schema, we can still prove existence of terminal and initial types, since in these cases we only need to use parametricity on the closed types  $T$  and  $I$ .

The proofs of existence of products and coproducts, however, fail when  $\sigma$  and  $\tau$  are open types, since we need to use the parametricity condition on the open types  $\sigma \times \tau$  and  $\sigma + \tau$ .

The case of initial algebras goes through, since the proof only uses parametricity of  $\mu\alpha. \sigma(\alpha)$ , which is a closed type. The proof of Lemma 5.24, however, uses parametricity of the type  $\prod \alpha. ((\alpha \rightarrow \sigma(\alpha)) \times \alpha \rightarrow \beta)$  where  $\beta$  is a type variable, so this proof does not go through with only the weak parametricity schema. In other words, in the setting of reflexive graphs as in Definition 7.1, we do not have a proof of existence of final coalgebras.

See also (Robinson and Rosolini, 1994) for a related discussion.

## 8. A parametric completion process

In this section we give a description of a parametric completion process that given a model of  $\lambda_2$  internal to some category satisfying certain requirements produces a parametric APL-structure. The construction is related to the parametric completion process of (Robinson and Rosolini, 1994) in the sense that the process that constructs the  $\lambda_2$ -fibration contained in the APL-structure generated by our completion process is basically

the parametric completion process of (Robinson and Rosolini, 1994) (only the setup varies slightly). This means that if the ambient category is a topos, then the parametric completion process of (Robinson and Rosolini, 1994) produces models parametric in our new sense which then satisfies the consequences of parametricity of Section 5. This fact is no surprise, but, to our knowledge, it has not been proved in the literature.

The concrete model of Section 6 is a result of the parametric completion process described in this section. Before describing the completion process we recall the theory of internal models of  $\lambda_2$ .

### 8.1. Internal models for $\lambda_2$

Suppose we are given a locally cartesian closed category  $\mathbb{E}$ . Given a full internal category  $\mathbf{D}$  of  $\mathbb{E}$  we may consider the externalization  $\mathbf{D}$

$$\begin{array}{c} \text{Fam}(\mathbf{D}) . \\ \downarrow \\ \mathbb{E} \end{array}$$

We shall denote by  $\mathbf{D}_0$  the object of objects, and by  $\mathbf{D}_1$  the object of morphisms of  $\mathbf{D}$ . The fibre over  $\Xi \in \mathbb{E}$  is the internal functor category from  $\Xi$  considered as a discrete category to  $\mathbf{D}$ , i.e., objects are morphisms  $\Xi \rightarrow \mathbf{D}_0$  and morphism are morphisms of  $\mathbb{E}$ :  $\Xi \rightarrow \mathbf{D}_1$ .

**Proposition 8.1.** Suppose  $\mathbf{D}$  is a full internally cartesian closed category that has right Kan extensions for internal functors  $F : \Xi \rightarrow \mathbf{D}$  along projections  $\Xi \times \mathbf{D}_0 \rightarrow \Xi$ . Then the externalization of  $\mathbf{D}$  is a  $\lambda_2$ -fibration.

*Proof.* Since  $\mathbf{D}$  is internally cartesian closed, its externalization has cartesian closed fibres preserved under reindexing (Jacobs, 1999, Corollary 7.3.9). Clearly  $\mathbf{D}_0$  is a generic object for the fibration.

Polymorphism is modeled using the Kan extensions, since for any type  $\sigma : \Xi \times \mathbf{D}_0 \rightarrow \mathbf{D}$  the right Kan extension of  $\sigma$  along  $\pi : \Xi \times \mathbf{D}_0 \rightarrow \Xi$  is the functor  $\prod \alpha. \sigma$  in the diagram

$$\begin{array}{ccc} \Xi \times \mathbf{D}_0 & \xrightarrow{\sigma} & \mathbf{D} \\ \downarrow \pi & \nearrow \prod \alpha. \sigma & \\ \Xi & & \end{array}$$

The universality condition for the right Kan extension then gives the bijective correspondence

$$\text{Nat}(\tau \circ \pi, \sigma) \cong \text{Nat}(\tau, \prod \alpha. \sigma)$$

between the sets of natural transformations. Since  $\pi^* \tau = \tau \circ \pi$ , for  $\tau : \Xi \rightarrow \mathbf{D}$ , this states exactly that the right Kan extension provides the right adjoint to  $\pi^*$ , as required.

To show that the Beck-Chevalley condition is satisfied, we need to show that for  $u : \Xi' \rightarrow \Xi$  we have

$$u^*(\prod \alpha. \sigma) \cong \prod \alpha. ((u \times id)^* \sigma),$$

that is,

$$(\prod \alpha. \sigma) \circ u \cong \prod \alpha. (\sigma \circ (u \times id)).$$

By Lemma 8.2 below, we may write out the values of these two functors on objects  $A \in \Xi'$  as limits:

$$((\prod \alpha. \sigma) \circ u)(A) = \varprojlim_{u(A) \rightarrow \pi(A')} \sigma(A') \quad (24)$$

$$(\prod \alpha. (\sigma \circ u \times id))(A) = \varprojlim_{A \rightarrow \pi A''} \sigma(u \times id(A'')). \quad (25)$$

In (24) we take the limit over all maps  $f: u(A) \rightarrow \pi(A')$  in the discrete category  $\Xi$ . But since this is a discrete category, such maps only exist in the case  $\pi(A') = u(A)$ , so (24) can be rewritten as

$$\prod_{D' \in \mathbf{D}_0} \sigma(u(A), D').$$

Likewise (25) can be rewritten as

$$\prod_{D'' \in \mathbf{D}_0} \sigma(u(A), D''),$$

proving that the Beck-Chevalley condition is satisfied.  $\square$

**Lemma 8.2.** Suppose the Kan extension  $\text{RK}_H(F)$  in the diagram

$$\begin{array}{ccc} \mathbb{L} & \xrightarrow{H} & \mathbb{H} \\ F \downarrow & \swarrow & \text{RK}_H(F) \\ \mathbb{F} & & \end{array}$$

exists. If  $\mathbb{L}$ ,  $\mathbb{H}$  are discrete, then  $\text{RK}_H(F)$  is given as a pointwise limit construction (as in (Mac Lane, 1971, Theorem 1, p.237)).

Proposition 8.1 justifies the following definition.

**Definition 8.3.** An internal category  $\mathbf{D}$  of a locally cartesian closed category  $\mathbb{E}$  is called an *internal model* of  $\lambda_2$  if it satisfies the assumptions of Proposition 8.1.

### 8.2. Input for the parametric completion process

The parametric completion process takes the following ingredients as input:

- 1 A quasitopos  $\mathbb{E}$
- 2 An internal model  $\mathbf{D}$  of  $\lambda_2$  in  $\mathbb{E}$ .

We will further assume that the inclusion

$$\begin{array}{ccc} \text{Fam}(\mathbf{D}) & \longrightarrow & \mathbb{E} \rightarrow \\ & \searrow & \swarrow \text{cod} \\ & & \mathbb{E} \end{array}$$

which we have already assumed is full and faithful, preserves products and is closed under regular subobjects. The latter means that for each object  $E \in \mathbb{E}$ , the fibre category  $\mathbf{Fam}(\mathbf{D})_E$  is closed under regular subobjects as a subcategory of  $\mathbb{E}/E$ .

The logic  $\mathbf{RegSub}_{\mathbb{E}} \rightarrow \mathbb{E}$  of regular subobjects induces a logic on  $\mathbb{E}^{\rightarrow}$  by

$$\begin{array}{ccc} \mathbf{Q} & \longrightarrow & \mathbf{RegSub}_{\mathbb{E}} \\ \downarrow & \lrcorner & \downarrow \\ \mathbb{E}^{\rightarrow} & \xrightarrow{\text{dom}} & \mathbb{E}, \end{array}$$

which, by Lemma A.8, makes the composable fibration

$$\mathbf{Q} \longrightarrow \mathbb{E}^{\rightarrow} \xrightarrow{\text{cod}} \mathbb{E},$$

an indexed first-order logic fibration with an indexed family of generic objects, simple products and simple coproducts.

Let  $\Sigma$  be the regular subobject classifier of  $\mathbb{E}$ . We can now form an internal fibration<sup>‡</sup> by using the Grothendieck construction on the functor  $(d \in \mathbf{D}) \mapsto \Sigma^d$ , with  $\Sigma^d$  ordered pointwise. We think of this fibration as the internalization of  $\mathbf{RegSub}_{\mathbb{E}} \rightarrow \mathbb{E}$  restricted to  $\mathbf{D}$  and write it as  $a: \mathbf{Q} \rightarrow \mathbf{D}$ . Notice that since  $\mathbf{D}$  is closed under regular subobjects,  $\mathbf{Q} \rightarrow \mathbf{D}$  is a subfibration of the subobject fibration on  $\mathbf{D}$ , and since its externalization is simply the restriction of  $\mathbf{Q} \rightarrow \mathbb{E}^{\rightarrow}$ , it is closed under the logical operations  $\top, \wedge, \supset, \forall, =$  from the regular subobject fibration.

Associated to the model given by  $\mathbf{D}$  there is a canonical pre-APL- structure

$$\begin{array}{ccc} & & \mathbf{Q} \\ & & \downarrow \\ \mathbf{Fam}(\mathbf{D}) & \longrightarrow & \mathbb{E}^{\rightarrow} \\ & \searrow & \downarrow \\ & & \mathbb{E} \end{array} \tag{26}$$

To this we can associate, as usual, the fibration of relations denoted by  $\mathbf{Relations}_{\mathbf{D}} \rightarrow \mathbf{RelCtx}_{\mathbf{D}}$ .

### 8.3. The completion process

We define the category  $\mathbf{LR}(\mathbf{D})$  to have as objects logical relations of  $\mathbf{D}$  in the logic of  $\mathbf{Q}$  and as morphisms pairs of morphisms in  $\mathbf{D}$  that preserve relations.

**Lemma 8.4.** The category  $\mathbf{LR}(\mathbf{D})$  is an internal cartesian closed category of  $\mathbb{E}$ .

<sup>‡</sup> By internal fibration, we mean an internal functor, whose externalization is a fibration. By an internal fibration having structure such as  $\wedge, \supset, \forall, =$  we mean that the externalization has the same (indexed) structure



*Proof.* We set

$$\mathbf{LR}(\mathbf{D})_0 = \{(X, Y, \phi) \in \mathbf{D}_0 \times \mathbf{D}_0 \times \mathbf{Q}_0 \mid a(\phi) = X \times Y\}$$

and

$$\mathbf{LR}(\mathbf{D})_1 = \coprod_{(X, Y, \phi), (X', Y', \phi') \in \mathbf{LR}(\mathbf{D})_0} \{(f, g) \in \mathbf{D}_1 \times \mathbf{D}_1 \mid f : X \rightarrow X' \wedge g : Y \rightarrow Y' \wedge \phi \leq (f \times g)^* \phi'\}.$$

For the cartesian closed structure we define:

$$(X, Y, \phi) \times (X', Y', \phi') = (X \times X', Y \times Y', \phi \times \phi'),$$

where  $\phi \times \phi'((x, x'), (y, y')) = \phi(x, y) \wedge \phi'(x', y')$ , and

$$(X, Y, \phi) \rightarrow (X', Y', \phi') = (X \rightarrow X', Y \rightarrow Y', \phi \rightarrow \phi'),$$

where

$$\phi \rightarrow \phi'(f, g) = \forall x \in X \forall y \in Y (\phi(x, y) \supset \phi'(f(x), g(y))).$$

□

Let

$$G = \cdot \begin{array}{c} \longleftarrow \\ \rightleftarrows \\ \longrightarrow \end{array} \cdot$$

be the generic reflexive graph category, and consider the functor category  $\mathbb{E}^G$ . Since it is well known that  $\mathbf{Cat}(\mathbb{E}^G) \cong \mathbf{Cat}(\mathbb{E})^G$  and  $\mathbf{CCCat}(\mathbb{E}^G) \cong \mathbf{CCCat}(\mathbb{E})^G$  it follows that

**Lemma 8.5.**  $\mathbf{D} \begin{array}{c} \longleftarrow \\ \rightleftarrows \\ \longrightarrow \end{array} \mathbf{LR}(\mathbf{D})$  is an internal cartesian closed category of  $\mathbb{E}^G$ .

We now aim to prove that  $\mathbf{D} \begin{array}{c} \longleftarrow \\ \rightleftarrows \\ \longrightarrow \end{array} \mathbf{LR}(\mathbf{D})$  is an internal model of  $\lambda_2$ . By the lemma, all that remains is to prove that there are right Kan extensions for internal functors from  $\Xi \times \mathbf{D}_0 \begin{array}{c} \longleftarrow \\ \rightleftarrows \\ \longrightarrow \end{array} \Xi' \times \mathbf{LR}(\mathbf{D}_0)$  to  $\mathbf{D} \begin{array}{c} \longleftarrow \\ \rightleftarrows \\ \longrightarrow \end{array} \mathbf{LR}(\mathbf{D})$  along projections to  $\Xi \begin{array}{c} \longleftarrow \\ \rightleftarrows \\ \longrightarrow \end{array} \Xi'$ . This is the same as saying that the fibration

$$\mathbf{Fam}(\mathbf{D}^n \begin{array}{c} \longleftarrow \\ \rightleftarrows \\ \longrightarrow \end{array} \mathbf{LR}(\mathbf{D})^n) \rightarrow \mathbb{E}^G$$

has right adjoints to reindexing functors along projections.

We first consider the simpler case with spans in stead of reflexive graphs. Let  $\mathbf{R}(\mathbf{D})$  denote the internal category

$$\begin{array}{ccc} & \mathbf{LR}(\mathbf{D}) & \\ \partial_0 \swarrow & & \searrow \partial_1 \\ \mathbf{D} & & \mathbf{D} \end{array}$$

inside  $\mathbb{E}^\Lambda$ , where  $\Lambda$  is the obvious category.

An object of  $\mathbf{Fam}(\mathbf{R}(\mathbf{D}))$  is a triple of maps  $(f, g, \rho)$  such that

$$\begin{array}{ccccc} & & \mathbf{LR}(\mathbf{D})_0 & & \\ & \rho \nearrow & & \searrow & \\ \Theta & & & & \\ \swarrow & & \searrow & & \\ \Xi_0 & & \mathbf{D}_0 & & \mathbf{D}_0 \\ & f \nearrow & & \searrow g & \\ & \Xi_1 & & & \end{array} \quad (27)$$

commutes. Since  $\mathbf{LR}(\mathbf{D})_0$  is the object of all relations on objects of  $\mathbf{D}$ , the idea is that we can consider such a triple as a definable relation

$$\llbracket \Xi_0, \Xi_1 \mid \Theta \vdash \rho: \text{Rel}(f(\Xi_0), g(\Xi_1)) \rrbracket,$$

i.e., an object of  $\mathbf{Relations}_{\mathbf{D}}$ . We will make this intuition precise in Lemma 8.6.

A vertical morphism in the category  $\text{Fam}(\mathbf{R}(\mathbf{D}))$  from  $(f, g, \rho)$  to  $(f', g', \rho')$  is by definition a triple consisting of a morphism from  $f$  to  $f'$ , a morphism from  $g$  to  $g'$  and a morphism from  $\rho$  to  $\rho'$ . But since morphisms in  $\mathbf{LR}(\mathbf{D})$  are pairs of morphisms preserving relations, and since the triple of morphisms is required to make the obvious diagram commute, we can consider such a morphism as a pair  $(s: f \rightarrow f', t: g \rightarrow g')$  such that

$$\forall A \in \Theta. \forall x: f(\partial_0(A)), y: g(\partial_1(A)). \rho(x, y) \supset \rho'(s_{\partial_0(A)}(x), t_{\partial_1(A)}(y)),$$

as interpreted in the internal language of the quasi-topos, where  $\supset$  refers to the internal ordering in  $\mathbf{Q}$ .

**Lemma 8.6.** There is an isomorphism of fibrations

$$\left( \begin{array}{c} \text{Fam}(\mathbf{R}(\mathbf{D})) \\ \downarrow \\ \mathbb{E}^\Lambda \end{array} \right) \xrightarrow{\cong} \left( \begin{array}{c} \mathbf{Relations}_{\mathbf{D}} \\ \downarrow \\ \mathbf{RelCtx}_{\mathbf{D}} \end{array} \right)$$

*Proof.* Unwinding the definition of  $\mathbf{RelCtx}_{\mathbf{D}}$ , we find that the objects are triples  $(\Xi_0, \Xi_1, \Xi)$  together with maps  $\Xi \rightarrow \Xi_0 \times \Xi_1$  in  $\mathbb{E}$ . A map from  $\Xi \rightarrow \Xi_0 \times \Xi_1$  to  $\Xi' \rightarrow \Xi'_0 \times \Xi'_1$  is a triple  $\rho: \Xi \rightarrow \Xi', f: \Xi_0 \rightarrow \Xi'_0, g: \Xi_1 \rightarrow \Xi'_1$  making the obvious diagram commute. Thus  $\mathbf{RelCtx}_{\mathbf{D}} \cong \mathbb{E}^\Lambda$ .

Objects in  $\mathbf{Relations}_{\mathbf{D}}$  are given as morphism in  $\mathbf{RelCtx}_{\mathbf{D}}$  into the interpretation of  $\alpha, \beta \mid R: \text{Rel}(\alpha, \beta)$  in (26). But the interpretation of this is easily seen to be

$$\coprod_{\alpha, \beta \in \mathbf{D}_0} \Sigma^{\alpha \times \beta} \rightarrow \mathbf{D}_0 \times \mathbf{D}_0,$$

and since  $\mathbf{LR}(\mathbf{D})_0 = \coprod_{\alpha, \beta \in \mathbf{D}_0} \Sigma^{\alpha \times \beta}$  we get a bijective correspondence between objects of  $\mathbf{Relations}_{\mathbf{D}}$  and objects of  $\text{Fam}(\mathbf{R}(\mathbf{D}))$ . For morphisms, a vertical morphism in  $\text{Fam}(\mathbf{R}(\mathbf{D}))$  from  $(f, g, \rho)$  to  $(f', g', \rho')$  is by the above discussion a pair of morphisms  $t: f \rightarrow f', s: g \rightarrow g'$  satisfying  $\rho \supset (t \times s)^* \rho'$ , which is exactly the same as a vertical morphism in  $\mathbf{Relations}_{\mathbf{D}}$ .  $\square$

**Lemma 8.7.** All internal functors  $\begin{array}{c} \Xi \\ \swarrow \quad \searrow \\ \Xi_0 \quad \Xi_1 \end{array} \times \mathbf{R}(\mathbf{D})_0 \rightarrow \mathbf{R}(\mathbf{D})$  have right Kan extensions along the projection to  $\begin{array}{c} \Xi \\ \swarrow \quad \searrow \\ \Xi_0 \quad \Xi_1 \end{array}$

*Proof.* The statement to be proved is equivalent to the statement that the fibration on the left hand side of the isomorphism of Lemma 8.6 has simple products. Since we know that the fibration on the right of the isomorphism has simple products, we are done.  $\square$

Let us now consider the case that we are really interested in. We shall assume that we are given a functor  $(f^t, f^r)$  in  $\mathbb{E}^G$ :

$$\begin{array}{ccc}
 \Xi' \times \mathbf{LR}(\mathbf{D})_0 & \xrightarrow{\pi} & \Xi' \\
 \partial_0 \downarrow \uparrow \downarrow \uparrow \partial_1 & & \partial_0 \downarrow \uparrow \downarrow \uparrow \partial_1 \\
 \Xi \times \mathbf{D}_0 & \xrightarrow{\pi} & \Xi \\
 & \searrow f^r & \\
 & & \mathbf{LR}(\mathbf{D}) \\
 & \searrow f^t & \\
 & & \mathbf{D},
 \end{array} \tag{28}$$

and we would like to find a right Kan extension of  $(f^t, f^r)$  along  $(\pi, \pi)$  (notice that we have used the notation  $\partial_0, \partial_1, I$  for the structure maps of all objects of  $\mathbb{E}^G$  - this should not cause any confusion, since it will be clear from the context which map is referred to). Let us call this extension  $(\prod_{par} f^t, \prod_{par} f^r)$ . An obvious idea is to try the pair  $(\prod f^t, \prod f^r)$  provided by Lemma 8.7. However,  $\prod_{par} f^r$  should commute with  $I$ , and we cannot know that  $\prod f^r$  will do that. Consider  $\prod f^r(I(A))$  for some  $A \in \Xi$ :

$$\begin{array}{c}
 \prod f^r(I(A)) \\
 \downarrow \\
 \prod f^t(A) \times \prod f^t(A).
 \end{array}$$

If we pull this relation back along the diagonal on  $\prod f^t(A)$  we get a subobject

$$|\prod f^r(I(A))| \twoheadrightarrow \prod f^t(A)$$

(called the *field* of  $\prod f^r(I(A))$ ). Logically,  $|\prod f^r(I(A))|$  is the set  $\{x \in \prod f^t(A) \mid (x, x) \in \prod f^r(I(A))\}$ , so if we restrict  $\prod f^r(I(A))$  to this subobject, we get a relation containing the identity relation. The other inclusion will be easy to prove. Thus the idea is to let  $\prod_{par} f^t$  be the map that maps  $A$  to  $|\prod f^r(I(A))|$ , and let  $\prod_{par} f^r(R)$  be the relation obtained by restricting  $\prod f^r(R)$  to  $\prod_{par} f^t(\partial_0(R)) \times \prod_{par} f^t(\partial_1(R))$ .

**Theorem 8.8.** For  $(f^t, f^r)$ ,  $(\pi, \pi)$  as in (28), the right Kan extension of  $(f^t, f^r)$  along  $(\pi, \pi)$  exists.

*Proof.* We will define  $\prod_{par} f^t(A)$  as the pullback

$$\begin{array}{ccc}
 (\prod_{par} f^t)(A) & \longrightarrow & (\prod f^r)(I(A)) \\
 \downarrow \lrcorner & & \downarrow \\
 \prod f^t(A) & \xrightarrow{\Delta} & \prod f^t(A) \times \prod f^t(A)
 \end{array}$$

where  $\Delta$  is the diagonal map. We define  $\prod_{par} f^r(R)$  for  $R \in \Xi'$ , to be the pullback

$$\begin{array}{ccc} (\prod_{par} f^r)(R) & \longrightarrow & (\prod f^r)(R) \\ \downarrow \lrcorner & & \downarrow \\ \prod_{par} f^t(\partial_0 R) \times \prod_{par} f^t(\partial_1 R) & \twoheadrightarrow & \prod f^t(\partial_0 R) \times \prod f^t(\partial_1 R). \end{array}$$

First we will show that  $\prod_{par} f^r(I(A)) = I(\prod_{par} f^t(A))$  for all  $A$ . Logically

$$\begin{aligned} \prod_{par} f^r(I(A)) &= \{(x, y) \in \prod f^r(I(A)) \mid (y, y), (x, x) \in \prod f^r(I(A))\} \supseteq \\ &\quad \{(x, x) \mid x \in \prod f^r(I(A))\} = I(\prod_{par} f^t(A)) \end{aligned}$$

To prove the other inclusion suppose  $(x, y) \in \prod_{par} f^r(I(A)) \subseteq \prod f^r(I(A))$ . Then for any  $\sigma_{n+1} \in D_0$ ,

$$(x, y) \in \pi^*(\prod f^r)(I(A), I(\sigma_{n+1})).$$

Let  $\epsilon_{A, \sigma_{n+1}}$  denote the appropriate component of the counit for  $\pi^* \dashv \prod$ . Then

$$(\epsilon_{A, \sigma_{n+1}} x, \epsilon_{A, \sigma_{n+1}} y) \in \pi^*(\prod f^r)(I(A), I(\sigma_{n+1})) = I(f^t(A, \sigma_{n+1})),$$

so  $\epsilon_{A, \sigma_{n+1}} x = \epsilon_{A, \sigma_{n+1}} y$ . Since  $\prod f^t(A)$  is the product of  $f^t(A, \sigma_{n+1})$  over  $\sigma_{n+1}$  in  $\mathbf{D}_0$ , and  $\epsilon_{A, \sigma_{n+1}}$  is simply the projection onto the  $\sigma_{n+1}$ -component,  $\epsilon_{A, \sigma_{n+1}} x = \epsilon_{A, \sigma_{n+1}} y$  for all  $\sigma_{n+1}$  implies  $x = y$  as desired.

Finally we will show that  $\prod_{par}$  provides the desired right adjoint. Recall that a morphism from  $(g^t, g^r)$  to  $(h^t, h^r)$ , where

$$\begin{array}{ccc} \Xi' & \xrightarrow{g^r} & \mathbf{LR}(\mathbf{D})_0 \\ \partial_0 \downarrow \uparrow \downarrow \uparrow \partial_1 & & \partial_0 \downarrow \uparrow \downarrow \uparrow \partial_1 \\ \Xi & \xrightarrow{g^t} & \mathbf{D}_0 \end{array}$$

and likewise  $(h^t, h^r)$  is a morphism  $s: g^t \rightarrow h^t$  preserving relations. In the internal language this means that for each  $A \in \Xi$  we have a map  $s_A: g^t(A) \rightarrow h^t(A)$  such that for  $R$  with  $\partial_0(R) = A$ ,  $\partial_1(R) = B$ ,  $(x, y) \in g^r(R)$  implies  $(s_A(x), s_B(y)) \in h^r(R)$ .

Now, from Lemma 8.7 we easily derive a one-to-one correspondence between maps  $(g^t, g^r) \rightarrow (\prod f^t, \prod f^r)$  and maps  $(g^t \circ \pi, g^r \circ \pi) \rightarrow (f^t, f^r)$ . Since  $\prod_{par} f^t(A) \subseteq \prod f^t(A)$ , for this correspondence to carry over, we only need to check that if  $s$  denotes a map from  $(g^t \circ \pi, g^r \circ \pi)$  to  $(f^t, f^r)$ , and  $\tilde{s}$  the adjoint correspondent to  $s$ , then  $\tilde{s}$  preserves relations, and if  $x \in g^t(A)$ , then  $\tilde{s}(x) \in \prod_{par} f^t(A)$ . But since  $(x, x) \in g^r(I(A)) = I(g^t(A))$ , we must have  $(\tilde{s}(x), \tilde{s}(x)) \in \prod f^r(I(A))$ , so  $\tilde{s}(x) \in \prod_{par} f^t(A)$  as desired. For the preservation of relations, suppose  $(x, y) \in g^r(R)$ . Then

$$(\tilde{s}(x), \tilde{s}(y)) \in \prod f^r(R) \cap \prod_{par} f^t(\partial_0 R) \times \prod_{par} f^t(\partial_1 R) = \prod_{par} f^r(R).$$

□

**Corollary 8.9.** The fibration  $\text{Fam}(\mathbf{LR}(\mathbf{D}) \overset{\leftarrow}{\rightleftarrows} \mathbf{D}) \rightarrow \mathbb{E}^G$  is a  $\lambda_2$ -fibration.

**Remark 8.10.** If  $\mathbb{E}$  is a topos then  $\mathbf{Q}$  is the subobject fibration on  $\mathbf{D}$ , and  $\mathbb{T} \rightarrow \mathbb{K}$  is in fact the model of  $\lambda_2$  that Robinson and Rosolini prove to be parametric in the sense of reflexive graphs (Definition 7.1) in (Robinson and Rosolini, 1994). One interesting difference however, is that (Robinson and Rosolini, 1994) considered only models of  $\lambda_2$  that satisfied a “suitability for polymorphism” condition stating that the model is closed under  $\mathbf{LR}(\mathbf{D})_0$ -products. In our setup, this condition is replaced by the condition that the regular subobject fibration models  $\forall$ , and that the internal category  $\mathbf{D}$  is closed under regular subobjects.

**Remark 8.11.** Consider a morphism  $\xi$  between types  $f$  and  $g$  in the model  $\mathbb{T} \rightarrow \mathbb{K}$ . At first sight, such a morphism is a pair of morphism  $(\xi_0, \xi_1)$  with  $\xi_i : f_i \rightarrow g_i$ . But morphisms in  $\mathbf{LR}(\mathbf{D})$  are given by pairs of maps in  $\mathbf{D}$ , and commutativity of

$$\begin{array}{ccc} \mathbf{LR}(\mathbf{D})_0^n & \xrightarrow{\xi_1} & \mathbf{LR}(\mathbf{D})_1 \\ \partial_i \downarrow & & \downarrow \partial_i \\ \mathbf{D}_0^n & \xrightarrow{\xi_0} & \mathbf{D}_1 \end{array}$$

tells us that  $\xi_1$  must be given by  $(\xi_0, \xi_0)$ . Thus *morphisms between types are morphisms between the usual interpretations of types preserving the relational interpretations.*

#### 8.4. The APL-structure

In this section we embed the  $\lambda_2$  fibration of Corollary 8.9 into a full parametric APL-structure.

Consider the functor  $(\cdot)_0 : \mathbb{E}^G \rightarrow \mathbb{E}$  that maps a diagram  $X_0 \begin{smallmatrix} \longleftarrow \\ \rightleftarrows \\ \longrightarrow \end{smallmatrix} X_1$  to  $X_0$ , and consider the pullback of (26) along  $(\cdot)_0$ :

$$\begin{array}{ccc} & & \mathbb{P} \\ & & \downarrow \\ \mathbb{T} & \hookrightarrow & \mathbb{C} \\ & \searrow & \downarrow \\ & & \mathbb{E}^G. \end{array} \tag{29}$$

**Lemma 8.12.** The functor  $(\cdot)_0$  extends to a morphism of fibrations:

$$\begin{array}{ccc} \mathbf{Fam} \left( \begin{array}{c} \mathbf{LR}(\mathbf{D}) \\ \downarrow \uparrow \downarrow \\ \mathbf{D} \end{array} \right) & \xrightarrow{(\cdot)_0} & \mathbf{Fam}(\mathbf{D}) \\ \downarrow & & \downarrow \\ \mathbb{E}^G & \xrightarrow{(\cdot)_0} & \mathbb{E}. \end{array}$$

*Proof.* The required map maps an object

$$\begin{pmatrix} X_1 \\ \downarrow \uparrow \downarrow \\ X_0 \end{pmatrix} \longrightarrow \begin{pmatrix} \mathbf{LR}(\mathbf{D})_0 \\ \downarrow \uparrow \downarrow \\ \mathbf{D}_0 \end{pmatrix}$$

of  $\text{Fam} \left( \begin{array}{c} \mathbf{LR}(\mathbf{D})_0 \\ \Downarrow \Uparrow \Downarrow \\ \mathbf{D}_0 \end{array} \right)$  to the object  $X_0 \longrightarrow \mathbf{D}_0$  of  $\text{Fam}(\mathbf{D})$ . Likewise for morphisms.  $\square$

As a consequence of Lemma 8.12 we can extend (29) to

$$\begin{array}{ccccc} & & & & \mathbb{P} \\ & & & & \downarrow \\ \text{Fam} \left( \begin{array}{c} \mathbf{LR}(\mathbf{D})_0 \\ \Downarrow \Uparrow \Downarrow \\ \mathbf{D}_0 \end{array} \right) & \longrightarrow & \mathbb{T} & \xrightarrow{\subset} & \mathbb{C} \\ & \searrow & \searrow & & \downarrow \\ & & & & \mathbb{E}^G. \end{array} \quad (30)$$

If we erase  $\mathbb{T}$  from (30) we obtain the diagram

$$\begin{array}{ccccc} & & & & \mathbb{P} \\ & & & & \downarrow \\ \text{Fam} \left( \begin{array}{c} \mathbf{LR}(\mathbf{D}) \\ \Downarrow \Uparrow \Downarrow \\ \mathbf{D} \end{array} \right) & \xrightarrow{\subset \quad I} & & & \mathbb{C} \\ & \searrow & & & \downarrow \\ & & & & \mathbb{E}^G. \end{array} \quad (31)$$

**Theorem 8.13.** The diagram (31) defines a parametric APL-structure.

We will prove Theorem 8.13 in a series of lemmas.

**Corollary 8.14.** If  $\mathbf{D}$  is an internal model of  $\lambda_2$  in a topos, which is closed under subobjects, then the parametric completion process of (Robinson and Rosolini, 1994) provides a  $\lambda_2$ -fibration that satisfies the consequences of parametricity provable in Abadi & Plotkin's logic.

*Proof.* This follows from Remark 8.10.  $\square$

**Remark 8.15.** The types (the objects of  $\text{Fam} \left( \begin{array}{c} \mathbf{LR}(\mathbf{D}) \\ \Downarrow \Uparrow \Downarrow \\ \mathbf{D} \end{array} \right)$ ) in the APL-structure (31) are morphisms

$$\left( \begin{array}{c} \mathbf{LR}(\mathbf{D})_0 \\ \Downarrow \Uparrow \Downarrow \\ \mathbf{D}_0 \end{array} \right)^n \rightarrow \left( \begin{array}{c} \mathbf{LR}(\mathbf{D})_0 \\ \Downarrow \Uparrow \Downarrow \\ \mathbf{D}_0 \end{array} \right)$$

in  $\mathbb{E}^G$ . Thus types contain both the usual interpretation (the map  $f_0 : \mathbf{D}_0^n \rightarrow \mathbf{D}_0$ ) and a relational interpretation (the map  $f_1 : \mathbf{LR}(\mathbf{D})_0^n \rightarrow \mathbf{LR}(\mathbf{D})_0$ ). But since the map  $\text{Fam} \left( \begin{array}{c} \mathbf{LR}(\mathbf{D})_0 \\ \Downarrow \Uparrow \Downarrow \\ \mathbf{D}_0 \end{array} \right) \rightarrow \mathbb{T}$  forgets the relational interpretation, the logic on types, given by  $\mathbb{P}$ , is given only by the logic on the usual interpretation of the types. To be more precise, a logical relation in the model of (31) between types  $f$  and  $g$  is a relation in the sense of the logic  $\mathbb{Q}$  between  $\coprod_{\vec{d} \in \mathbf{D}_0^n} f_0(\vec{d}) \rightarrow \mathbf{D}_0^n$  and  $\coprod_{\vec{d} \in \mathbf{D}_0^n} g_0(\vec{d}) \rightarrow \mathbf{D}_0^n$ .

Notice also that the relational interpretation of a type (given by  $f_1$ ) is in a sense parametric since the diagram

$$\begin{array}{ccc} \mathbf{LR}(\mathbf{D})_0^n & \xrightarrow{f_1} & \mathbf{LR}(\mathbf{D})_0 \\ \uparrow i & & \uparrow i \\ \mathbf{D}_0^n & \xrightarrow{f_0} & \mathbf{D}_0 \end{array}$$

is required to commute. This is basically the reason why the APL-structure is parametric.

**Remark 8.16.** One may restrict the APL-structure of (31) to the full subcategory of  $\mathbb{E}^G$  on powers of the generic object. This way one obtains a  $\lambda_2$ -fibration in which **Type** is the only kind. To prove that this defines a parametric APL-structure, one will need to change the proof presented here slightly to obtain the reflexive graph.

**Lemma 8.17.**  $\mathbb{C} \rightarrow \mathbb{K}$  is fibred cartesian closed and  $I$  is a faithful product-preserving functor.

*Proof.* The first statement follows from the fact that  $\mathbb{E}^\rightarrow \rightarrow \mathbb{E}$  is a fibred cartesian closed fibration.

$I$  is a restriction of the composition

$$\begin{array}{ccccc} \text{Fam} \left( \begin{array}{c} \mathbf{LR}(\mathbf{D}) \\ \Downarrow \Uparrow \Downarrow \\ \mathbf{D} \end{array} \right) & \longrightarrow & \mathbb{T} & \hookrightarrow & \mathbb{C}' \\ & & & \searrow & \downarrow \\ & & & & \mathbb{E}^G \end{array}$$

The map  $\mathbb{T} \rightarrow \mathbb{C}'$  is the pullback of the inclusion of the externalization of a full internal cartesian closed category into  $\mathbb{E}^\rightarrow$ . This is faithful and product preserving by assumption.

The map  $\text{Fam} \left( \begin{array}{c} \mathbf{LR}(\mathbf{D}) \\ \Downarrow \Uparrow \Downarrow \\ \mathbf{D} \end{array} \right) \rightarrow \mathbb{T}$  is the map that maps

$$f : \left( \begin{array}{c} \mathbf{LR}(\mathbf{D})_0 \\ \Downarrow \Uparrow \Downarrow \\ \mathbf{D}_0 \end{array} \right)^n \rightarrow \left( \begin{array}{c} \mathbf{LR}(\mathbf{D})_i \\ \Downarrow \Uparrow \Downarrow \\ \mathbf{D}_i \end{array} \right)$$

to  $f_0 : \mathbf{D}_0^n \rightarrow \mathbf{D}_i$  (for  $i = 0, 1$  denoting objects and morphisms respectively). Since product structure of internal categories of graph categories is given pointwise, this map clearly preserves fibred products.

As mentioned in Remark 8.11, a morphism from  $f$  to  $g$  with

$$f, g : \left( \begin{array}{c} \mathbf{LR}(\mathbf{D})_0 \\ \Downarrow \Uparrow \Downarrow \\ \mathbf{D}_0 \end{array} \right)^n \rightarrow \left( \begin{array}{c} \mathbf{LR}(\mathbf{D})_0 \\ \Downarrow \Uparrow \Downarrow \\ \mathbf{D}_0 \end{array} \right)$$

is just a map from  $f_0$  to  $g_0$  preserving relations. Thus the first map is also faithful.  $\square$

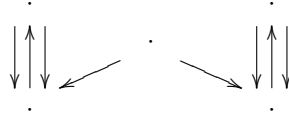
**Lemma 8.18.** The composable fibration  $\mathbb{P} \rightarrow \mathbb{C} \rightarrow \mathbb{K}$  is an indexed first-order logic fibration with an indexed family of generic objects. Moreover, the composable fibration has simple products, simple coproducts and very strong equality.

*Proof.* The composable fibration  $\mathbb{P} \rightarrow \mathbb{C} \rightarrow \mathbb{K}$  is a pullback of  $\mathbb{Q} \rightarrow \mathbb{E} \rightarrow \mathbb{E}$  which has the desired properties according to Lemma A.8. All of this structure is always preserved under pullback, except simple products and coproducts. These are preserved since the map  $\mathbb{K} \rightarrow \mathbb{E}$  preserves products.  $\square$

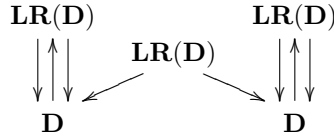
As in Remark 3.4 we can now construct the functor  $U$  as needed in Definition 3.3. Thus we have:

**Proposition 8.19.** The diagram (31) defines a pre-APL-structure with very strong equality.

Consider the graph  $W$ :



where we assume that the two graphs included are reflexive graphs. The graph  $\mathbf{W}$ :



defines an internal category in  $\mathbb{E}^W$ .

An object of  $\text{Fam}(\mathbf{W})$  can be denoted by a triple  $(f, g, \rho)$ , where  $f$  and  $g$  are types in the same fibre (that is, objects of  $\text{Fam} \left( \begin{smallmatrix} \text{LR}(\mathbf{D}) \\ \Downarrow \Uparrow \Downarrow \\ \mathbf{D} \end{smallmatrix} \right)$  in the same fibre) and  $\rho$  is a morphism  $\text{LR}(\mathbf{D})_0^n \rightarrow \text{LR}(\mathbf{D})_0$  such that the diagram

$$\begin{array}{ccccc}
 & & & \text{LR}(\mathbf{D})_0 & \\
 & & \rho & \nearrow & \\
 \text{LR}(\mathbf{D})_0^n & & & & \\
 \swarrow & & \searrow & & \\
 \mathbf{D}_0^n & & \mathbf{D}_0 & & \mathbf{D}_0 \\
 \nearrow & f_0 & \rightarrow & & \nearrow g_0 \\
 \mathbf{D}_0^n & & & & 
 \end{array} \tag{32}$$

commutes.

Now, as noted in Remark 8.15 types in the pre-APL structure (31) are given by both an ordinary interpretation of types and a relational interpretation of types, but relations between types are just given by relations between the ordinary interpretation of types. Thus we may think of such triples as objects of the form

$$\llbracket \vec{\alpha}, \vec{\beta} \mid \vec{R} : \text{Rel}(\vec{\alpha}, \vec{\beta}) \vdash \phi(R) : \text{Rel}(f(\vec{\alpha}), g(\vec{\beta})) \rrbracket$$

in the category **Relations** as formed from the pre-APL structure (31), in the same way as in Lemma 8.6.

Note that since we have proved that the diagram (31) defines a pre-APL-structure, we can reason about it using the parts of Abadi & Plotkin's logic not involving the relational interpretation of types. In the following we shall use this to work in the internal language of the pre-APL-structure.



**Proposition 8.20.** There is an isomorphism of fibrations:

$$\left( \begin{array}{c} \mathbf{Fam}(\mathbf{W}) \\ \downarrow \\ \mathbb{E}^W \end{array} \right) \xrightarrow{\cong} \left( \begin{array}{c} \mathbf{Relations} \\ \downarrow \\ \mathbf{RelCtx} \end{array} \right)$$

*Proof.* The argument is essentially the same as the proof of Lemma 8.6.  $\square$

**Lemma 8.21.** The graph  $\mathbf{W}$  is an internal model of  $\lambda_2$  in  $\mathbb{E}^W$ .

*Proof.* This is a consequence of Proposition 8.20.  $\square$

**Proposition 8.22.** There is a reflexive graph of  $\lambda_2$ -fibrations

$$\left( \begin{array}{c} \mathbf{Fam} \left( \begin{array}{c} \mathbf{LR}(\mathbf{D}) \\ \downarrow \uparrow \downarrow \\ \mathbf{D} \end{array} \right) \\ \downarrow \\ \mathbb{E}^G \end{array} \right) \begin{array}{c} \longleftarrow \\ \longrightarrow \\ \longleftarrow \end{array} \left( \begin{array}{c} \mathbf{Fam}(\mathbf{W}) \\ \downarrow \\ \mathbb{E}^W \end{array} \right)$$

**Remark 8.23.** The reflexive graph in (Robinson and Rosolini, 1994) arises this way, although the setup of (Robinson and Rosolini, 1994) is slightly different.

*Proof.* An object of  $\mathbf{Fam}(\mathbf{W})$  is a map in  $\mathbb{E}^W$

$$\left( \begin{array}{c} \Xi_1 \\ \uparrow \downarrow \\ \Xi_2 \end{array} \right) \begin{array}{c} \longleftarrow \\ \longrightarrow \\ \longleftarrow \end{array} \left( \begin{array}{c} \Xi_4 \\ \uparrow \downarrow \\ \Xi_5 \end{array} \right) \rightarrow \left( \begin{array}{c} \mathbf{LR}(\mathbf{D})_0 \\ \uparrow \downarrow \\ \mathbf{D}_0 \end{array} \right) \begin{array}{c} \longleftarrow \\ \longrightarrow \\ \longleftarrow \end{array} \left( \begin{array}{c} \mathbf{LR}(\mathbf{D})_0 \\ \uparrow \downarrow \\ \mathbf{D}_0 \end{array} \right)$$

Let us denote such objects as triples  $(f, g, \rho)$  where  $f: \left( \begin{array}{c} \Xi_1 \\ \uparrow \downarrow \\ \Xi_2 \end{array} \right) \rightarrow \left( \begin{array}{c} \mathbf{LR}(\mathbf{D})_0 \\ \uparrow \downarrow \\ \mathbf{D}_0 \end{array} \right)$ ,  $g: \left( \begin{array}{c} \Xi_4 \\ \uparrow \downarrow \\ \Xi_5 \end{array} \right) \rightarrow \left( \begin{array}{c} \mathbf{LR}(\mathbf{D})_0 \\ \uparrow \downarrow \\ \mathbf{D}_0 \end{array} \right)$  and  $\rho: \Xi_3 \rightarrow \mathbf{LR}(\mathbf{D})_0$ . The domain and codomain maps of the postulated reflexive graph map  $(f, g, \rho)$  to  $f$  and  $g$  respectively, and the last map maps  $f$  to  $(f, f, f_1)$ .

The domain and codomain map preserve simple products since from the viewpoint of Proposition 8.22 these are just the domain and codomain map of Lemma 3.7. The middle map component of the simple products in  $\mathbf{Fam}(\mathbf{W}) \rightarrow \mathbb{E}^W$  is computed by computing the simple products as in Lemma 8.7 and then restricting the the right domain and codomain. Since this is the same as the computation of the relational part of the simple products of  $\mathbf{Fam} \left( \begin{array}{c} \mathbf{LR}(\mathbf{D}) \\ \uparrow \downarrow \\ \mathbf{D} \end{array} \right)$ , the last map of the reflexive graph also commutes with simple products.  $\square$

**Proposition 8.24.** The pre-APL-structure (31) has a full APL-structure.

*Proof.* This follows from Proposition 8.22 and Proposition 8.20.  $\square$

**Lemma 8.25.** The APL-structure (31) satisfies extensionality.

*Proof.* The model has very strong equality, which implies extensionality (4.2).  $\square$

**Lemma 8.26.** The APL-structure (31) satisfies the identity extension axiom.

*Proof.* Consider a type  $f$  with  $n$  free variables. We need to show that

$$\langle id_{\Omega^n}, id_{\Omega^n} \rangle^* J(f) \circ [\vec{\alpha} \mid - \mid - \vdash eq_{\vec{\alpha}}] = [\vec{\alpha} \vdash eq_{f(\vec{\alpha})}].$$

The map  $J$  is defined as the composition of two maps. The first map maps  $f$  to  $(f, f, f_1)$ :

$$\left( \begin{array}{ccc} \mathbf{LR}(\mathbf{D})_0^n & & \mathbf{LR}(\mathbf{D})_0^n \\ \downarrow \uparrow \downarrow & \mathbf{LR}(\mathbf{D})_0^n & \downarrow \uparrow \downarrow \\ \mathbf{D}_0^n & & \mathbf{D}_0^n \end{array} \right) \rightarrow \left( \begin{array}{ccc} \mathbf{LR}(\mathbf{D})_0 & & \mathbf{LR}(\mathbf{D})_0 \\ \downarrow \uparrow \downarrow & \mathbf{LR}(\mathbf{D})_0 & \downarrow \uparrow \downarrow \\ \mathbf{D}_0 & & \mathbf{D}_0 \end{array} \right).$$

Since  $f$  makes the diagram

$$\begin{array}{ccc} \mathbf{LR}(\mathbf{D})_0^n & \xrightarrow{f_1} & \mathbf{LR}(\mathbf{D})_0 \\ \downarrow \uparrow \downarrow & & \downarrow \uparrow \downarrow \\ \mathbf{D}_0^n & \xrightarrow{f_0} & \mathbf{D}_0 \end{array}$$

commute we know that  $f_1(eq_{\vec{\alpha}}) = eq_{f_0(\vec{\alpha})}$ .  $\square$

Theorem 8.13 is now the collected statement of 8.19 8.24, 8.25 and 8.26.

**Remark 8.27.** As mentioned in the introduction to this section, the concrete APL-structure of Section 6 can be considered as a result of the parametric completion process. If we consider the internal category  $\mathbf{Per}$  in the category  $\mathbf{Asm}$  of assemblies, then using the parametric completion process on this data we obtain the APL-structure of Section 6. To see this, we need to use the fact that there exists an isomorphism of fibrations

$$\begin{array}{ccc} \mathbf{UFam}(\mathbf{Asm}) & \xrightarrow{\cong} & \mathbf{Asm}^{\rightarrow} \\ & \searrow & \swarrow \\ & \mathbf{Asm}. & \end{array}$$

This proves Theorem 6.2.

## 9. Parametric Internal Models

The definition of APL-structure admittedly asks for a substantial amount of structure. In this section we sketch how much of that structure may be derived in the case of internal models of  $\lambda_2$ .

Let  $\mathbb{E}$  be a quasi-topos and let  $j$  be a local operator (also known as closure operator or Lawvere-Tierney topology) on  $\mathbb{E}$ . We write  $\mathbb{E}_j$  for the full subcategory of  $j$ -sheaves,  $\mathbf{a}$  for the associated sheaf functor,  $I$  for the inclusion of  $j$ -sheaves, and  $\eta$  for the natural transformation  $Id \rightarrow I\mathbf{a}$ .

Let  $\mathbf{C}$  be an internal model of  $\lambda_2$  in  $\mathbb{E}$ . Then  $\mathbf{a}\mathbf{C}$  is an internal category in  $\mathbb{E}_j$  and  $\eta : \mathbf{C} \rightarrow \mathbf{a}\mathbf{C}$  is an internal functor.

Consider the following diagram:

$$\begin{array}{ccccccc}
 & & \mathbb{P} & \longrightarrow & \mathbb{S} & \longrightarrow & \text{RegSub}_{\mathbb{E}_j} \\
 & & \downarrow \lrcorner & & \downarrow \lrcorner & & \downarrow \\
 \mathbf{Fam}(\mathbf{C}) & \xrightarrow{I} & \mathbf{Fam}(\mathbb{E}_j) & \longrightarrow & \mathbb{E}_j \dashrightarrow & \xrightarrow{\text{dom}} & \mathbb{E}_j \\
 & \searrow & \downarrow \lrcorner & & \downarrow & & \\
 & & \mathbb{E} & \xrightarrow{\mathbf{a}} & \mathbb{E}_j & & 
 \end{array} \tag{33}$$

where  $I$  is the functor induced by the composition of the internal functor  $\eta : \mathbf{C} \rightarrow \mathbf{a}\mathbf{C}$  and the inclusion of the externalization of  $\mathbf{C}$  into  $\mathbb{E}_j \dashrightarrow$  is faithful.

Suppose that

- the internal functor  $\eta : \mathbf{C} \rightarrow \mathbf{a}\mathbf{C}$  is faithful,
- the internal category  $\mathbf{a}\mathbf{C}$  is a subcategory of  $\mathbb{E}_j$  (i.e., the inclusion of the externalization of  $\mathbf{C}$  into  $\mathbb{E}_j \dashrightarrow$  is faithful).

Then the functor  $I$  in the above diagram is faithful and the leftmost part of the diagram (33) (the part going down and left from  $\mathbb{P}$ ) is a pre-APL-structure, and we can thus define that **the internal  $\lambda_2$  model  $\mathbf{C}$  in  $\mathbb{E}$  is parametric with respect to  $j$**  if this pre-APL-structure is a parametric APL-structure.

One should, of course, think of  $j$  as specifying the logic with respect to which the model is parametric.

The completion process presented in the previous section takes a full internal  $\lambda_2$  model in a quasi-topos  $\mathbb{F}$  and produces an internal model in  $\mathbb{E} = \mathbb{F}^G$  with  $j$  on  $\mathbb{E}$  such that  $\mathbb{F} = \mathbb{E}_j$  (the associated sheaf functor  $\mathbf{a}$  takes  $\downarrow \uparrow \downarrow$  to  $X_0$ ) and which satisfies the two items above ensuring that  $I$  is faithful. The results in the previous section then show that the internal model in  $\mathbb{F}$  is parametric with respect to this  $j$ .

This description of parametric internal models allows us to state precisely the (still) open problem of whether there exists parametric models that are inherently parametric (not constructed though a completion process):

**Problem 9.1.** Does there exist a full internal  $\lambda_2$  model in a quasi-topos  $\mathbb{E}$  that is parametric with respect to the trivial topology  $j$  (such that  $\mathbb{E}_j = \mathbb{E}$ ) ?

## 10. Conclusion

We have defined the notion of an APL-structure and proved that it provides sound and complete models for Abadi and Plotkin's logic for parametricity, thereby answering a question posed in (Plotkin and Abadi, 1993, page 5). We have also defined a notion of parametric APL-structures, for which we can prove the expected consequences of parametricity using the internal logic. The consequences proved in this document are existence of inductive and coinductive datatypes. These consequences have, to our knowledge not been proved in general for models parametric in the sense of Ma & Reynolds, but only for specific models.

We have presented a family of parametric models, some of which are not well-pointed. This means that our notion of parametricity is useful also in the absence of well-pointedness.

We have provided an extension of the parametric completion process of (Robinson and Rosolini, 1994) that produces parametric APL-structures. This means that for a large class of models, we have proved that the parametric completion of Robinson and Rosolini produce models that satisfy the consequences of parametricity.

In subsequent papers we will show how to modify the parametric completion process to produce domain-theoretic parametric models and how to extend the notion of APL-structure to include models of polymorphic linear lambda calculus (Plotkin, 1993).

## Appendix A. Composable Fibrations

This appendix is concerned with the theory of composable fibrations, by which we simply mean pairs of fibrations such that the codomain of the first is the domain of the second fibration. This appendix contains definitions referred to in the text.

Suppose we are given a composable fibration:

$$\mathbb{F} \xrightarrow{p} \mathbb{E} \xrightarrow{q} \mathbb{B}$$

We observe that

- The composite  $qp$  is a fibration. This is easily seen from the definition.
- If  $p$  and  $q$  are cloven, we may choose a cleavage by lifting  $u$  twice to  $\bar{u}$  for each  $I$  in  $\text{Obj } \mathbb{F}$  and  $u : X \rightarrow qpI$ .
- If  $p, q$  are split the composite fibration will be split since  $\overline{\bar{v}u} = \overline{\bar{v} \circ \bar{u}} = \overline{\bar{v}} \circ \bar{u}$ .

Thus in the case above we may consider the composable fibration as a doubly indexed category, and reindexing in  $\mathbb{F}$  with respect to  $u$  in  $\mathbb{B}$  is given by  $\bar{u}^*$

The lemmas below refer to the fibrations  $p, q$  above.

**Definition A.1.** We say that  $(\Omega_A)_{A \in \text{Obj } \mathbb{B}}$  is an indexed family of generic objects for the composable pair of fibrations  $(p, q)$  if for all  $A$ ,  $\Omega_A \in \text{Obj } \mathbb{E}_A$  is a generic object for the restriction of  $p$  to  $\mathbb{E}_A$  and if the family is closed under reindexing, i.e., for all morphisms  $u : A \rightarrow B$  in  $\mathbb{B}$ ,  $u^*(\Omega_B) \cong \Omega_A$ .

Before we define the concept of an indexed first-order logic fibration, we recall the definition of first-order logic fibration from (Jacobs, 1999) .

**Definition A.2.** A fibration  $p : \mathbb{F} \rightarrow \mathbb{E}$  is called a *first-order logic fibration* if

- $p$  is a fibred preorder that is fibred bicartesian closed.
- $\mathbb{E}$  has products.
- $p$  has simple products and coproducts, i.e., right, respectively left adjoints to reindexing functors induced by projections, and these satisfy the Beck-Chevalley condition.
- $p$  has fibred equality, i.e., left adjoints to reindexing functors induced by  $id \times \Delta : I \times J \rightarrow I \times J \times J$ , satisfying the Beck-Chevalley condition.

Readers worried about the Frobenius condition should note that this comes for free in fibred cartesian closed categories.

**Definition A.3.** We say that  $(p, q)$  has indexed (simple) products/coproducts/equality if each restriction of  $p$  to a fibre of  $q$  has the same satisfying the Beck-Chevalley condition, and these commute with reindexing, i.e., if  $u$  is a map in  $\mathbb{B}$  then there is a natural isomorphism  $\bar{u}^* \prod_f \cong \prod_{u^*f} \bar{u}^*$  or  $\bar{u}^* \coprod_f \cong \coprod_{u^*f} \bar{u}^*$  (this can also be viewed as a Beck-Chevalley condition).

**Definition A.4.** We say that  $(p, q)$  is an *indexed first order logic fibration* if  $p$  is a fibrewise bicartesian closed preorder, and  $(p, q)$  has indexed simple products, indexed simple coproducts and indexed equality.

We can also talk about composable fibrations  $(p, q)$  simply having products, coproducts, etc. This should be the case if the composite  $qp$  has (co-)products, but we should also require the right Beck-Chevalley conditions to hold. Notice that since  $u^*$  in  $qp$  is the same as  $\bar{u}^*$  in  $p$  we can write the product as either  $\prod_u$  in  $qp$  or  $\prod_{\bar{u}}$  in  $p$ .

**Definition A.5.** We say that the composable fibration  $(p, q)$  has products / coproducts if for each map  $u: I \rightarrow J$  in  $\mathbb{B}$ , and each object  $X \in \mathbb{E}_J$  the reindexing functor  $u^*: \mathbb{F}_X \rightarrow \mathbb{F}_{u^*X}$  has a right / left adjoint. Moreover, these (co)-products must satisfy the Beck-Chevalley condition for two sorts of diagram corresponding to reindexing in  $\mathbb{B}$  and  $\mathbb{E}$  respectively. First if

$$\begin{array}{ccc} H & \xrightarrow{v} & K \\ a \downarrow & \lrcorner & \downarrow b \\ I & \xrightarrow{u} & J \end{array}$$

is a pullback diagram in  $\mathbb{B}$ , then by (Jacobs, 1999, Exercise 1.4.4)

$$\begin{array}{ccc} a^*u^*X & \xrightarrow{\bar{v}} & b^*X \\ \bar{a} \downarrow & & \downarrow \bar{b} \\ u^*X & \xrightarrow{\bar{u}} & X \end{array}$$

is a pullback diagram in  $\mathbb{E}$ , and we require that the Beck-Chevalley condition is satisfied with respect to this diagram. Second, if  $f: Y \rightarrow X$  is a vertical map in  $\mathbb{E}$ , then the Beck-Chevalley condition should be satisfied with respect to the diagram

$$\begin{array}{ccc} u^*Y & \xrightarrow{\bar{u}} & Y \\ u^*f \downarrow & \lrcorner & \downarrow f \\ u^*X & \xrightarrow{\bar{u}} & X \end{array} \tag{34}$$

which by the way is a pullback by (Jacobs, 1999, Exercise 1.4.4).

The composable fibration  $(p, q)$  has simple (co-)products if it has (co-)products with respect to projections as defined above.

In the case of the APL-structures, the logical content of the Beck-Chevalley condition for diagrams of the form (34) will be that

$$(\forall \alpha: \text{Type}. \phi)[t/x] = \forall \alpha: \text{Type}. (\phi[t/x]).$$

**Definition A.6.** We say that a first-order logic fibration has *very strong equality* if internal equality in the fibration implies external equality.

**Definition A.7.** We say that the indexed first order logic fibration  $(p, q)$  has *very strong equality* if each restriction of  $p$  to a fibre of  $q$  has.

The next lemma gives a way of obtaining indexed first-order logic fibrations.

**Lemma A.8.** Suppose  $\mathbb{Q}' \rightarrow \mathbb{E}$  is a first-order logic fibration with a generic object on a locally cartesian closed category  $\mathbb{E}$ . Suppose further, that  $\mathbb{Q}' \rightarrow \mathbb{E}$  has products and coproducts with respect to maps  $A \times_B A' \rightarrow A$  from pullback diagrams

$$\begin{array}{ccc} A \times_B A' & \longrightarrow & A \\ \downarrow & \lrcorner & \downarrow \\ A' & \longrightarrow & B, \end{array}$$

and coproducts with respect to maps

$$id_C \times_B \Delta_A : C \times_B \times A \rightarrow C \times_B A \times_B A,$$

all satisfying the Beck-Chevalley condition. Then the composable fibration

$$\mathbb{Q} \longrightarrow \mathbb{E} \rightarrow \xrightarrow{\text{cod}} \mathbb{E},$$

where  $\mathbb{Q} \rightarrow \mathbb{E}^\rightarrow$  is the pullback

$$\begin{array}{ccc} \mathbb{Q} & \longrightarrow & \mathbb{Q}' \\ \downarrow & \lrcorner & \downarrow \\ \mathbb{E}^\rightarrow & \xrightarrow{\text{dom}} & \mathbb{E}, \end{array}$$

is an indexed first-order logic fibration with an indexed family of generic objects, simple products and simple coproducts. Moreover, if  $\mathbb{Q}' \rightarrow \mathbb{E}$  has very strong equality, so does the composable fibration.

*Proof.* The fibred bicartesian structure exists since the fibres of  $\mathbb{Q} \rightarrow \mathbb{E}^\rightarrow$  are the fibres of  $\mathbb{Q}' \rightarrow \mathbb{E}$ . This structure is clearly preserved by reindexing.

The fibrewise product of  $A \rightarrow B$  and  $A' \rightarrow B$  in  $\mathbb{E}^\rightarrow$  is  $A \times_B A' \rightarrow B$  with projection

$$\begin{array}{ccc} A \times_B A' & \xrightarrow{\pi} & A \\ & \searrow & \swarrow \\ & B & \end{array}$$

The indexed (co-)product along this map in  $\mathbb{Q} \rightarrow \mathbb{E}^\rightarrow$  is the (co-)product along  $\pi$  in  $\mathbb{E}$ , which exists by assumption. For the Beck-Chevalley condition for vertical pullbacks,

recall that the domain functor  $\mathbb{E}^\rightarrow \rightarrow \mathbb{E}$  preserves pullbacks, so for a vertical map

$$\begin{array}{ccc} A'' & \xrightarrow{f} & A \\ & \searrow & \swarrow \\ & B & \end{array}$$

taking the pullback of  $\pi$  along  $f$  in the category  $\mathbb{E}^\rightarrow$ , and then applying the domain functor gives the pullback

$$\begin{array}{ccc} A'' \times_B A' & \longrightarrow & A \times_B A' \\ \downarrow \lrcorner & & \downarrow \\ A'' & \xrightarrow{f} & A \end{array}$$

in  $\mathbb{E}$ , so that the Beck-Chevalley condition in this case reduces to Beck-Chevalley for the fibration  $\mathbb{Q}' \rightarrow \mathbb{E}$ .

To prove that these indexed simple (co-)products commute with reindexing, consider a map  $u: B' \rightarrow B$  in  $\mathbb{E}$ . We need to prove that for the diagram

$$\begin{array}{ccccc} & & u^*(A) \times_{B'} u^*(A') & \xrightarrow{\bar{u}} & A \times_B A' \\ & \swarrow \pi & \downarrow & & \swarrow \pi \\ u^*A & \xrightarrow{\bar{u}} & A & & A \\ & \searrow & \downarrow & & \downarrow \\ & & B' & \xrightarrow{u} & B, \end{array}$$

we have, for products  $\bar{u}^* \prod_\pi \cong \prod_\pi \bar{u}^*$  and for coproducts  $\bar{u}^* \coprod_\pi \cong \coprod_\pi \bar{u}^*$ . But this follows from the Beck-Chevalley condition in  $\mathbb{Q}' \rightarrow \mathbb{E}$ .

Indexed fibred equality is given by coproduct along maps

$$id_C \times_B \Delta_A: C \times_B A \rightarrow C \times_B A \times_B A,$$

which are required to exist. As with indexed (co-)products, the Beck-Chevalley conditions reduce to the Beck-Chevalley conditions for  $\mathbb{Q}' \rightarrow \mathbb{E}$ .

We define the family of generic objects to be the projections  $(\Sigma \times B \rightarrow B)_{B \in \mathbb{E}}$  in  $\mathbb{E}^\rightarrow$  where  $\Sigma$  is the generic object of  $\mathbb{Q} \rightarrow \mathbb{E}$ . This family is clearly closed under reindexing, and maps

$$\begin{array}{ccc} A & \xrightarrow{h} & \Sigma \times B \\ & \searrow f & \swarrow \pi \\ & B & \end{array}$$

correspond to maps  $A \rightarrow \Sigma$  in  $\mathbb{E}$ , which correspond to objects of  $\mathbb{Q}'_A \cong \mathbb{Q}_f$ .

We shall prove that we have simple products; simple coproducts are proved similarly.

Suppose  $\pi : D \times D' \rightarrow D$  is a projection in  $\mathbb{E}$ . For  $f : A \rightarrow D$  in  $\mathbb{E}^{\rightarrow}$ ,  $\bar{\pi}$  is the map

$$\begin{array}{ccc} A \times D' & \xrightarrow{\pi} & A \\ f \times id \downarrow & \lrcorner & \downarrow f \\ D \times D' & \xrightarrow{\pi} & D. \end{array}$$

Reindexing along this map in  $\mathbb{Q}$  corresponds to reindexing in  $\mathbb{Q}'$  along  $\pi : A \times D' \rightarrow A$ , so by existence of simple products in  $\mathbb{Q}' \rightarrow \mathbb{E}$  we have a right adjoint  $\pi^* \dashv \prod_{\pi}$ .

We need to prove Beck-Chevalley first for pullbacks in  $\mathbb{E}$ . In this case a pullback in  $\mathbb{E}$

$$\begin{array}{ccc} D \times D'' & \xrightarrow{id_D \times u} & D \times D' \\ \downarrow \lrcorner \pi' & & \downarrow \pi' \\ D'' & \xrightarrow{u} & D' \end{array}$$

lifts to the pullback

$$\begin{array}{ccccc} & & D \times u^* A & \xrightarrow{id \times \bar{u}} & D \times A \\ & \swarrow \pi' & \downarrow \bar{u} & \swarrow \pi' & \downarrow id \times f \\ u^* A & \xrightarrow{\quad} & A & & \\ \downarrow & \swarrow \pi' & \downarrow f & \swarrow \pi' & \\ D'' & \xrightarrow{u} & D' & & \end{array}$$

in  $\mathbb{E}^{\rightarrow}$ . The Beck-Chevalley condition for this pullback reduces to the Beck-Chevalley condition for the upper square in  $\mathbb{Q}' \rightarrow \mathbb{E}$  which is known to hold.

We should also check that the Beck-Chevalley condition holds in the case of the pullback.

$$\begin{array}{ccccc} & & A' \times D' & \xrightarrow{\bar{\pi}} & A' \\ & \swarrow h \times id & \downarrow & \swarrow h & \downarrow \\ A \times D' & \xrightarrow{\quad} & A & & \\ & \searrow & \downarrow & \searrow & \\ & & D \times D' & \xrightarrow{\pi} & D \end{array}$$

But again this reduces to the Beck-Chevalley condition for  $\mathbb{Q}' \rightarrow \mathbb{E}$  because  $\bar{\pi}$  is a projection.

Very strong equality is clearly preserved. □

**References**

Dunphy, B. (2004). *Parametricity as a notion of uniformity in reflexive graphs*. PhD thesis. 1, 6.1  
 Dunphy, B. and Reddy, U. S. (2004). Parametric limits. In *Proceedings of the 19th IEEE Symposium on Logic in Computer Science (LICS-04)*, pages 242–251.



- Hasegawa, R. (1994). Categorical data types in parametric polymorphism. *Mathematical Structures in Computer Science*, 4:71–109. 1, 5
- Hyland, J., Robinson, E., and Rosolini, G. (1990). The discrete objects in the effective topos. *Proc. London Math. Soc.*, 3(60):1–36. 1
- Jacobs, B. (1999). *Categorical Logic and Type Theory*, volume 141 of *Studies in Logic and the Foundations of Mathematics*. Elsevier Science Publishers B.V. 1, †, 2, 2.1, 3.1, 4, 5.6, 6, 6, 6, 7, 8.1, A, A.5, A
- Ma, Q. and Reynolds, J. (1992). Types, abstraction, and parametric polymorphism, part 2. In Brookes, S., Main, M., Melton, A., Mislove, M., and Schmidt, D., editors, *Mathematical Foundations of Programming Semantics*, volume 598 of *Lecture Notes in Computer Science*, pages 1–40. Springer-Verlag. 1, 1, 7
- Mac Lane, S. (1971). *Categories for the Working Mathematician*. Springer-Verlag. 8.2
- Mitchell, J. (1996). *Foundations for Programming Languages*. MIT Press. 1
- Pitts, A. (1987). Polymorphism is set theoretic, constructively. In Pitt, D. H., Poigné, A., and Rydeheard, D. E., editors, *Category Theory and Computer Science, Proc. Edinburgh 1987*, volume 283 of *Lecture Notes in Computer Science*, pages 12–39. Springer-Verlag. 1
- Pitts, A. M. (1989). Non-trivial power types can't be subtypes of polymorphic types. In *4th Annual Symposium on Logic in Computer Science*, pages 6–13. IEEE Computer Society Press, Washington. 1
- Plotkin, G. (1993). Second order type theory and recursion. Notes for a talk at the Scott Fest. 1, 10
- Plotkin, G. and Abadi, M. (1993). A logic for parametric polymorphism. In *Typed lambda calculi and applications (Utrecht, 1993)*, volume 664 of *Lecture Notes in Comput. Sci.*, pages 361–375. Springer, Berlin. (document), 1, 2, 2.3, 2.4, 3.2, 3.18, 4, 5, 5.1, 5.4, 5.16, 10
- Reynolds, J. (1983). Types, abstraction, and parametric polymorphism. *Information Processing*, 83:513–523. 1
- Reynolds, J. (1984). Polymorphism is not set-theoretic. In Kahn, G., MacQueen, D. B., and Plotkin, G. D., editors, *Semantics of Data Types*, volume 173 of *Lecture Notes in Computer Science*, pages 145–156. Springer-Verlag. 1
- Robinson, E. and Rosolini, G. (1994). Reflexive graphs and parametric polymorphism. In Abramsky, S., editor, *Proc. 9th Symposium in Logic in Computer Science*, pages 364–371, Paris. I.E.E.E. Computer Society. 1, 7, 8, 8.10, 8.14, 8.23, 10
- Rosolini, G. (1995). Notes on synthetic domain theory. Draft.
- Rummelhoff, I. (2004). Polynat in PER-models. *Theoretical Computer Science*, 316(1–3):215–224.
- Seely, R. (1987). Categorical semantics of higher-order polymorphic lambda calculus. *The Journal of Symbolic Logic*, 52(4):969–989. 1
- Takeuti, I. (1998). An axiomatic system of parametricity. *Fund. Inform.*, 33(4):397–432. Typed lambda-calculi and applications (Nancy, 1997). 2.2
- Wadler, P. (1989). Theorems for free! In *4th Symposium on Functional Programming Languages and Computer Architecture, ACM, London*. 1, 1
- Wadler, P. (2004). The Girard-Reynolds isomorphism (second edition). Manuscript. 2.2