MATHEMATICAL PROBLEMS IN COMPUTATION THEORY
BANACH CENTER PUBLICATIONS, VOLUME 21
PWN – POLISH SCIENTIFIC PUBLISHERS
WARSAW 1988

# CATEGORIES OF PARTIAL MORPHISMS AND THE RELATION BETWEEN TYPE STRUCTURES*

## PARTIAL VARIATIONS ON A THEME OF FRIEDMAN AND STATMAN

ANDREA ASPERTI
GIUSEPPE LONGO**

*Universita' degli Studi di Pisa, Dipartimento di Informatica, Pisa, Italia*

## Introduction

In Friedman [7] a result of semantic completeness for typed $\lambda$-calculus is given, by using the full type structure over $\omega$. That is, two typed terms are shown to be provably equal iff they define the same functional of finite type over $\omega$ (i.e. the same morphism in the category of sets and all functions). The key to Friedman's result is a simple and elegant notion of homomorphism between type-structures. We wish to extend Friedman's notion and its consequences to more constructive settings.

As a matter of fact both typed and type-free $\lambda$-calculus have been primarily regarded as a formalization of the concept of effective process or computation. Indeed, $\lambda$-terms are extremely adequate to describe computable functions and functionals (see Barendregt [1], Goedel [9] and Troelstra [35], say). Moreover, since Scott's work, the role of $\lambda$-calculus and its extensions in denotational semantics of programming languages is well known, as it provides the core of functional programming languages.

For these reasons, we are interested in models which yield useful properties for the theory of programs. For example, they should provide (effective) solutions to equations which recursively define programs and data

---

* A preliminary version of this paper, focusing on Section 3, appeared in LNCS, Springer-Verlag (CAAP'86). Research partially supported by Min. P. I., fondi 40% (Mat).

** Invited Lecture, Semester in Theory of Computation, Banach Mathematical Center, Warsaw, December 1985.

types. These may be given by using the effectively given domains of Scott [28]. Actually, one may take, as objects, the countable collections of the computable elements in the effectively given domains, that is, one may consider the category **CD** of constructive domains (see Giannini–Longo [8] for an introduction). **CD** is a model for typed $\lambda$-calculus, is closed under inverse limits and limits are also preserved by the functors which give higher type objects (product, exponentiation...). By this, the semantics of the recursive definitions of programs and data may be soundly and effectively given (see Kanda [12], Smyth [29], Smyth–Plotkin [30]). By the work of Ershov, it also yields models for the Kleene–Kreisel (countable) recursive functionals (see Ershov [6], Longo–Moggi [14]).

Note that the fully effective flavour of **CD** is due to the fact that morphisms and functors are represented by recursive functions over suitable indexing of objects. Indeed, **CD** is one of the several interesting subcategories of the category $\mathbf{PER}_\omega$, of countable (quotient) sets, we will consider. $\mathbf{PER}_\omega$ may be loosely viewed as the constructive counterpart of **Set**, the usual category of sets and functions (see Hyland [11]). The choice of countable (and numbered) data types is a very natural one also for the purposes of Computer Science.

The aim of this paper is to compare type-structures, i.e., models of typed $\lambda$-calculus, when data types are taken to be (possibly structured) countable sets and morphisms to be effective transformations. This will be done by tools related to Friedman's homomorphisms and Plotkin–Statman's logical relations (see later or Friedman [7], Plotkin [21], Statman [32]).

A key point in the present perspective, as much as in Friedman's, is the possibility of handling partial functions. In our case, though, we cannot rely on the intuitive notion of partiality for set-theoretic functions. Therefore, Section 1 is devoted to an introduction to categories with partial morphisms and to "complete objects", which will turn out to be relevant notions for the proof of the main theorem. The elementary, but detailed, presentation of partial categories in Section 1 is independently motivated also by the increasing interest in (typed) partial computations and in their denotational semantics (see Plotkin [22]).

Section 2 introduces and discusses "partial retraction systems". The methods in Section 2 are endebted to the work in Friedman [7] and Plotkin [21]. Note, though, that Friedman's technique relies on an highly non-constructive use of the axiom of choice for sets. By an informal analogy, we may say that, in our approach, choice functions are replaced by retraction pairs, which are "effective" relatively to the intended category. Section 2 does not depend on Section 1.

Finally, in the section which originally motivated our work, Section 3, we apply the results presented in Sections 1 and 2 to the relation between $\mathbf{PER}_\omega$ and the typed $\lambda$-calculus by using partial retraction systems.

## 1. Partial morphisms and complete objects

In this section we survey, from a new perspective, and develop the notions introduced in Longo–Moggi [15]; in particular, we focus on the notion of "complete object".

The natural setting for partial morphisms, in Category Theory, is the Theory of Toposes, as it is mostly motivated by the categorical treatment of well-known set-theoretic notions (subset, inverse image...). We will not get into the details of Topos Theory and we sketch a more elementary approach to the interpretation of "divergence" in familiar categories. Moggi [18] and Rosolini [25] elegantly work out various aspects of a topos-theoretic approach to partial morphisms.

A stronger, but related, notion of Dominical Category may be found in Di Paola–Heller [5]. In Dominical Categories the "domain" of a morphism (see below) is given as an endomorphism; the results in this section are still valid in those categories. Our approach will be related to the notion of partial morphism in Obtułowicz [19].

A sound requirement for a category, in order to allow partial morphisms, is the existence of everywhere divergent morphisms. These must correspond to everywhere divergent functions and, thus, behave like a zero w.r.t. right and left composition.

**1.1.** Definition. A category $C$ has *partial morphisms* (is a pC) iff

$$\forall b, c \in Ob_C \exists O_{bc} \in C[b, c]\, \forall a, d \in Ob_C \forall f \in C[a, b]\, \forall g \in C[c, d],$$

$$O_{bc} \circ f = O_{ac} \quad \text{and} \quad g \circ O_{bc} = O_{cd}.$$

**1.2.** Definition. Let $C$ be a pC. A morphism $f \in C[a, b]$ is *total* iff for all $c \in Ob_C$ and all $g \in C[c, a]$, $f \circ g = O_{cb} \Rightarrow g = O_{ca}$.

**1.3.** *Remark.* Note that: (1) $f, g$ total $\Rightarrow f \circ g$ total,

(2) $f \circ g$ total $\Rightarrow g$ *total*,

(3) all monomorphisms are total.

**1.4.** Definition. Let $C$ be a pC and $f \in C[a, b]$. A morphism $g \in C[c, a]$ is *into the domain of* $f$ (in short: $g$ into $\mathrm{dom}(f)$) iff $f \circ g$ is total.

The idea should be clear: $g$ has values in the "domain of convergence" of $f$ iff $f \circ g$ "converges" everywhere. Just note that $g$ itself must be total; this may seems a little unpleasant, but the results below should convince the reader acquainted with the other approaches to partial morphisms that the definition above is both reasonable and useful (see, in particular, 1.6, 1.14, 1.22).

**1.5.** Definition. Let $C$ be a pC. The (associated) category of *total morphisms*, $C_T$, has the same objects as $C$ and as morphisms the total ones.

**Inc**: $C_T \to C$ is the inclusion functor defined in the obvious way (it will be omitted if there is no ambiguity).

**1.6.** Lemma. *Let $C$ be a* pC *and $f \in C[a, b]$. Then the following are equivalent*:

1. $(\mathrm{id}_a$ *into* $\mathrm{dom}(f))$,
2. $f \in C_T[a, b]$,
3. $\forall c \in Ob_C \, \forall h \in C_T[c, a]$ $(h$ *into* $\mathrm{dom}(f))$.

*Proof.* $1 \Rightarrow 2$. By $f \circ \mathrm{id}_a = f$.

$2 \Rightarrow 3$. By 1.3.1.

$3 \Rightarrow 1$. Obvious. ∎

Examples I. **pSet** is the category of sets with partial maps as morphisms. **pR** = **PR** is the monoid of the Partial Recursive functions (PR). **pEN** is the pC whose objects are pairs $\boldsymbol{a} = (a, e)$, where $a$ is a countable set and $e$: $\omega \to a$ (onto) is an enumeration of $a$. Moreover, $f$ is in **pEN**$[\boldsymbol{a}, \boldsymbol{a}']$ iff there exists $f' \in$ PR such that the f.d.c.:

$$\begin{tikzcd} \omega \arrow[r, "f'"] \arrow[d, "e"'] & \omega \arrow[d, "e'"] \\ a \arrow[r, "f"'] & a' \end{tikzcd}$$

Clearly, $(\mathbf{pSet})_T = \mathbf{Set}$, $\mathbf{PR}_T = R$ and $(\mathbf{pEN})_T = \mathbf{EN}$, that is Malcev's category of numbered sets.

Since the beginning of denotational semantics of programming languages, the basic notions of approximation and continuity suggested the introduction of posets with a least element $\perp$. The bottom $\perp$ provides the meaning to diverging computations over non-trivial mathematical structures. This is mathematically very clear in several specific categories, such as continuous or algebraic lattices or cpo's, Scott's domains... It is not so obvious in interesting categories for computations such as **EN**, say (see Example III below).

(For typographical reasons, we write $a^\circ$ instead of $a^\perp$; lifting and complete objects below were first defined, under different names, in Longo–Moggi [15].)

**1.7.** Definition. Let $C$ be a pC. Then the *lifting* of $a \in Ob_C$ is the object $a^\circ$ such that the functors $C[-, a] \circ \mathbf{Inc}$, $C_T[-, a^\circ]$: $C_T \to \mathbf{Set}$ are naturally isomorphic.

Recall that, by definition of natural transformation, of hom-functor and by the definition of **Inc**, 1.7 requires the existence of a function $\tau$ such that the f.d.c., for all $b$, $c \in C$ and $f \in C_T[c, b]$:

(Diag. $N$)

$$\begin{tikzcd} C[b,a] \arrow[r, "\tau b"] \arrow[d, "\_\circ f"'] & C_T[b,a^\circ] \arrow[d, "\_\circ f"] \\ C[c,a] \arrow[r, "\tau c"'] & C_T[c,a^\circ] \end{tikzcd}$$

That is, $\tau c(g \circ f) = \tau b(g) \circ f$ and, also, $(\tau c)^{-1}(h \circ f) = (\tau b)^{-1}(h) \circ f$.

It is easy to check that $a^\circ$ defined as in 1.7 is unique, if it exists. The idea for $a^\circ$ is very simple: as $C[-, a] \simeq C_T[-, a^\circ]$, in a pC any partial morphism may be uniquely extended to a total one, when the target object is "lifted" (and the lifting exists).

Recall now that in a category $K$, given objects $a$ and $b$, $a$ is a *retract* of $b$ (*notation*: $a < b$) if there exist $i \in K[a, b]$ and $j \in K[b, a]$ such that $j \circ i = \mathrm{id}_a$.

**1.8.** LEMMA. *Let $C$ be a* pC *and $a^\circ$ be the lifting of $a \in Ob_C$. Assume also that $\tau$ is the natural isomorphism in* 1.7. *Then $a$ is a retract of $a^\circ$ in $C$ (notation: $a <_p a^\circ$) via* $(\mathrm{in}_a, \mathrm{ex}_a)$, *where*

$$\mathrm{in}_a = \tau a(\mathrm{id})\colon\ a \to a^\circ,$$

$$\mathrm{ex}_a = (\tau a^\circ)^{-1}(\mathrm{id})\colon\ a^\circ \to a.$$

*Proof.* Immediate, by the diagram corresponding to (Diag. $N$), i.e. by naturality. ■

**1.9.** PROPOSITION. *Let $C$ be a* pC *and assume that for each $a \in Ob_C$ there exists the lifting $a^\circ$. Then there is a (unique) extension of the map $a \mapsto a^\circ$ to a functor $-^\circ\colon C \to C_T$* (the *lifting functor*).

*Proof.* *Let* $\mathrm{ex}_a$ be as in 1.8 and $f \in C[a, b]$. Define

$$f^\circ = \tau a^\circ(f \circ \mathrm{ex}_a) \in C_T[a^\circ, b^\circ],$$

where $\tau$ is the natural isomorphism between $\mathbf{Inc} \circ C[-, b]$ and $C_T[-, b^\circ]$. Then one has $\mathrm{id}^\circ = \mathrm{id}$ and, for $f \in C[a, b]$ and $g \in C[b, c]$, $g^\circ \circ f^\circ = (g \circ f)^\circ$, by naturality. ■

EXAMPLES II.1. The lifting functor for **pSet** is obvious. It can be easily guessed also for the category **pPo** of posets and partial monotone functions with upward closed domains: just add a fresh least element and the rest is easy by applying Proposition 1.9. Note that, by monotonicity, the lifting functor does not exist if one does not assume that the domains are upward closed.

2. The category **pCPO** is given by defining complete partial orders

under the assumption that directed sets are not empty. Thus, the objects of **pCPO** do not need to have a least, bottom, element. As morphisms, take the partial continuous functions with open domains following Plotkin [22]. Clearly, the lifting functor is defined as for **pPo**.

EXAMPLE III. Let **pEN** be as in Example I. Given $\boldsymbol{a} = (a, e) \in Ob_{\mathbf{pEN}}$, define $\boldsymbol{a}^\circ = (a^\circ, e^\circ)$ by adding a new element $\perp$ to the set $a$ and by defining

$$e^\circ(n) = \text{if } \varphi_n(0) \text{ converges, then } e(\varphi_n(0)) \text{ else } \perp.$$

Clearly, $e^\circ$: $\omega \to a^\circ$ onto. Let now $\boldsymbol{b} = (b, e')$ and $f \in \mathbf{pEN}[\boldsymbol{b}, \boldsymbol{a}]$, thus there exists $f' \in \mathrm{PR}$ s.t. $f \circ e' = e \circ f'$. We define $f \in \mathrm{EN}[\boldsymbol{b}, \boldsymbol{a}^\circ]$ which extends $f$ by giving $f' \in R$ which represents $f$. That is, set $\varphi_{f'(n)}(0) = f'(n)$. Such an $f' \in R$ exists by the *s-m-n* (iteration) theorem. Then

$$\begin{aligned} f(e'(n)) &= e^\circ(f'(n)) \\ &= \text{if } \varphi_{f'(n)}(0) \text{ conv, then } e(\varphi_{f'(n)}(0)) \text{ else } 1. \end{aligned}$$

Therefore, if $f(e'(n)) = e(f'(n))$ is defined, $f(e'(n)) = e(\varphi_{f'(n)}(0)) = f(e'(n))$. Finally, set $\tau_{\boldsymbol{a}}\boldsymbol{b}(f) = f$. For each $\boldsymbol{a}$, $\tau_{\boldsymbol{a}}$ gives the required natural isomorphism, as $\forall \boldsymbol{g} \in \mathrm{EN}[\boldsymbol{b}, \boldsymbol{a}^\circ]\ \exists! f \in \mathbf{pEN}[\boldsymbol{b}, \boldsymbol{a}], f'(n) = \varphi_{g'(n)}(0)$. By Proposition 1.9, this defines the lifting functor in **pEN**.

**1.10.** DEFINITION (*Complete objects*). Let $C$ be a pC. Then $a \in Ob_C$ is *complete* iff $a < a^\circ$ in $C_T$.

The intuition should be clear. An object is complete when it "already contains", in a sense, the extra $\perp$. Think of an object $d$ of *pCPO* and take its lifting $d^\circ$, i.e., add a least element $\perp$ to $d$. Then $d$ is complete ($d < d^\circ$ via $(i, j)$, say) iff $d$ already contained a least element, $j(\perp)$ to be precise. Obviously, the objects of **CPO** are exactly the complete objects of **pCPO**. 1.13 below characterizes the complete objects in all pC's.

**1.11.** *Remark.* Let $C$ be a pC and $a < a^\circ$ via $(i, j)$ (in $C_T$). Then

$$\exists\, \text{out} \in C_T[a^\circ, a],\ a < a^\circ \text{ via (in, out)},$$

where in is as in 1.8. As for the proof, just set out $= j \circ \tau a^\circ(\text{ex} \circ i \circ \text{ex})$, for ex as in 1.8, and apply the naturality of $\tau$.

The following fact gives the main motivation for the invention of complete objects: exactly on complete objects as targets, all partial morphisms may be extended to total ones, *with the same target.*

**1.12.** DEFINITION. Let $C$ a pC. $f \in C_T[b, a]$ *extends* $f \in C[b, a]$ iff $\forall c \in Ob_C \forall h \in C[c, b]$ ($h$ into $\mathrm{dom}(f)$) $\Rightarrow f \circ h = f \circ h$.

**1.13.** THEOREM. *Let* $C$ *be a* pC *and* $a^\circ$ *be the lifting of* $a \in Ob_C$ *Then*

$$a < a^\circ \Leftrightarrow \forall b\, \forall f \in C[b, a] \exists f \in C_T[b, a]\ f \text{ extends } f.$$

*Proof.* ($\Rightarrow$) Set $\bar{f} = \text{out} \circ \tau b(f)$: $b \to a^\circ \to a$. Then, for $h$ into $\text{dom}(f)$,

$$\begin{aligned} \bar{f} \circ h &= \text{out} \circ \tau b(f) \circ h \\ &= \text{out} \circ \tau t(f \circ h) \\ &= \text{out} \circ \tau a(\text{id}) \circ f \circ h \\ &= \text{out} \circ \text{in} \circ f \circ h \quad \text{by definition of in} \\ &= f \circ h. \end{aligned}$$

($\Leftarrow$) Just take $\mathbf{ex} \in C_T[a^\circ, a]$, for $\text{ex} = (\tau a^\circ)^{-1}(\text{id})$ in 1.8, and note that in is into $\text{dom}(\text{ex})$. ∎

**1.14.** *Remarks.* 1. Note that, if $\bar{f} \in C_T[b, a]$ extends $f \in C[b, a]$, then $\bar{f} = f$ iff $f \in C_T[b, a]$, by 1.6 ($1 \Leftarrow 2$). In particular, if $a < a^\circ$, one has $f = \text{out} \circ \tau b(f)$ iff $f \in C_T[b, a]$. This suggests the definition of a very natural pre-order on each $C[b, a]$:

$f \leqslant g$ iff $\forall c \in Ob_C\ \forall h \in C[c, b]$ ($h$ into $\text{dom}(f)$) $\Rightarrow f \circ h = g \circ h$. Then, in $(C[b, a], \leqslant)$, $O_{ba}$ is the least element and the total arrows are exactly the maximal ones. Moreover, the composition is monotone in each argument w.r.t. "$\leqslant$". (See also Remark 1.22.)

2. $b < a < a^\circ \Rightarrow b < b^\circ$. (Indeed, let $b < a$ via $(i, j)$ and, for any $c \in Ob_C$ and $f \in C[c, b]$, consider the extension $\overline{i \circ f} \in C_T[c, a]$ of $i \circ f \in C[c, a]$. Then $j \circ \overline{i \circ f} \in C_T[c, b]$ extends $f$, since $\forall h$ ($h$ into $\text{dom}(f)$) $j \circ \overline{i \circ f} \circ h = j \circ (i \circ f \circ h) = f \circ h$.) Note also that $a^\circ < a^{\circ\circ}$.

The point with the categorical approach to complete objects is that their properties may be inherited at higher types.

**1.15.** Definition. Let $C$ be a pC such that $C_T$ is a Cartesian Category (CC; see Lambek [1]) with the usual product functor $(\_x\_)_T$: $C_T x C_T \to C_T$ and tupling $\langle\_,\_\rangle$. Then $C$ is a *partial Cartesian Category* (pCC) iff there exists a functor $(\_x\_)$: $CxC \to C$ satisfying:

- $(\_x\_) = \mathbf{Inc} \circ (\_x\_)_T$ on objects;
- for $f \in C[c, a]$ and $g \in C[c, b]$:

$\forall h \forall k$ ($h$ into $\text{dom}(f)$) ($k$ into $\text{dom}(g)$) $fxg \circ \langle h, k \rangle = \langle f \circ h, g \circ k \rangle$

(recall that $h$ and $k$ are total by 1.4 and 1.3.2).

Set also $C[\_xa, b] = C[\_, b] \circ (\_xa) \circ \mathbf{Inc}$.

Note that we only require that $(\_x\_)$ exists with the above properties: the choice of $(\_x\_)$ on morphisms is not unique, as in Hoenke [2]. However, given objects $a$ and $b$, $(axb)$ is uniquely determined.

(1) J. Lambek, *Functional completeness of Cartesian Closed Categories*, Ann. Math. Logic, 6 (1974), 259–292.

(2) H.-J. Hoenke, *On partial algebras*, Universal Algebra, Csakany *et al.* (eds), Coll. Math. Soc. J. Bolyai 29, Amsterdam 1982.

**1.16.** Definition. A pC $\boldsymbol{C}$ is a *partial Cartesian Closed Category* (pCCC) iff $\boldsymbol{C}_T$ is a CC and for all $a$, $b \in Ob_{\boldsymbol{C}}$ there exists a (unique) object $b^a{}_p$ (the *representation of partial morphisms*) and an arrow eval: $b^a{}_p xa \to b$ which is universal from $(-xa)$. Inc to $a$, i.e., such that, for all $c \in Ob_{\boldsymbol{C}}$ and $f \in \boldsymbol{C}[cxa, b]$, the f.d.c.:

$$\begin{tikzcd} cxa \arrow[r, "f"] \arrow[d, dashed, "\lambda(f)xid"'] & b \\ b^a{}_p xa \arrow[ur, "eval"'] & \end{tikzcd}$$

As usual, $\lambda(f)$ is the *carrying* of $f$.

The above condition uniquely defines an adjunction between $\boldsymbol{C}$ and $\boldsymbol{C}_T$. In particular there is a natural isomorphism

$$\lambda\colon\ \boldsymbol{C}[-xa, b] \simeq \boldsymbol{C}_T[-, b^a{}_p].$$

In a pCCC we write $b^a$, if it exist, for the representation of total morphisms. That is, for the (unique, if it exist) object $b^a$ such that there is a natural isomorphism

$$\lambda_T\colon\ \boldsymbol{C}_T[-xa, b] \simeq \boldsymbol{C}_T[-, b^a].$$

As for ordinary CCC's, the naturality of $\lambda_T$ is equivalently expressed by the following diagram, where $g \in \boldsymbol{C}_T[d, c]$:

(Diag. $\lambda_T$)
$$\begin{tikzcd} \boldsymbol{C}_T[cxb,a] \arrow[r, "\lambda_T"] \arrow[d, "\_\circ(gxid)"'] & \boldsymbol{C}_T[c,a^b] \arrow[d, "\_\circ g"] \\ \boldsymbol{C}_T[dxb,a] \arrow[r, "\lambda_T"'] & \boldsymbol{C}_T[d,a^b] \end{tikzcd}$$

That is, $\lambda_T(f)\circ g = \lambda_T(f\circ(gx\,\mathrm{id}))$. The same holds for pCCC's, provided that the equation is restricted to domains. Namely, for $g \in \boldsymbol{C}[d, c]$, one has

(Eq. $\lambda$) $\qquad \forall h\ (h \text{ into } \mathrm{dom}(g))\ \lambda(f)\circ g\circ h = \lambda(f\circ(gx\,\mathrm{id}))\circ h.$

Indeed, (Eq. $\lambda$) is proved as follows:

$$\lambda(f)\circ g\circ h = \lambda\big(f\circ((g\circ h)x\,\mathrm{id})\big) = \lambda(f\circ gx\mathrm{id}\circ hx\,\mathrm{id}) = \lambda(f\circ(gx\,\mathrm{id}))\circ h,$$

as, for $h$ into $\mathrm{dom}(g)$, one has

$$(g\circ h)x\,\mathrm{id} = \langle g\circ h\circ p_1, p_2\rangle = gx\,\mathrm{id}\circ\langle h\circ p_1, p_2\rangle = gx\,\mathrm{id}\circ hx\,\mathrm{id}.$$

Observe that, in 1.16 $\lambda$ goes from $C[-xa, b]$ to $C_T[-, b^a{}_p]$. This may be understood in terms of classical Recursion Theory, which actually inspired Longo–Moggi [15]. Let $\mathbf{PR} = (\mathrm{PR}, \varphi)$ be a goedel-numbering of PR and $\omega = (\omega, \mathrm{id})$. Observe, say, that $\varphi$ is the currying of the universal function $u(n, m) = \varphi(n)(m)$. Thus, the partial binary function $u$ is curried to a total unary function $\varphi$, whose range is a representative of partial morphisms, namely $\omega^\omega{}_p = \mathrm{PR}$. Or, also, for each $f \in \mathbf{pEN}[\omega x \omega, \omega]$ of index $i$, $\lambda(f) \in \mathbf{EN}[\omega, \omega^\omega{}_p]$, where $\lambda(f)(n) = \varphi_{s(i,n)}$, by the $s$-$m$-$n$ theorem. That is, $f$ and $\lambda(f)(n)$ are partial maps, while $\lambda(f)$ is total.

**1.17.** *Remark*. It is easy to extend $\lambda^{-1}$ from $C_T[-, b^a{}_p]$ to $C[-, b^a{}_p]$. For all $c$ and $f \in C[c, b^a{}_p]$, set $\lambda^{-1}(f) = \mathrm{eval} \circ (f x \,\mathrm{id})$. Then

$$C[-xa, b] < C[-, b^a{}_p] \text{ via } (\lambda, \lambda^{-1}).$$

**1.18.** PROPOSITION. *Let $C$ be a* pCCC *and $t$ be the terminal object in $C_T$. Then*

1. $-^t{}_p$: $C \to C$ *is the lifting functor in* $C$,
2. *for all* $a, b \in Ob_C$, $b^a{}_p$ *is a complete object.*

*Proof.* 1. $C[-, a] \circ \mathbf{Inc} \simeq C[-xt, a] \simeq C_T[-, a^t{}_p]$, by 1.15, 1.16.

2. Let $\lambda^{-1}$ be as in 1.17. For any $c \in Ob_C$ and $f \in C[c, b^a{}_p]$, define $\bar{f} = \lambda(\lambda^{-1}(f))$. Then

$$\begin{aligned}\forall h\ (h \text{ into } \mathrm{dom}(f))\ \lambda(\lambda^{-1}(f)) \circ h &= \lambda(\mathrm{eval} \circ (f x\, \mathrm{id})) \circ h \\ &= \lambda(\mathrm{eval}) \circ f \circ h \quad \text{by (Eq. } \lambda) \\ &= f \circ h \quad \text{by definition of eval.}\end{aligned}$$

That is, $\bar{f} \in C_T[c, b^a{}_p]$ extends $f$. ∎

**1.19.** LEMMA. *Let $C$ be a* pCCC *and* $a, b \in Ob_C$. *Assume also that*

1. $b^a$ *exists;*
2. $b < b^\circ$ *via* (in, out) (*i.e.*, *b is complete*).

*Let $\tau$ be the isomorphism* $C[b^a{}_p xa, b] \simeq C_T[b^a{}_p xa, b^\circ]$ *and define* $\mathrm{out}' = \lambda_T(\mathrm{out} \circ \tau(\mathrm{eval}))$. *Then*

$$\forall c \in Ob_C \forall f \in C[cxa, b]\ \mathrm{out}' \circ \lambda(f) = \lambda_T(\mathrm{out} \circ \tau(f)).$$

*Proof.* Clearly,

$$C[b^a{}_p xa, b] \xrightarrow{\tau} C_T[b^a{}_p xa, b^\circ] \xrightarrow{\mathrm{out} \circ -} C_T[b^a{}_p xa, b] \xrightarrow{\lambda_T} C_T[b^a{}_p, b^a].$$

That is, $\mathrm{out}' \in C_T[b^a{}_p, b^a]$. Compute then

$$
\begin{aligned}
\mathrm{out}' \circ \lambda(f) &= \lambda_T(\mathrm{out} \circ \tau(\mathrm{eval})) \circ \lambda(f) \\
&= \lambda_T(\mathrm{out} \circ \tau(\mathrm{eval}) \circ (\lambda(f) \times \mathrm{id})) \quad \text{by (Diag. } \lambda_T) \\
&= \lambda_T(\mathrm{out} \circ \tau.(\mathrm{eval} \circ (\lambda(f) \times \mathrm{id}))) \\
&= \lambda_T(\mathrm{out} \circ \tau(f)). \quad \blacksquare
\end{aligned}
$$

The next theorem "internalizes" the operation which extends each partial morphism to a total one, when the target is a complete object. It even gives a retraction $b^a < b^a{}_p$ in $C_T$.

**1.20.** THEOREM. *Let $C$, $a$, $b$ be as in* 1.19. *Then*

$$b^a < b^a{}_p \textit{ via } (\mathrm{in}', \mathrm{out}'),$$

*where* out′ *is as in* 1.19 *and* $\mathrm{in}' = \lambda(\lambda_T^{-1}(\mathrm{id}_{b^a}))$.

*Proof.* Clearly,

$$C_T[b^a, b^a] \xrightarrow{\lambda_T^{-1}} C_T[b^a \times a, b] \subseteq C[b^a \times a, b] \xrightarrow{\lambda} C_T[b^a, b^a{}_p].$$

Thus $\mathrm{in}' \in C_T[b^a, b^a{}_p]$. Compute then

$$
\begin{aligned}
\mathrm{out}' \circ \mathrm{in}' &= \mathrm{out}' \circ \lambda(\lambda_T^{-1}(\mathrm{id})) \\
&= \lambda_T(\mathrm{out} \circ \tau(\lambda_T^{-1}(\mathrm{id}))) \quad \text{by Lemma 1.19} \\
&= \lambda_T(\lambda_T^{-1}(\mathrm{id})) \quad \text{by Remark 1.14.1} \\
&= \mathrm{id}. \quad \blacksquare
\end{aligned}
$$

**1.21.** COROLLARY. *Let $C$ be a* pCCC. *Assume that $a, b \in Ob_C$ are such that $b^a$ exists. Then*

$$b < b^\circ \Rightarrow b^a < (b^a)^\circ.$$

*Proof.* By 1.14, 1.18 and the theorem. ■

By this, completeness is inherited at higher types, also for total morphisms.

**1.22.** *Remark.* (Following a suggestion of P.L. Curien.) In Obtułowicz [19] partial categories are defined by equipping each hom-set with a partial order and a distinguished set of maximal arrows, called total. The maximal arrows are required to include the identity and to be closed under composition; moreover, composition is assumed to be monotonic in each argument. As pointed out in 1.14.1, we do not need to axiomatize these properties, since they are essentially derived in our approach. We just needed to assume the existence of a "zero" morphism (1.1) and to define the notion of "being into dom$(f)$" (1.4). As for the other notions and results, we think that the merits of our approach partly consist in the use of consolidated notions in Category Theory. For example, lifting functors are defined by classical naturality on total morphisms. However, one could also use the concept of quasi-naturality in Obtułowicz [19], which coincides with naturality on total arrows, and carry on the discussion in a slightly more abstract setting, i.e., with no "zero" morphisms.

**A few remarks on concrete categories.** A suitable notion of "concreteness" may be helpful in the study of categories of functions. Concrete categories are widely used in denotational semantics, where they are defined by a (less general) "enough points" requirement (see below and references in Section 2).

**R.1.** Definition (MacLane [17]). Let $C$ be a category, $t \in Ob_C$ is a *generator* iff for all $a$, $b \in Ob_C$ and all $f$, $g \in C[a, b]$

$$f \neq g \Rightarrow \exists h \in C[t, a]\ f \circ h \neq g \circ h.$$

**R.2.** Definition. A category $C$ is *concrete* iff it has a generator.

A category has *enough points* iff there exists a generator $t$ which is terminal in the given category. In contrast to concreteness, this notion does not extend naturally to pC's, as non-trivial pC's have no terminal objects: in a pC each object $a$ has at least $O_{aa}$ and $\mathrm{id}_{aa}$ as morphisms. Moreover, there is no relevant reason to have a particular generator.

**R.3.** Proposition. *Let* $C$ *be a concrete* pC *and* $t$ *a generator; then* $f \in C[a, b]$ *is total iff* $\forall h \in C[t, a]\ f \circ h = O_{tb} \Rightarrow h = O_{ta}$.

*Proof.* ($\Rightarrow$) Obvious.

($\Leftarrow$) Let $g \in C[c, a]$ be s.t. $f \circ g = O_{cb}$. Thus $\forall h \in C[t, c]\ f \circ g \circ h = O_{tb}$ and, by the assumption, $g \circ h = O_{ta}$. Recall now that $t$ is a generator, then $g = O_{ca}$. ∎

**R.4.** Proposition. *Let* $C$ *be a* pC. *Assume that* $C_T$ *is concrete and let* $t$ *be a generator for* $C_T$. *Then* $f \in C_T[b, a]$ *extends* $f \in C[b, a]$ *iff* $\forall h \in C[t, b]$ (*h into* $\mathrm{dom}(f)$) $\Rightarrow f \circ h = f \circ h$.

*Proof.* ($\Rightarrow$) Obvious.

($\Leftarrow$) Let $g \in C[c, b]$ be into $\mathrm{dom}(f)$. Then $\dot{f} \circ g$ and $f \circ g$ are both total. Suppose now that they are different; then, as $t$ is a generator in $C_T$, there exists $h \in C_T[t, c]$ s.t. $f \circ g \circ h \neq f \circ g \circ h$. This is impossible since $g \circ h \in C[t, b]$ is into $\mathrm{dom}(f)$. ∎

Observe also that both results above do not depend on the choice of the generator $t$.

Finally, consider a pC $C$ such that $C_T$ has enough points. Then, for any $f \in C[t \times a, b]$, $\lambda(f) \in C_T[t, {b^a}_p]$ is the "point" which represents $f$ in ${b^a}_p$. Similarly for $\lambda_T$ w.r.t. $b^a$. Thus, if we identify $C_T[a, b]$ with $b^a$, Lemma 1.19

gives

$$\text{(R.5)} \qquad \mathrm{out}'(f) = \mathrm{out} \circ \tau(f).$$

Therefore $\mathrm{out}'(f)$ is the extension $\bar{f}$ of $f$ in 1.13. This will be used in Section 3.

## 2. Partial retraction systems

**2.1. Definition.** The collection of types $Tp$ over a ground set At of atomic type symbols is inductively defined by:

(i) $\mathrm{At} \subseteq \mathrm{Tp}$;

(ii) if $\sigma$, $\tau \in \mathrm{Tp}$ then $\sigma\tau \in \mathrm{Tp}$.

Given sets $C_i$'s and $C = \{C_i | \ i \in \mathrm{At}\}$, $T_C = \{C_\sigma\}_{\sigma \in \mathrm{Tp}}$ is a *pre-type-structure* over $C$ if $\forall \sigma$, $\tau \in \mathrm{Tp}$ $C_{\sigma\tau} \subseteq C_\sigma \to C_\tau$ $(= \mathbf{Set}[C_\sigma, C_\tau])$, the set of all functions from $C_\sigma$ to $C_\tau$. For $x \in C_{\sigma\tau}$ and $y \in C_\sigma$, we write $xy$ for $x(y)$, if there is no ambiguity.

A *type-structure* $T_A = (\{A_\sigma\}_{\sigma \in \mathrm{Tp}}, [\ ])$ is a pre-type-structure which is a model of typed $\lambda$-calculus. That is, given an $A$-environment $h$: Var $\to \bigcup_{\sigma \in \mathrm{Tp}} \{A_\sigma\}$, one has:

(Var) $[x^\sigma]_h = h(x^\sigma) \in A_\sigma$,

(App) $[M^{\sigma \to \tau} N^\sigma]_h = [M^{\sigma \to \tau}]_h([N^\sigma]_h) \in A_\tau$,

($\beta$) $[\lambda x^\sigma . M^\tau]_h(a) = [M^\tau]_{h[a/x^\sigma]} \in A_\tau$ for $a \in A_\sigma$.

Note that, in a type-structure, axiom ($\eta$) and rule ($\varepsilon$) (i.e., $\lambda y . My = M$, for $y$ not free in $M$, and $M = N \Rightarrow \lambda x . M = \lambda x . N$, respectively, with the intended types) are always realized, since one has:

($\eta$) $\forall a\, [\lambda y^\sigma . M^{\sigma \to \tau} y^\sigma]_h(a) = [M^{\sigma \to \tau}]_h(a)$ and

($\varepsilon$) $\forall a\, [M^\tau]_{h[a/x^\sigma]} = [N^\tau]_{h[a/x^\sigma]} \Rightarrow [\lambda x^\sigma . M^\tau]_h = [\lambda x^\sigma . N^\tau]_h$, by ($\beta$) and by $[\lambda x^\sigma . p^\tau]_h \in A_{\sigma\tau} \subseteq A_\sigma \to A_\tau$.

(*Notation*: we may omit types, if there is no ambiguity; given $T_A = (\{A_\sigma\}_{\sigma \in \mathrm{Tp}}, [\ ])$, $T_A \models M = N$ means that, for all $A$-environment $h$, $[M]_h = [N]_h$.)

Thus a pre-type-structure $T_A$ is a type-structure iff

$$\lambda\beta\eta \vdash M = N \Rightarrow T_A \models M = N.$$

In particular, one obtains type-structures from "concrete" categories, i.e., from categories whose objects and morphisms may be viewed as sets and functions "in extenso" (see R.1–R.6 in Section 1), provided that they are closed under formation of morphisms spaces. More formally, let $K$ be a Cartesian Closed Category (CCC) and $C = \{C_i | \ i \in \mathrm{At}\} \subseteq Ob_K$. Then $K_C = \{C_\sigma | \ \sigma \in \mathrm{Tp}\}$ is the type-structure generated by $C$ in $K$, that is each $C_{\sigma\tau}$ coincides with $C_\tau^{C_\sigma}$, the representative of the collection $K[C_\sigma, C_\tau]$ of morphisms from $C_\sigma$ to $C_\tau$ (if needed and not ambiguous, we may identify

$C_\tau^{C_\sigma}$ and $K[C_\sigma, C_\tau]$). By the well-known results relating CCC's and typed $\lambda$-calculus (see Lambek [13], Scott [27], Dibjer [4], Poigne [23]), $K_C$ is indeed a type-structure. The equivalence between the assumption on "concreteness" (more precisely, "enough points") and rule ($\varepsilon$) is tidely investigated in Berry [2] and Poigne [23], for the typed case, and Koymans in 1982, and Barendregt [1] for the type-free calculus.

2.2. EXAMPLES. The simplest type-structures are the full type-structures and the term model of typed $\lambda\beta\eta$. That is $\mathbf{Set}_A$, where $A$ is a collection of sets, and $\mathbf{Term} = (\{\mathrm{Term}_\sigma\}_{\sigma\in\mathrm{Tp}}, [\ ])$, where $\mathrm{Term}_\sigma$ is the set of terms of type $\sigma$ modulo $\beta\eta$ convertibility. (If there is no ambiguity, we identify $M^\sigma$ and $[M^\sigma] = \{N^\sigma \mid \lambda\beta\eta \vdash M^\sigma = N^\sigma\}$.)

Given a CCC $K$, there is a canonical way to inherit retractions (see before 1.8) at higher types; namely, if $B_\sigma < A_\sigma$ via $(\alpha_\sigma, \beta_\sigma)$ and $B_\tau < A_\tau$ via $(\alpha_\tau, \beta_\tau)$, then $B_{\sigma\tau} < A_{\sigma\tau}$ via

$$\alpha_{\sigma\tau}(x) = \alpha_\tau \circ x \circ \beta_\sigma, \qquad \beta_{\sigma\tau}(y) = \beta_\tau \circ y \circ \alpha_\sigma. \tag{2.3}$$

Clearly, $\alpha_{\sigma\tau} \in K[A_{\sigma\tau}, B_{\sigma\tau}]$, $\beta_{\sigma\tau} \in K[B_{\sigma\tau}, A_{\sigma\tau}]$ and $\alpha_{\sigma\tau} \circ \beta_{\sigma\tau} = \mathrm{id}$.

Retractions play a major role in the semantic investigation of type theory, as they provide a strong and precise notion of "subtype". A simple result in this section will be the following (see Corollary 2.15):

*Let $A = \{A_i\}_{i\in\mathrm{At}}$ and $B = \{B_i\}_{i\in\mathrm{At}}$ be collections of objects in a* CCC $K$; *if* $\forall i \in \mathrm{At}$ $B_i < A_i$, *then*

$$K_A \models M = N \Rightarrow K_B \models M = N.$$

This fact gives a strong consequence in any type, by some information on the ground types. However, it uses the assumption that both type-structures are built in the same category. That is, for all $\sigma$, $\tau \in \mathrm{Tp}$, both $A_{\sigma\tau}$ and $B_{\sigma\tau}$ are exactly the morphisms in $K$ of the intended type. 2.12 and 2.14 will prove more by comparing type-structures which do not need to satisfy this strong assumption. That comparison will be made possible by an essential use of partial morphisms.

How does partiality come in? Given type-structures $T_A$ and $T_B$, assume that, for all $\sigma \in \mathrm{Tp}$, $A_\sigma$ and $B_\sigma$ are objects of a CCC $K$. Very roughly, the idea is to take the category $K$ where the type structure with "more morphisms", $T_A$ say, is built in (or just the category $\mathbf{Set}$). Then, even if $A_{\sigma\tau} = A_\tau^{A_\sigma}$, $B_{\sigma\tau}$ may be smaller than $B_\tau^{B_\sigma}$, for some $\sigma$, $\tau \in \mathrm{Tp}$. Thus, for $x \in A_{\sigma\tau}$, $\alpha_{\sigma\tau}(x) = \alpha_\tau \circ x \circ \beta_\sigma$ does not need to be in $B_{\sigma\tau}$, that is $\alpha_{\sigma\tau}\colon A_{\sigma\tau} \to B_{\sigma\tau}$ does not need to be defined on $x$.

For the purposes of this section we only need the classical notion of partiality for set-theoretic functions. In particular, we write $B <_p A$ if there exist partial functions $\alpha\colon A \to B$ and $\beta\colon B \to A$ s.t. $\alpha \circ \beta = \mathrm{id}$. Clearly, then, $\alpha$ is a (possibly partial) surjection and $\beta$ a total injection.

**2.4. Definition.** Let $\{A_\sigma\}_{\sigma\in\mathrm{Tp}}$, $\{B_\sigma\}_{\sigma\in\mathrm{Tp}}$ be pre-type-structures. Then $\{(\alpha_\sigma, \beta_\sigma)\}_{\sigma\in\mathrm{Tp}}$ is a *partial retraction system* (p.r.s.) from $\{A_\sigma\}_{\sigma\in\mathrm{Tp}}$ onto $\{B_\sigma\}_{\sigma\in\mathrm{Tp}}$ if $\forall i\in\mathrm{At}\ B_i <_p A_i$ via partial functions $(\alpha_i, \beta_i)$ and

cond. (1): $\forall x\in A_{\sigma\tau}\,[\alpha_\tau\circ x\circ\beta_\sigma\in B_{\sigma\tau}\Rightarrow\alpha_{\sigma\tau}(x)=\alpha_\tau\circ x\circ\beta_\sigma]$,
cond. (2): $\forall z\in B_{\sigma\tau}\,\forall y\in\mathrm{dom}\,\alpha_\sigma\,(\beta_{\sigma\tau}(z))y=\beta_\tau(z(\alpha_\sigma(y)))$.

**2.5. Proposition.** *Let* $\{A_\sigma\}_{\sigma\in\mathrm{Tp}}$, $\{B_\sigma\}_{\sigma\in\mathrm{Tp}}$ *be pre-type-structures and let* $\{(\alpha_\sigma, \beta_\sigma)\}_{\sigma\in\mathrm{Tp}}$ *be a p.r.s. from* $\{A_\sigma\}_{\sigma\in\mathrm{Tp}}$ *onto* $\{B_\sigma\}_{\sigma\in\mathrm{Tp}}$. *Then* $\forall\sigma,\tau\in\mathrm{Tp}$ *one has*:

(i) $\alpha_\sigma\circ\beta_\sigma=\mathrm{id}$,
(ii) $(\beta_{\sigma\tau}(z))(\beta_\sigma(y))=\beta_\tau(zy)$.

*Proof.* If $\sigma\in\mathrm{At}$, (i) holds, by definition of p.r.s.. Suppose then that (i) holds for $\sigma$, $\tau$. One then has, for all $z\in B_{\sigma\tau}$ and $y\in B_\sigma$:

$$\begin{aligned}(\beta_{\sigma\tau}(z))(\beta_\sigma(y)) &= \beta_\tau(z(\alpha_\sigma(\beta_\sigma(y)))) && \text{by cond. (2) and induction on (i)}\\ &= \beta_\tau(zy) && \text{by induction on (i).}\end{aligned}$$

This is (ii) and, thus, for all $y\in B_\sigma$:

$$\begin{aligned}\alpha_\tau((\beta_{\sigma\tau}(z))(\beta_\sigma(y))) &= \alpha_\tau(\beta_\tau(zy))\\ &= zy \quad \text{by induction on (i).}\end{aligned}$$

Observe now that $\{B_\sigma\}_{\sigma\in\mathrm{Tp}}$ is extensional and then

$$\alpha_\tau\circ(\beta_{\sigma\tau}(z))\circ\beta_\sigma=z.$$

Thus cond. (1) applies and gives $\forall z\in B_{\sigma\tau}\ \alpha_{\sigma\tau}(\beta_{\sigma\tau}(z))=z$ or, equivalently, (i) at the higher type. ■

By (i) in the proposition, for each $\sigma\in\mathrm{Tp}$, $B_\sigma <_p A_\sigma$. Moreover, by (ii), the injection $\beta_\sigma$ is a total "homomorphism", i.e. it preserves functional application. As this is a fundamental notion, we briefly survey the connections between p.r.s., homomorphisms and logical relations.

Let $T_A=\{A_\sigma\}_{\sigma\in\mathrm{Tp}}$ and $T_B=\{B_\sigma\}_{\sigma\in\mathrm{Tp}}$ be a pre-type-structure and $R_i\subseteq A_i x B_i$, $i\in\mathrm{At}$. Define then a *logical relation* $\{R_\sigma\}_{\sigma\in\mathrm{Tp}}$, with $R_{\sigma\tau}\subseteq A_{\sigma\tau}xB_{\sigma\tau}$, by

$$R_{\sigma\tau}(a,b)\Leftrightarrow\forall x,y(R_\sigma(x,y)\Rightarrow R_\tau(a(x),b(y))).$$

**2.6. Proposition.** *Let* $\{R_\sigma\}_{\sigma\in\mathrm{Tp}}$ *be a logical relation.*

(1) *If* $\forall i\in\mathrm{At}\,R_i$ *is single valued and* $\forall\sigma\,R_\sigma$ *is surjective, then* $\forall\sigma\,R_\sigma$ *is single valued.*

(2) *If* $\forall\sigma\,R_\sigma$ *is single valued, set* $\alpha^R{}_\sigma(a)=b$ *iff* $R_\sigma(a,b)$. *Then* $\{\alpha^R{}_\sigma\}_{\sigma\in\mathrm{Tp}}$ *is an homomorphism, i.e.,* $\forall\sigma,\tau\ \alpha^R{}_{\sigma\tau}(a)(\alpha^R{}_\sigma(c))=\alpha^R{}_\tau(a(c))$. *Conversely, each surjective homomorphism* $\{\alpha'_\sigma\}_{\sigma\in\mathrm{Tp}}$ *defines a logical relation* $\{R'_\sigma\}_{\sigma\in\mathrm{Tp}}$ *such that* $\forall\sigma\ R'_\sigma(a,b)$ *iff* $\alpha'_\sigma(a)=b$.

(The *proof* is easy.)

Recall that also the $\beta_\sigma$'s in 2.5 yield an homomorphism. The way $\beta_\sigma$'s and the $\alpha^R{}_\sigma$'s relate is expressed by 2.7–2.8. Assume that a p.r.s. $\{\alpha_\sigma, \beta_\sigma\}_{\sigma\in\mathrm{Tp}}$ is given from a pre-type-structure $\{A_\sigma\}_{\sigma\in\mathrm{Tp}}$ onto $\{B_\sigma\}_{\sigma\in\mathrm{Tp}}$. Define a logical relation $\{R_\sigma\}_{\sigma\in\mathrm{Tp}}$ as above over $R_i$, where $R_i(a, b)$ iff $\alpha_i(a) = b$. Then one has:

2.7. THEOREM. $\forall\sigma\in\mathrm{Tp}$ (1) $\forall b\in B_\sigma$ $R_\sigma(\beta_\sigma(b), b)$, (2) $\forall a\in A_\sigma$ $\forall b\in B_\sigma$ $R_\sigma(a, b)\Rightarrow\alpha_\sigma(a) = b$.

*Proof.* (By combined induction on types.) $\sigma\in\mathrm{At}$: O.K.

(1) $\sigma = \gamma\to\delta$: Let $R_\gamma(a, d)$. Then $\alpha_\gamma(a) = d$ by induction on (2). Moreover, $a\in\mathrm{dom}\,\alpha_\gamma$ and, then, $\beta_\sigma(b)(a) = \beta_\delta(b(\alpha_\gamma(a))) = \beta_\delta(b(d))$. Thus $R_\delta(\beta_\sigma(b)(a), b(d))$, since $R_\delta(\beta_\delta(b(d)), b(d))$ by induction on (1).

| | | |
|---|---|---|
| (2) | $\sigma = \gamma\to\delta: R_\gamma(\beta_\gamma(c), c)$ | for all $c\in B_\gamma$, by induction on (1) |
| | $\Rightarrow R_\delta(a(\beta_\gamma(c)), b(c))$ | by $R_\sigma(a, b)$ |
| | $\Rightarrow\alpha_\delta(a(\beta_\gamma(c))) = b(c)$ | by induction on (2) |
| | $\Rightarrow\alpha_\delta\circ a\circ\beta_\gamma = b\in B_\sigma$ | by extensionality |
| | $\Rightarrow\alpha_\sigma(a) = b$. ∎ | |

Thus any p.r.s. $\{\alpha_\sigma, \beta_\sigma\}_{\sigma\in\mathrm{Tp}}$ gives a logical relation $\{R_\sigma\}_{\sigma\in\mathrm{Tp}}$ which is surjective (2.7.1) and single valued (2.7.2) and, hence, a surjective homomorphism ($\alpha^R{}_\sigma$ in 2.6.2). In general, $\alpha^R{}_\sigma$ and $\alpha_\sigma$ are partial maps:

2.8. COROLLARY. $\forall\sigma\in\mathrm{Tp}$ *range* $\beta_\sigma\subseteq\mathrm{dom}\,R_\sigma = \mathrm{dom}\,\alpha^R{}_\sigma\subseteq\mathrm{dom}\,\alpha_\sigma$. *Moreover,* $\forall a\in\mathrm{dom}\,\alpha^R{}_\sigma$ $\alpha_\sigma(a) = \alpha^R{}_\sigma(a)$ *and, hence,* $\alpha^R{}_\sigma\circ\beta_\sigma = \mathrm{id}$.

As it will be pointed out in 2.10 and 2.12, the existence of a surjective homomorphism between two type-structures has several consequences. The first, 2.10, has been recently communicated by Statman to the authors.

2.9. DEFINITION. Let $T_A$ be a type-structure. $f\in A_{\sigma\tau}$ is *n-piecewise-λ-definable* iff $\forall a_1, \ldots, a_n\in A_\sigma\exists M\in\mathrm{Term}_{\sigma\tau}$ (closed) $f(a_i) = [M]_h(a_i)$ for $1\leqslant i\leqslant n$.

2.10. PROPOSITION. *Let* $T_A$ *be a type-structure such that there exists a surjective homomorphism from* $T_A$ *onto* **Term** *and let* $f\in A_{\sigma\tau}$ *be 2-piecewise-λ-definable. Then* $f$ *is λ-definable.*

2.11. FUNDAMENTAL THEOREM ON LOGICAL RELATIONS (Statman [32]). *Let* $T_A$ *and* $T_B$ *be type structures and* $\{R_\sigma\}_{\sigma\in\mathrm{Tp}}$ *a logical relation. Then*

$$\forall\sigma\,\forall h_A, h_B(\forall x^\sigma R_\sigma(h_A(x^\sigma), h_B(x^\sigma)))\Rightarrow\forall M\, R_\sigma([M]_{hA}, [M]_{hB}).$$

2.12. COROLLARY (Friedman [7]). *If* $\{R_\sigma\}_{\sigma\in\mathrm{Tp}}$ *is a surjective and single valued logical relation (a surjective homomorphism) from* $T_A$ *onto* $T_B$, *then* $T_A\models M = N\Rightarrow T_B\models M = N$.

The main difficulty with logical relations is to construct singled valued and surjective ones, i.e., to find surjective homomorphisms. P.r.s.' may be viewed as a tool for defining them in a way which is constructive w.r.t. the intended category. Indeed, this is how p.r.s.'s will be used in Section 3.

We give a proof of the relation between type-structures in 2.12 by a direct use of p.r.s.'. Recall that all typed terms possess a normal form; moreover, if $M$ is in normal form and $M \equiv PQ$, then, for no $P'$, $P \equiv \lambda x.P'$.

**2.13.** LEMMA. *Let* $T_A = (\{A_\sigma\}_{\sigma\in \mathrm{Tp}}, [\ ])$, $T_B = (\{B_\sigma\}_{\sigma\in \mathrm{Tp}}, [\ ]')$ *be type-structures and* $\{(\alpha_\sigma, \beta_\sigma)\}_{\sigma\in \mathrm{Tp}}$ *a p.r.s. from* $T_A$ *onto* $T_B$. *For each B-environment* $h$ *define an A-environment* $h'$ *by*

$$\forall \sigma \in \mathrm{Tp}\, \forall x^\sigma\ h'(x^\sigma) = \beta_\sigma(h(x^\sigma)).$$

*Then one has*:

(i) *if* $M$, *of type* $\sigma$, *is in* n.f.,

$$M \equiv x \text{ or } M \equiv PQ \Rightarrow [M]_{h'} = \beta_\sigma([M]'_h),$$

$$M \equiv \lambda x.P \Rightarrow \alpha_\sigma([M^\sigma]_{h'}) = [M^\sigma]'_h;$$

(ii) *for all* $M$, *of type* $\sigma$, $\alpha_\sigma([M]_{h'}) = ([M]'_h)$.

*Proof*. (i) By induction on the structure of $M$.

$M \equiv x^\sigma$: $[x]_{h'} = h'(x) = \beta_\sigma(h(x)) = \beta_\sigma([x]'_h)$,

$M \equiv PQ$: by the remark above, one has $P = x$ or $P \equiv RS$.

By the inductive hypothesis, $[P]_{h'} = \beta_{\sigma\tau}([P]'_h)$ and $\alpha_\sigma([Q]_{h'}) = [Q]'_h$. Then

| | |
|---|---|
| $[PQ]_{h'} = ([P]_{h'})([Q]_{h'})$ | by definition of [ ] |
| $= (\beta_{\sigma\tau}([P]'_h))([Q]_{h'})$ | by induction |
| $= \beta_\tau([P]'_h(\alpha_\sigma([Q]_{h'})))$ | by cond. (2) |
| $= \beta_\tau([P]'_h([Q]'_h))$ | by induction |
| $= \beta_\tau([PQ]'_h)$ | by definition of [ ]'. |

$M \equiv \lambda x^\sigma.P^\tau$: for all $y \in B_\sigma$

| | |
|---|---|
| $\alpha_\tau([\lambda x.P]_{h'}(\beta_\sigma(y))) = \alpha_\tau([P]_{h'}[\beta_\sigma(y)/x])$ | by definition of [ ] |
| $= [P]'_h[y/x]$ | by induction |
| $= [\lambda x.P]'_h(y)$ | by definition of [ ]'. |

Thus by ' extensionality:

$$\alpha_\tau \circ [\lambda x.P]_{h'} \circ \beta_\sigma = [\lambda x.P]'_h$$

and by cond. (1):

$$\alpha_{\sigma\tau}([\lambda x.P]_{h'}) = [\lambda x.P]'_h.$$

(ii) Just observe that each $\lambda$-term $M$ has a n.f. $M'$, and, as $\boldsymbol{T_A}$ and $\boldsymbol{T_B}$ are models,

$$\begin{aligned}\alpha_\sigma([M]_{h'}) &= \alpha_\sigma([M']_{h'})\\ &= [M']'_h \qquad \text{by (i) above and (i) in 2.5}\\ &= [M]'_h. \ \blacksquare\end{aligned}$$

**2.14.** THEOREM. *Let $\boldsymbol{T_A}$ and $\boldsymbol{T_B}$ be type-structures and suppose that there exists a p.r.s. from $\boldsymbol{T_A}$ onto $\boldsymbol{T_B}$. Then $\boldsymbol{T_A} \models M = N \Rightarrow \boldsymbol{T_B} \models M = N$.*

*Proof.* For each $B$-environment $h$ define an $A$-environment $h'$ by $\forall \sigma \in \mathrm{Tp}\ \ \forall x^\sigma\, h'(x^\sigma) = \beta_\sigma(h(x^\sigma))$. If $\boldsymbol{T_A} \models M = N$ then $[M]'_h = \alpha_\sigma([M]_{h'}) = \alpha_\sigma([N]_{h'}) = [N]'_h$. Hence $\boldsymbol{T_B} \models M = N$. ∎

**2.15.** COROLLARY. *Let $A = \{A_i\}_{i \in \mathrm{At}}$ and $B = \{B_i\}_{i \in \mathrm{At}}$ be collections of objects in a* CCC $\boldsymbol{K}$; *if $\forall i \in \mathrm{At}\ B_i < A_i$, then*

$$\boldsymbol{K_A} \models M = N \Rightarrow \boldsymbol{K_B} \models M = N.$$

*Proof.* By 2.3. ∎

**2.16.** COROLLARY. (Completeness, Friedman [7].) *Let $A = \{A_i |\ i \in \mathrm{At}\}$ be a collection of infinite sets. Then*

$$\mathrm{Set}_A \models M = N \Leftrightarrow \lambda\beta\eta \vdash M = N.$$

*Proof.* We only need to prove ($\Rightarrow$). Clearly, for any $i \in \mathrm{At}$, $\mathrm{Term}_i < A_i$ via some $(\alpha_i, \beta_i)$ in **Set**. Define then a p.r.s. $\{(\alpha_\sigma, \beta_\sigma)\}_{\sigma \in \mathrm{Tp}}$, in **pSet**, by using cond. (1) and cond. (2) in 2.4 in the obvious way, with $B_\sigma = \mathrm{Term}_\sigma$. ∎

Note that, in 2.16, $\alpha_\sigma$, $\beta_\sigma$ are well defined as partial functions, for all $\sigma \in \mathrm{Tp}$; indeed, this is all what we need, as **pSet**, with the obvious partial morphisms, is used as "frame" category (by frame category we intend the category whose objects include the types of the considered type-structures). Also Plotkin [21], which was brought to our attention when this paper was in preparation, uses a particular notion of partial retractions in our sense, for the special case of cpo's, in the frame of **pSet**. More generally, a notion of partiality in arbitrary categories is needed, as presented in Section 1. By this it will be possible to apply theorem 2.14 to type-structures of more structured objects (numbered sets, domains, cpo's...). Of course, the definition of a p.r.s. cannot be given, in general, as simply as in 2.16 for **pSet** and p.r.s.'s must be constructed "effectively", in a sense which is relative to the intended frame category.

*Remark.* A recent result of Statman, the "1-section theorem", fully characterizes "complete" type-structures, in the sense of 2.16, by a necessary and sufficient condition. Apparently, the 1-section theorem may be recovered by digging deeply into Statman [31], [33]. We state it here in its great elegance as we could borrow it from Types [34]. This simple statement uses models with a unique atomic type, $D_0$.

THEOREM (1-section). *A type-structure $T_D$ is complete iff for any two algebraic closed terms s and t, constructed from a constant c and a binary function symbol f there exist*:

*An interpretation of c in $D_0$,*
*An interpretation of f in $D_{0\to(0\to 0)}$,*

*such that the resulting interpretations of s and t in $D_0$ are distinct.*

Note though that the existence of a p.r.s. and, hence, of a surjective homomorphism from a type-structure onto **Term** is a stronger property which gives some important extra information (e.g. 2.10).

## 3. Types as quotient sets

The data types one is usually dealing in effective computations are countable sets, possibly structured by an order or similar relations. Indeed, since its origin, denotational semantics was based on the idea of interpreting (higher type) computations by countable approximations of (possibly infinite) processes. Thus even uncountable sets for the interpretation of formal types have a countable and effective core. This is the leading idea for the various categories of Scott's domains ($\omega$-algebraic cpo's, effectively given domains...).

Countable or, more precisely, numbered sets may be viewed as quotients over the set $\omega$ of natural numbers. That is, each $A = (A, e_A)$, where $e_A: \omega \to A$ is an onto map (numbering), defines an equivalence relation on $\omega$ by $n A m$ iff $e_A(n) = e_A(m)$ (and conversely). Thus each element of $A$ corresponds exactly to an equivalence class in $\omega$ and we may view at Malcev–Ershov category of numbered sets as the category $\mathbf{ER}_\omega$ of equivalence relations on $\omega$, whose morphisms are defined as follows. Let $(P)R$ be the set of (partial) recursive functions. Then $f \in \mathbf{ER}_\omega[A, B]$ iff there exists $f' \in R$ s.t. the following diagram commutes:

$$\begin{tikzcd} \omega \arrow[r, "f'"] \arrow[d, "e_A"'] & \omega \arrow[d, "e_B"] \\ A \arrow[r, "f"'] & B \end{tikzcd}$$

$\mathbf{ER}_\omega$ is a cartesian category (products are obvious), but not a CCC. It contains, however, several interesing full subCCC, such as Scott's constructive domains (see later) and a lot of higher type recursion theory may be carried on within it, see Ershov [6] (or Longo–Moggi [14]).

Clearly, given numbered sets $A$ and $B$, not all $f' \in R$ induces an $f \in \mathbf{ER}_\omega[A, B]$, as $f'$ must preserve $A$-equivalences, that is $nAm \Rightarrow f'(n)\,\boldsymbol{B}f'(m)$. This suggests a way to introduce higher type objects and thus to define a cartesian closed extension of $\mathbf{ER}_\omega$. Let $\{\varphi_i\}_{i\in\omega}$ be any acceptable goedel-numbering of $PR$. Define then

$$\text{(Quot.)} \qquad pB^A q \quad \text{iff} \quad nAm \Rightarrow \varphi_p(n)\,B\varphi_q(m).$$

$A^B$ is a partial equivalence relation on $\omega$, as it is defined on a subset of $\omega$. Indeed, $\mathrm{dom}(B^A) = \{p \mid pB^A p\} \subseteq \omega$, and a partial numbering $\pi_{AB}$: $\mathrm{dom}(B^A) \to A^B$ is given by $\pi_{AB}(n) = \{m/nB^A m\}$. In general, each partial surjective $\pi$: $\omega \to C$ uniquely defines a partial equivalence relation (and conversely).

3.1. Definition. The category $\mathbf{PER}_\omega$ of partial equivalence relations on $\omega$ has as objects the subsets of $\omega$ modulo an equivalence relation. Given objects $\boldsymbol{a} = (a, \pi_a)$ and $\boldsymbol{b} = (b, \pi_b)$, where $\pi_a$, $\pi_b$ are partial numberings, $f \in \mathbf{PER}_\omega[\boldsymbol{a}, \boldsymbol{b}]$ iff there exists $f' \in PR$ s.t. the f.d.c., where $\mathrm{dom}(\pi_x) = \omega_x$:

$$\begin{tikzcd}
\omega_a \arrow[r, "f'"] \arrow[d, "\pi_a"'] & \omega_b \arrow[d, "\pi_b"] \\
a \arrow[r, "f"'] & b
\end{tikzcd}$$

Clearly, $\mathbf{ER}_\omega$ is a full subcategory of $\mathbf{PER}_\omega$. Moreover, $\mathbf{PER}_\omega$ is a CCC; it is actually a full subCCC of the "effective topos" (see Hyland [11]), a tool widely used for the semantics of intuitionistic logic because of its constructive nature. Note that the representative $\boldsymbol{b}^{\boldsymbol{a}}$ of $\mathbf{PER}_\omega[\boldsymbol{a}, \boldsymbol{b}]$ is partially enumerated by the quotient subset of $\omega$ determined by the partial relation $\boldsymbol{b}^{\boldsymbol{a}}$ (see (Quot.) above). That is, $\pi_{ab}(i) = f$ iff $f \circ \pi_a = \pi_b \circ \varphi_i$.

In Computer Science, $\mathbf{PER}_\omega$ is also known as the quotient set semantics of types over $\omega$, following the ideas in Scott [26] on $\lambda$-models (see Longo–Moggi [16] for details and further work on arbitrary (partial) combinatory algebras).

Classical computability suggests now a natural way to extend $\mathbf{PER}_\omega$ to a category with partial morphisms. Note that, by definition, if $p\boldsymbol{b}^{\boldsymbol{a}}p$ then $\varphi_p$ is a (partial) recursive function which is total on $\mathrm{dom}(\boldsymbol{a})$, as we are defining an ordinary category with total morphisms. Just drop this condition and define $\mathbf{pPER}_\omega$ exactly as $\mathbf{PER}_\omega$ by allowing $f \in \mathbf{pPER}_\omega[\boldsymbol{a}, \boldsymbol{b}]$ to be partial. More formally,

$$f \in \mathbf{pPER}_\omega[\boldsymbol{a}, \boldsymbol{b}] \text{ iff } \exists f' \in PR\ (n\boldsymbol{a}m \textbf{ and } f'(n)\downarrow \Rightarrow f'(m)\downarrow)$$
$$\textbf{and } f \circ \pi_a(n) = \pi_b \circ f'(n).$$

By checking the condition in 1.16 one may actually prove:

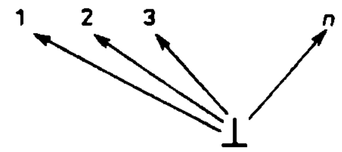
**3.2.** THEOREM. **pPER**$_\omega$ *is a* pCCC.

*Clearly*, $(\mathbf{pPER}_\omega)_T = \mathbf{PER}_\omega$.

(*Notation.* 1. We will keep writing $f'$, $M'$, $\alpha'$,... for the functions in $PR$ defining the morphism $f$, $M$, $\alpha$ ... in (**p**)**PER**$_\omega$.

2. Set $\pi_{abp}$ for the enumeration of $(b^a)_p$.)

**pPER**$_\omega$ is the "effective" frame category which we are going to use. It will serve as a tool for comparing type-structures of quotient sets within **PER**$_\omega$ to the typed term model **Term**. As an immediate consequence, this will give us full information on p.r.s. onto **Term** from type-structures built out of usual "flat cpo's" of data as ground types

e.g. 


Observe first that **Term** $= (\{\mathrm{Term}_\sigma\}_{\sigma\in\mathrm{Tp}}, [\ \ ])$ is a collection of countable sets. Indeed each $\mathrm{Term}_\sigma$ can be numbered by an injective $e_\sigma: \omega \to \mathrm{Term}_\sigma$. Just code all terms in normal form and set $e_\sigma(i) = [N]$ if $i$ is the code of the $\beta\eta$-normal form of $N$. Thus we may view $\mathrm{Term}_\sigma$ as an object $(\mathrm{Term}_\sigma, e_\sigma)$ in $\mathbf{ER}_\omega$ and, hence, in $\mathbf{PER}_\omega$.

Moreover, $\mathrm{Term}_{\sigma\tau} \subseteq \mathbf{PER}_\omega[\mathrm{Term}_\sigma, \mathrm{Term}_\tau]$, since for each $M \in \mathrm{Term}_{\sigma\tau}$ there exists $M' \in R$ s.t. the f.d.c.:

$$\begin{tikzcd}
\omega \arrow[r, "M'"] \arrow[d, "e_\sigma"'] & \omega \arrow[d, "e_\tau"] \\
\mathrm{Term}_\sigma \arrow[r, "M"'] & \mathrm{Term}_\tau
\end{tikzcd}$$

Clearly, $M'$ depends uniformely effectively on $M$. That is,

$$\exists l \in R \ \ e_{\sigma\tau}(i) = M \Rightarrow \varphi_{l(i)} = M'. \tag{3.2.1}$$

This will be used in Theorem 3.4, jointly with the "inverse" property. The later property is formalized in 3.3 and gives some new information on typed terms and their relation to computable functions.

**3.3.** MAIN LEMMA. *There exist a partial recursive function $f$ such that for all $i \in \omega$, if there is a term $M^{\sigma\tau}$ for which one has $M^{\sigma\tau} \circ e_\sigma = e_\tau \circ \varphi_i$, then $M^{\sigma\tau} = e_{\sigma\tau}(f(i))$.*

*That is, for all $i \in \omega$ we can uniformely effectively find (an index for) $M$ such that the following diagram commutes, if $M$ exists*:

(Diag. $M$)

$$\begin{tikzcd} \omega \arrow[r, "\varphi_i"] \arrow[d, "e_\sigma"'] & \omega \arrow[d, "e_\tau"] \\ \mathrm{Term}_\sigma \arrow[r, "M"'] & \mathrm{Term}_\tau \end{tikzcd}$$

*Proof.* Clearly, if (Diag. $M$) commutes, then

(1) $\exists M \in \mathrm{Term}_{\sigma\tau}\ e_\tau(\varphi_i(e_\sigma^{-1}(x^\sigma))) = Mx^\sigma$ for $x^\sigma$ not in $FV(M)$.

*Claim.* Let $M^{\sigma\tau}$ be in *n.f.*, and $x^\sigma \neq y^\sigma$ two variables of type $\sigma$.

$$Mx^\sigma[y^\sigma/x^\sigma] = My^\sigma \Leftrightarrow x^\sigma \quad \text{is not in } FV(M).$$

*Proof.* $M \equiv zP$: $(zPx^\sigma)[y^\sigma/x^\sigma] = (zP)[y^\sigma/x^\sigma]y^\sigma$, and this is equal to $My^\sigma$ if and only if $x^\sigma$ is not in $FV(M)$,

$$M \equiv \lambda y v.zP\colon\ (\lambda y v.zP)x^\sigma[y^\sigma/x^\sigma] = (\lambda y v.zP)y^\sigma[y^\sigma/x^\sigma]$$

and again this is equal to $My^\sigma$ if and only if $x^\sigma$ is not free in $M$. This concludes the proof of the claim.

By definition, for all $N \in \mathrm{Term}_\sigma$, $e_\tau(\varphi_i(e_\sigma^{-1}(N))) \in \mathrm{Term}_\tau$. Then the algorithm which defines $f$ goes as follows: given $i$, find $x^\sigma$ and $y^\sigma$, if any, such that

(2) $$e_\tau(\varphi_i(e_\sigma^{-1}(x^\sigma)))[y^\sigma/x^\sigma] = e_\tau(\varphi_i(e_\sigma^{-1}(y^\sigma))).$$

If and when the variables in (2) are found, set $f(i)$ equal to the $e_{\sigma\tau}$-number of $\lambda x^\sigma \cdot e_\tau(\varphi_i(e_\sigma^{-1}(x^\sigma)))$, that is

$$e_{\sigma\tau}(f(i)) = \lambda x^\sigma.e_\tau(\varphi_i(e_\sigma^{-1}(x^\sigma))).$$

Observe now that (1) implies (2). (The converse does not need to hold.) Therefore, if (1) applies, $e_{\sigma\tau}(f(i)) = \lambda x^\sigma.e_\tau(\varphi_i(e_\sigma^{-1}(x^\sigma))) = \lambda x^\sigma.Mx^\sigma = M$ by axiom $(\eta)$. ∎

**3.4.** THEOREM. *Let* $a = \{a_i \mid i \in \mathrm{At}\}$ *be a collection of complete objects in* $\mathbf{pPER}_\omega$ *s.t.* $\mathrm{Term}_i <_p a_i$, *for all* $i \in \mathrm{At}$. *Then there exist a p.r.s.* $\{\alpha_\sigma, \beta_\sigma\}_{\sigma \in \mathrm{Tp}}$ *in* $\mathbf{pPER}_\omega$ *from* $(\mathrm{pPER}_\omega)_a$ *onto* **Term**.

*Proof.* Let $a_i >_p \mathrm{Term}_i$ via $(\alpha_i, \beta_i)$. Assume by induction that $(\alpha_\sigma, \beta_\sigma)$, $(\alpha_\tau, \beta_\tau)$, partial retractions in $\mathbf{pPER}_\omega$, have been defined. We will first construct $\beta_{\sigma\tau} \in \mathbf{PER}_\omega[\mathrm{Term}_{\sigma\tau}, a_{\sigma\tau}]$ satisfying cond. (1) in 2.4.

Let $M \in \mathrm{Term}_{\sigma\tau}$. By definition, the f.d.c.:

$$\begin{tikzcd} \omega_\sigma \arrow[r, "\alpha'_\sigma"] \arrow[d, "\pi_\sigma"'] & \omega \arrow[r, "M'"] \arrow[d, "e_\sigma"] & \omega \arrow[r, "\beta'_\tau"] \arrow[d, "e_\tau"] & \omega_\tau \arrow[d, "\pi_\tau"] \\ a_\sigma \arrow[r, "\alpha_\sigma"'] & \mathrm{Term}_\sigma \arrow[r, "M"'] & \mathrm{Term}_\tau \arrow[r, "\beta_\tau"'] & a_\tau \end{tikzcd}$$

Define now $h_{\sigma\tau}$: $\mathrm{Term}_{\sigma\tau} \to ({a_\tau}^{a_\sigma})_p$ by

$$h_{\sigma\tau}(M) = \beta_\tau \circ M \circ \alpha_\sigma \in \mathbf{pPER}_\omega[a_\sigma, a_\tau].$$

Clearly $h_{\sigma\tau}$ is a well defined total map.

*Claim.* $h_{\sigma\tau} \in \mathbf{PER}_\omega[\mathrm{Term}_{\sigma\tau}, ({a_\sigma}^{a_\tau})_p]$.

*Proof.* Let $l \in R$ be as in (3.2.1). Since the composition of partial recursive functions is an effective operation, one has

$$\begin{aligned} \exists k \in R\, h_{\sigma\tau}(e_{\sigma\tau}(i)) \circ \pi_\sigma &= \beta_\tau \circ e_{\sigma\tau}(i) \circ \alpha_\sigma \circ \pi_\sigma \\ &= \pi_\tau \circ \beta'_\tau \circ \varphi_{l(i)} \circ \alpha'_\sigma \qquad \text{by the diagram} \\ &= \pi_\tau \circ \varphi_{k(i)}. \end{aligned}$$

By the definition of $\pi_{\sigma\tau p}$, the partial enumeration of $({a_\sigma}^{a_\tau})_p$ in $\mathbf{PER}_\omega$, one has $\pi_\tau \circ \varphi_j = \pi_{\sigma\tau p}(j) \circ \pi_\sigma$ and, hence, $h_{\sigma\tau}(e_{\sigma\tau}(i)) = \pi_{\sigma\tau}(k(i))$ for all $i \in \omega$, by the computation above. Equivalently,

$$h_{\sigma\tau} \circ e_{\sigma\tau} = \pi_{\sigma\tau p} \circ k.$$

That is, $h_{\sigma\tau} \in \mathbf{PER}_\omega[\mathrm{Term}_{\sigma\tau}, ({a_\sigma}^{a_\tau})_p]$ with $h'_{\sigma\tau} = k$. This concludes the proof of the claim.

By 1.20, $a_{\sigma\tau} < ({a_\tau}^{a_\sigma})_p$ via (in′, out′) in $\mathbf{PER}_\omega$.
Set then

$$\beta_{\sigma\tau} = \mathrm{out}' \circ h_{\sigma\tau} \in \mathbf{PER}_\omega[\mathrm{Term}_{\sigma\tau}, a_{\sigma\tau}].$$

By R.5 in Section 1, $\mathrm{out}' \circ h_{\sigma\tau}(M) = \mathrm{out}'(h_{\sigma\tau}(M)) = \mathrm{out} \circ \tau(h_{\sigma\tau}(M))$ and this is exactly the definition of the extension of $h_{\sigma\tau}(M) \in \mathbf{pPER}_\omega[a_\sigma, a_\tau]$, in the sense of 1.13, to a morphism in $\mathbf{PER}_\omega[a_\sigma, a_\tau]$. Indeed, by assumption on $a$ and 1.21, $\forall \tau \in \mathrm{Tp}$, $a_\tau$ is a complete object. Thus, if $h_{\sigma\tau}(M)(y)$ converges (i.e., $y \in \mathrm{dom}\, \alpha_\sigma$), $\beta_{\sigma\tau}(M)(y) = h_{\sigma\tau}(M)(y) = \beta_\tau \circ M \circ \alpha_\sigma(y)$ and this is exactly cond. (2) in 2.4 for $\beta_{\sigma\tau}$.

We now need to define $\alpha_{\sigma\tau} \in \mathbf{pPER}_\omega[a_{\sigma\tau}, \mathrm{Term}_{\sigma\tau}]$. By definition the f.d.c.:

$$\begin{tikzcd} \omega \arrow[r, "\beta'_\sigma"] \arrow[d, "e_\sigma"'] & \omega_\sigma \arrow[r, "\varphi_l"] \arrow[d, "\pi_\sigma"] & \omega_\tau \arrow[r, "\alpha'_\tau"] \arrow[d, "\pi_\tau"] & \omega \arrow[d, "e_\tau"] \\ \mathrm{Term}_\sigma \arrow[r, "\beta_\sigma"'] & a_\sigma \arrow[r, "\pi_{\sigma\tau}(i)"'] & a_\tau \arrow[r, "\alpha_\tau"'] & \mathrm{Term}_\tau \end{tikzcd}$$

Define then $g_{\sigma\tau}$: $(a_{\sigma\tau}) \to ({\mathrm{Term}_\tau}^{\mathrm{Term}_\sigma})_p$ by

$$g_{\sigma\tau}(\pi_{\sigma\tau}(i)) = \alpha_\tau \circ \pi_{\sigma\tau}(i) \circ \beta_\sigma \in \mathbf{pPER}_\omega[\mathrm{Term}_\sigma, \mathrm{Term}_\tau].$$

Clearly $g_{\sigma\tau}$ is a well defined total map.

*Claim.* $g_{\sigma\tau} \in \mathbf{PER}_\omega[a_{\sigma\tau}, (\mathrm{Term}_\tau{}^{\mathrm{Term}_\sigma})_p]$.

*Proof.* As the composition of partial recursive functions is effective, one has

$$\begin{aligned} \exists r \in R\, g_{\sigma\tau}(\pi_{\sigma\tau}(i)) \circ e_\sigma &= \alpha_\tau \circ \pi_{\sigma\tau}(i) \circ \beta_\sigma \circ e_\sigma \\ &= e_\tau \circ \alpha'_\tau \circ \varphi_i \circ \beta'_\sigma \quad \text{by the diagram} \\ &= e_\tau \circ \varphi_{r(i)}. \end{aligned}$$

Moreover, if $\pi_{\sigma\tau p}$ is the enumeration of $(\mathrm{Term}_\tau{}^{\mathrm{Term}_\sigma})_p$ in $\mathbf{PER}_\omega$, one has $e_\tau \circ \varphi_i = \pi_{\sigma\tau p}(i) \circ e_\sigma$ and, hence,

$$(*) \qquad \begin{aligned} g_{\sigma\tau}(\pi_{\sigma\tau}(i)) \circ e_\sigma &= e_\tau \circ \varphi_{r(i)} \quad \text{for all } i \in \omega \\ &= \pi_{\sigma\tau p}(r(i)) \circ e_\sigma. \end{aligned}$$

That is, $g_{\sigma\tau} \circ \pi_{\sigma\tau} = \pi_{\sigma\tau p} \circ r$ and, thus,

$$g_{\sigma\tau} \in \mathbf{PER}_\omega[a_{\sigma\tau}, (\mathrm{Term}_\tau{}^{\mathrm{Term}_\sigma})_p] \qquad \text{with } g'_{\sigma\tau} = r.$$

This concludes the proof of the claim.

Let $f$ be the function given by Lemma 3.3. Define then $\alpha_{\sigma\tau}$ by

$$\alpha_{\sigma\tau}(\pi_{\sigma\tau}(i)) = e_{\sigma\tau}(f(r(i))) \quad \text{for all } i \in \omega.$$

Note now that, by definition, $f$ above preserves indexes, that is $\varphi_i = \varphi_j \Rightarrow e_{\sigma\tau}(f(i)) = e_{\sigma\tau}(f(j))$. Thus $\alpha_{\sigma\tau} \in \mathbf{pPER}_\omega[a_{\sigma\tau}, \mathrm{Term}_{\sigma\tau}]$ with $\alpha'_{\sigma\tau} = f \circ r$. We only have to show that $\alpha_{\sigma\tau}$ satisfies cond. (1). That is, suppose that

$$(**) \qquad \exists M \in \mathrm{Term}_{\sigma\tau} \; \alpha_\tau \circ \pi_{\sigma\tau}(i) \circ \beta_\sigma = M$$

(this is the assumption in cond. (1)). Recall that, by $(*)$ above,

$$\alpha_\tau \circ \pi_{\sigma\tau}(i) \circ \beta_\sigma \circ e_\sigma = g_{\sigma\tau}(\pi_{\sigma\tau}(i)) \circ e_\sigma = e_\tau \circ \varphi_{r(i)}$$

and, hence, condition $(**)$ can be rewritten as:

$$({}^{\,*}_{**}) \qquad \exists M \in \mathrm{Term}_{\sigma\tau} \; e_\tau \circ \varphi_{r(i)} = M \circ e_\sigma.$$

Finally, by the construction of $f$, if $({}^{\,*}_{**})$ holds, one has:

$$M = e_{\sigma\tau}(f(r(i))) = \alpha_{\sigma\tau}(\pi_{\sigma\tau}(i))$$

and this proves cond. (1). ∎

We conclude this section by a very loose summary of applications of the previous results and their consequences (see Section 2) to some interesting type-structures in denotational semantics.

The category **CD** of constructive domains may be naturally defined from Scott's effectively given domains (see Giannini–Longo [8] or Ershov

[6] for a detailed definition). In short, each constructive domain is the collection of computable elements of some effectively given domain. By generalized Myhill–Shepherdson theorem **CD** is a full subCCC of $\mathbf{ER}_\omega$. Indeed, it may be embedded into $\mathbf{PER}_\omega$ by a full and faithfull functor which preserves products and function spaces. Clearly all objects in **CD** are complete as they possess a least, bottom, element.

As already pointed out, the usual ground types of data are trivially in **CD**, e.g. all flat cpo's (such as $\omega$). Moreover, given $i \in \mathrm{At}$, all what is required in order to have a partial retraction, from a constructive domain $a$ onto $\mathrm{Term}_i$, is that $a$ contains as many incompatible elements as the cardinality of $\mathrm{Term}_i$. Thus, one may also take as ground types the following objects in **CD** and 3.4 applies: the partial recursive functions, the effectively generated trees on some countable alphabet... Interesting instances of the later example of data type are the (possibly infinite) parenthesized expressions in a language (e.g. LISP $S$-expressions).

## Acknowledgements

We are greatly endebted to Eugenio Moggi for a few stimulating discussions which provided the main hints for the solution of some problems we had. Moreover, Eugenio's recent work on partial morphisms set the basis for the results stated here on that matter.

## References

[1] H. Barendregt, *The lambda calculus; its syntax and semantics*, Revised edition, North-Holland, 1984.

[2] G. Berry, *Modèles complètement adequats et stables des lambda-calculus type*, Thèse de Doctorat, Université Paris, VII, 1979.

[3] –, P. L. Curien, *Sequential Algorithms on Concrete Data Structures*, Theor. Comp. Sci. 20, 265–321.

[4] P. Dibjer, *Category-theoretic logics and algebras of programs*, Ph. D. Thesis, Chalmers University, Goeteborg 1983.

[5] A. Di Paola, A. Heller, *Dominical Categories*, City Univ. New York 1984.

[6] Yu. L. Ershov, *Model C of the partial continuous functionals*, Logic Colloquium 76 (Gandy, Hyland eds.) North-Holland, 1977.

[7] H. Friedman, *Equality between functionals*, Logic Colloquium (Parikh ed.), LNM 453, Springer-Verlag, 1975.

[8] P. Giannini, G. Longo, *Effectively given domains and lambda calculus semantics*, Inform. and Control, 62, 1 (1984), 36–63.

[9] K. Goedel, *Ueber eine bicher noch nicht benuetze Erweiterung des finiten Standpuntes*, Dialectica 12 (1958), 280–287.

[10] R. Hindley, J. Seldin, *To H. B. Curry: essays in Combinatory Logic, lambda-calculus and formalism*, Academic Press, 1980.

[11] M. Hyland, *The effective Topos*, in: *The Brouwer Symposium* (Troelstra, Van Dalen eds.) North-Holland, 1982.
[12] A. Kanda, *Fully effective solutions of recursive domain equations*, MFCS'79, LNCS 74, Springer-Verlag, 1979.
[13] J. Lambek, *From lambda-calculus to cartesian closed categories*, in: Hindley–Seldin [10], 1980.
[14] G. Longo, E. Moggi, *The Hereditary Partial Recursive Functionals and Recursion Theory in higher types*, J. Symb. Logic 49 (1984), 1319–1332.
[15] –, –, *Cartesian Closed Categories of Enumerations and effective Type Structures*, Symposium on Semantics of Data Types (Khan, MacQueen, Plotkin eds.), LNCS 173, Springer-Verlag, 1984, 235–247.
[16] –, –, *Goedel-numberings, principal morphisms, Combinatory Algebras: a category-theoretic characterization of functional completeness* (prelim. version, Math. Found. Comp. Sci., Prague 1984 (Chytil, Koubek eds.) LNCS 176, Springer-Verlag, 1984, 397–406, TCS to appear.
[17] S. MacLane, *Categories for the working mathematician*, Springer-Verlag, 1971.
[18] E. Moggi, *Partial Morphisms in Categories of effective objects*, Inform. and Control, (1986), to appear.
[19] A. Obtułowicz, *The logic of categories of partial functions and its applications*, Ph. D. thesis, Warsaw 1982, Dissertationes Math. 241 (1986).
[20] –, *Algebra of Constructions I*, Inform. and Control, 1986, to appear.
[21] G. Plotkin, *Notes on completeness of the full continuous type hierarchy*, Manuscript, Lab. for Comp. Sci., M.I.T., 1982.
[22] –, *Types and partial functions*, C.S. Dept. Edinburgh, 1984.
[23] A. Poigne, *On specification Theories and Models with higher types*, Inform. and Control, 1985, to appear.
[24] G. Rosolini, *Domains and dominical categories*, C.S. Dept., C.M.U. 1985.
[25] –, *Continuity and effectiveness in Topoi*, D. Phil. Thesis, Oxford Univ., 1986, in preparation.
[26] D. Scott, *Data types as lattices*, SIAM Journal of Computing 5 (1976), 522–587.
[27] –, *Relating theories of the lambda-calculus*, in: Hindley–Seldin [10], 1980.
[28] –, *Domains for denotational semantics* (preliminary version), Proceedings ICALP 82, LNCS 140, Springer-Verlag, 1982.
[29] M. Smyth, *Effectively Given Domain*, Theoret. Comp. Sci. 5 (1977), 255–272.
[30] –, G. Plotkin, *The category-theoretic solution of recursive domain equations*, SIAM Journal of Computing 11 (1982), 761–783.
[31] R. Statman, *Completeness, invariance and lambda-definability*, J. Symb. Logic 47, 1 (1980), 17–26.
[32] –, *Logical relations and typed lambda-calculus*, Inform. and Control, (1984), to appear.
[33] –, *Equality between functionals revisited*, in: H. Friedman's research on the found. of Math. (Harrington et al. eds), North-Holland, 1985.
[34] Types, *Discussion on the 1-section theorem between V. Breazu-Tannen and R. Statman*, Types mailing list, EDU computer net, 1986.
[35] A. Troelstra, *Metamathematical investigation of Intuitionistic Arithmetic and Analysis*, LNM 344, Springer-Verlag, Berlin 1973.

*Presented to the semester*
*Mathematical Problems in Computation Theory*
*September 16–December 14, 1985*