Electronic Notes in Theoretical Computer Science 70 No. 1 (2003)
URL: http://www.elsevier.nl/locate/entcs/volume70.html 16 pages

Subtyping in Logical Form

Ugo de'Liguoro [1,2]

Dipartimento di Informatica
Università di Torino
c.so Svizzera 185, 10149 Torino, Italy

Abstract

By using intersection types and filter models we formulate a theory of types for a λ-calculus with record subtyping via a finitary programming logic. Types are interpreted as spaces of filters over a subset of the language of properties (the intersection types) which describes the underlying type free realizability structure. We show that such an interpretation is a PER semantics, proving that the quotient space arising from "logical" PERs taken with the intrinsic ordering is isomorphic to the filter semantics of types.

1 Introduction

Subtyping is a form of polymorphism which is based on the intuition that any term of type A might safely occur in a context of type B whenever A is a subtype of B. The basic approach to the theory of subtyping is syntactic in nature: looking for semantic investigations of this relation one is led to the successful approach which has been proposed in [6]. This is based on the same interpretation of types than second order types, namely PERs over the Kleene partial combinatory algebra $(\omega, \{\_\}\_)$; in this framework the subtyping relation is modeled simply by subset inclusion of PERs.

The study of models of polymorphism has largely profited of Cardone and Amadio [8,4] proposal to move from ω to D_∞ models of the type-free lambda-calculus, seen as realizability structures. The advantage is that D_∞ carries a topological structure that can be exploited to interpret a rich variety of type constructors, like recursive types and bounded quantification. Building over this theory [2] provides a general way to ordering the domain of complete and uniform PERs introduced in [8,4] in such a way that it is an ω-algebraic

[1] Partially supported by MURST Cofin'01 COMETA Project, IST-2001-33477 DART Project and IST-2001-322222 MIKADO Project.
[2] Email: deligu@di.unito.it

cpo. This construction has been framed in [7] in a general theory of "acceptable" PERs which give rise to models of F_ω with F-bounded quantification, a problem which was left open in [2].

Filter models based on intersection type assignment systems [5] and domain logic [3] using (pre)-locales as the base logic provide a logical approach to domain theory and denotational semantics, where domains are essentially sets of theories and the denotation of a program is the set of sentences true of it (its theory). While filter models have been invented to model type free calculi, domain logic provides a framework to model (first order) typed languages within the category of 2/3 SFP.

When dealing with models of polymorphism over realizability structures we are in an intermediate situation, where the term interpretation is type free (based on erasure maps), and the type structure is recovered via partial equivalence relations. As remarked in [4] these models "suffer from a typical drawback of denotational semantics, namely their equational theories are hard to characterize and typically not even r.e. Therefore there are obvious difficulties to extract from the models and justify a *finitary programming logic*".

We face here this problem: by restricting to intersection types (but we think that our construction can be carried on to the framework of domain logic with a modest overhead) we are able to show that a filter model, close to that one used in [9] to study termination of type free ς-terms, models a λ-calculus with record subtyping in such a way that types are interpreted into certain subdomains of the underlying realizability structure which admit a logical description.

The basic idea to capture subtyping logically is that terms are identified with the sets of their properties, and properties are classified according to types. A subtype A of some type B is then associated to a finer language than the language associated to B, so that any pair of terms which cannot be distinguished according to A, will be such with respect to B. So if $M : A$ is the set of properties of M of type A and $A <: B$ then we expect from the theory that $M : B = (M : A) \cap B$. Therefore terms are not equal in general, rather they are (or are not) equal with respect a type A: $M = N : A$ means (roughly) that $M \cap A = N \cap A$. Combining these two, if $M = N : A$ and $A <: B$ then $M = N : B$ as expected.

Then we prove that this semantics is a PER semantics, although different from the standard one: indeed subtyping is modeled by discriminability w.r.t. certain sets of properties rather than by relation inclusion.

2 Subtyping over realizability structures

For the sake of concreteness we introduce first order types with record types, a notion of subtyping syntactically defined by an inference system, and a simply typed λ-calculus with records. The choice of the calculus is motivated by the fact that it is the first order fragment of what is needed to encode object-

Type grammar

$$A, B ::= G \mid \{\ell_i : B_i \ ^{(i \in I)}\} \mid A \to B$$

Subtyping System

$$\frac{}{\Sigma \vdash A <: A} \qquad \frac{\Sigma \vdash A <: B \quad \Sigma \vdash B <: C}{\Sigma \vdash A <: C} \qquad \frac{\Sigma \vdash A' <: A \quad \Sigma \vdash B <: B'}{\Sigma \vdash A \to B <: A' \to B'}$$

$$\frac{G <: G' \in \Sigma}{\Sigma \vdash G <: G'} \qquad \frac{\Sigma \vdash A_j <: B_j \quad \forall j \in J \subseteq I}{\Sigma \vdash \{\ell_i : A_i \ ^{(i \in I)}\} <: \{\ell_j : B_j \ ^{(j \in J)}\}}$$

Term grammar

$$M, N ::= x \mid c \mid (\lambda x : A.M) \mid (MN) \mid \{\ell_i = M_i \ ^{(i \in I)}\} \mid M.\ell,$$

Typing System

$$\frac{x : A \in \Gamma}{\Gamma \vdash x : A} \qquad \frac{\Gamma, x : A \vdash M : B}{\Gamma \vdash \lambda x : A.M : A \to B} \qquad \frac{\Gamma \vdash M : A \to B \quad \Gamma \vdash N : A}{\Gamma \vdash MN : B}$$

$$\frac{c : G}{\Gamma \vdash c : G} \qquad \frac{\Gamma \vdash M_i : B_i \quad \forall i \in I}{\Gamma \vdash \{\ell_i = M_i \ ^{(i \in I)}\} : \{\ell_i : B_i \ ^{(i \in I)}\}} \qquad \frac{\Gamma \vdash M : \{\ell_i : B_i \ ^{(i \in I)}\} \quad j \in I}{\Gamma \vdash M.\ell_j : B_j}$$

$$\frac{\Gamma \vdash M : A \quad \Sigma \vdash A <: B}{\Gamma \vdash M : B} \ (Sub)$$

Fig. 1. Systems for deriving subtyping and typing judgments

calculi (but for the recursive types, which have not been considered here, for simplicity) [1,10,9].

The PER semantics of this calculus is then shortly introduced by fixing a realizability structure, introducing the PER interpretation of arrow and record types, and giving the erasure semantics for terms.

2.1 A simply typed λ-calculus with record subtyping

Definition 2.1 Types are generated by the grammar:

$$A, B ::= G \mid \{\ell_i : B_i \ ^{(i \in I)}\} \mid A \to B$$

where G ranges over some finite set of ground type constants, $I \subseteq \omega$ is a finite set of indexes and $\{\ell_j \mid j \in \omega\}$ a denumerable set of labels.

If Σ is a set of subtyping axioms among ground types of the shape $G <: G'$, the subtyping relation $\Sigma \vdash A <: B$ among types is defined according to the rules in Figure 1.

We consider a simply typed λ-calculus with records, whose pre-terms are generated by the grammar:

$$M, N ::= x \mid c \mid (\lambda x : A.M) \mid (MN) \mid \{\ell_i = M_i\ ^{(i \in I)}\} \mid M.\ell,$$

where c ranges over some countable set of constants of ground type, I ranges over finite sets of indexes. We adopt standard conventions for term notation. Terms are typed in the standard way by deriving judgments of the shape $\Gamma \vdash M : A$, where Γ is some finite set of assumptions $x : B$ with pair wise distinct subjects. The rules are given in Figure 1, where rule (*Sub*) has a premise which is the conclusion of a derivation in the subtyping system: by $\Gamma \vdash_\Sigma M : A$ we mean that $\Gamma \vdash M : A$ is derivable in the typing system fixing a set of subtyping axioms Σ.

2.2 Denotational Semantics

In the following we fix D as the initial solution, in a suitable category of domains, of the equation

$$(1) \qquad D \simeq O + E + [L \to D] + [D \to D]$$

where $O = \{\bot \sqsubseteq \top\}$, $E = E_1 + \cdots + E_k$ is a coalesced sum of domains interpreting constants of ground type, which are either flat or topped flat domains; L is a denumerable set of labels $\ell_0, \ell_1, \ldots$; $+$ is the coalesced sum. Being $[D \to D]$ a retract of D, D is a λ-model equipped with a continuous application function $app : D \times D \to D$ ($d \cdot e$ abbreviates $app(d, e)$). Records are interpreted as finite functions in $[L \to D]$, so that there exist the continuous mappings $sel : D \times L \to D$ and $lcond : D \times L \times D \to D$ satisfying certain axioms (see below definition 2.2 and [10] ch. 10, where this notion is introduced in the general case of partial combinatory algebras). Such a structure gives a model for the untyped λ-calculus with records, whose syntax is obtained from that of raw terms by erasing types (see Figure 2).

Definition 2.2 A *model* of the untyped λ-calculus with records is a structure $\langle D, \cdot, emp, sel, lcond\rangle$ such that $\langle D, \cdot\rangle$ is a λ-model, $emp \in D$ (the empty record) and:

(i) $sel(lcond\, x\, \ell_i\, y)\ell_i = y$,

(ii) $i \neq j \Rightarrow sel(lcond\, x\, \ell_i\, y)\ell_j = sel\, x\, \ell_j$.

We use the following abbreviations: $d.\ell \equiv sel\, d\, \ell$, $d.\ell := e \equiv lcond\, d\, \ell\, e$, $\{\ell_i = d_i\ ^{(i \in \{1..k\})}\} \equiv lcond(\ldots(lcond\, emp\, \ell_1\, d_1)\ldots)\, \ell_k\, d_k$: we assume that labels $\ell_1, \ldots, \ell_k$ are pair wise distinct, so that their actual order does not matter.

Proposition 2.3 *If D is a solution of domain equation (1), then it is a model of the untyped λ-calculus with records.*

Proof. Let $\varphi : D \to O + E + [L \to D] + [D \to D]$ be the isomorphism given by the solution of the domain equation, with inverse ψ; for H among

$O, E, [L \to D], [D \to D]$ let $in_H : H \to O + E + [L \to D] + [D \to D]$ be its continuous injection map. Then we define:

- $d \cdot e = \begin{cases} f(e) & \text{if } \varphi(d) = in_{[D\to D]}(f) \\ \perp & \text{otherwise} \end{cases}$
- $emp^D = \perp = \psi(in_{[L\to D]}(\lambda\ell.\perp))$
- $sel^D d\,\ell = \begin{cases} r(\ell) & \text{if } \varphi(d) = in_{[L\to D]}(r) \\ \perp & \text{otherwise} \end{cases}$
- $lcond^D d\,\ell\,e = \psi(in_{[L\to D]}(r))$, where $r(\ell') = \begin{cases} e & \text{if } \ell' = \ell \\ sel^D d\,\ell' & \text{otherwise} \end{cases}$

That this is a λ-model is known from the literature; that equations in definition 2.2 are satisfied is an easy check. $\square$

Interpreting types as PERs over a suitable partial combinatory algebra, the subtyping relation can be interpreted as set theoretic inclusion of PERs [6]. Instead of the Kleene algebra $(\omega, \{\_\}\_)$ one may consider the combinatory algebra $(D, \cdot)$ [8,4].

Definition 2.4 A PER over D is a symmetric and transitive binary relation over D; the *domain* of a PER R is the set $|R| = \{d \in D \mid \langle d, d\rangle \in R\}$; if $d \in D$ then $[d]_R = \{e \in D \mid \langle d, e\rangle \in R\}$; finally the *quotient* of D by R is the set $D/R = \{[d]_R \mid d \in |R|\}$.

We write $d\,R\,e$ for $\langle d, e\rangle \in R$; moreover we say simply PER in place of PER over D.

Proposition 2.5 *If R, S and R_i for all $i \in I$ are PERs then $(R \to S)$ and $\{\ell_i : R_i\ ^{(i\in I)}\}$ are such, where:*

(i) $d\,(R \to S)\,e \Leftrightarrow \forall d', e' \in D.\ d' R\, e' \Rightarrow (d \cdot d')\,S\,(e \cdot e')$,

(ii) $d\,\{\ell_i : R_i\ ^{(i\in I)}\}\,e \Leftrightarrow \forall i \in I.\ (sel\,d\,\ell_i)\,R_i\,(sel\,e\,\ell_i)$.

Proof. Easy from definitions. $\square$

Definition 2.6 Given a mapping η from ground types to PERs over E (hence over its image in D), the interpretation $[\![A]\!]_\eta$ of the type A over D is defined inductively:

(i) $[\![G]\!]_\eta = \eta(G)$,

(ii) $[\![A \to B]\!]_\eta = ([\![A]\!]_\eta \to [\![B]\!]_\eta)$,

(iii) $[\![\{\ell_i : B_i\ ^{(i\in I)}\}]\!]_\eta = \{\ell_i : [\![B_i]\!]_\eta\ ^{(i\in I)}\}$.

Proposition 2.7 *If $\eta(G) \subseteq \eta(G')$ for all $G <: G' \in \Sigma$ and $\Sigma \vdash A <: B$, then $[\![A]\!]_\eta \subseteq [\![B]\!]_\eta$.*

Untyped terms

$$U, V ::= x \mid c \mid (\lambda x.U) \mid (UV) \mid \{\ell_i = U_i\ ^{(i \in I)}\} \mid U.\ell$$

Erasure map

$$\begin{aligned}
erase(x) &= x \\
erase(c) &= c \\
erase(MN) &= erase(M)erase(N) \\
erase(\lambda x : A.M) &= \lambda x.erase(M) \\
erase(\{\ell_i = M_i\ ^{(i \in I)}\}) &= \{\ell_i = erase(M_i)\ ^{(i \in I)}\} \\
erase(M.\ell) &= erase(M).\ell
\end{aligned}$$

Interpretation of untyped terms

$$\begin{aligned}
[\![x]\!]^D_\rho &= \rho(x) \\
[\![c]\!]^D_\rho &= \psi(in_E(\mathbf{c})), \quad \text{for a constant } \mathbf{c} \in E \\
[\![UV]\!]^D_\rho &= [\![U]\!]^D_\rho \cdot [\![V]\!]^D_\rho \\
[\![\lambda x.U]\!]^D_\rho &= \psi(in_{[D \to D]}(\lambda d \in D.[\![U]\!]^D_{\rho[x:=d]})) \\
[\![\{\ell_i = U_i\ ^{(i \in I)}\}]\!]^D_\rho &= \{\ell_i = [\![U_i]\!]^D_\rho\ ^{(i \in I)}\} \\
[\![U.\ell]\!]^D_\rho &= [\![U]\!]^D_\rho.\ell
\end{aligned}$$

Fig. 2. Untyped interpretation of terms

Proof. By induction over the derivation of $\Sigma \vdash A <: B$ □

Any D satisfying the domain equation (1) can be turned into a model of this calculus by defining the obvious erasing mapping $erase(M)$ sending typed into type free terms and then giving them the standard interpretation $[\![\_]\!]^D_\rho$ as defined in Figure 2.

Proposition 2.8 *Let η be an interpretation of ground types into PERs satisfying Σ, and ρ a mapping from term variables to D; suppose that $\rho, \eta \models \Gamma$ that is $\rho(x) \in |[\![B]\!]_\eta|$ whenever $x : B \in \Gamma$. Then if $\Gamma \vdash_\Sigma M : A$ then $[\![erase(M)]\!]^D_\rho \in |[\![A]\!]_\eta|$.*

Proof. This follows by proving, by induction over derivations and by proposition 2.7, the statement that $[\![erase(M)]\!]^D_\rho [\![A]\!]_\eta [\![erase(M)]\!]^D_{\rho'}$ whenever $\rho(x)[\![B]\!]_\eta$ $\rho'(x)$ for all $x : B \in \Gamma$. (For details see e.g. [10], ch. 10). □

3 Complete Uniform PERs

A solution D of the domain equation (1) can be constructed as the inverse limit $D_\infty = \lim_\leftarrow D_n$, where $D_n = F^n(\perp)$, with F a continuous functor whose object action is described by equation (1), and $\bot$ is the initial object of the given category. Each D_n is isomorphic to a subdomain $\widehat{D}_n$ of D, which is the image of a continuous projection $\pi_n : D \to D$ such that $\bigsqcup_n \pi_n = \mathrm{Id}_D$ (with $\pi_0 = \lambda x.\bot$) and $\widehat{D}_n$ is a subset of $K(D)$, the set of compact (finite) elements of D. These well known facts determine a *notion of approximation* over D (in the sense of [8]), where the approximation at n of $d \in D$ is just $d_{[n]} = \pi_n(d)$; moreover for all $n > 0$ and $a \in E$ it is the case that $a_{[n]} = a$. The following notion of *complete uniform* PERs has been independently introduced by Cardone [8] (who calls them CUA relations), and Amadio [4]:

Definition 3.1 A *complete uniform* PER, shortly a CUPER over D is a PER $R \subseteq D \times D$ which is:

(i) *pointed:* $\bot R \bot$,

(ii) *complete:* if $\langle d, e\rangle = \bigsqcup_{r<\omega} \langle d^r, e^r\rangle$ and $d^r R\, e^r$ for all r, then $d\ R\ e$,

(iii) *uniform:* if $d\ R\ e$ then $d_{[n]}\ R\ e_{[n]}$ for all n.

As suggested by the definition, CUPERs are some kind of relational domains, and their construction is the inverse limit of suitable functors extending F, as shown by the "fundamental diagram" in [7]. Nonetheless the problem of ordering the quotient space D/R when R is a CUPER in such a way that it is an algebraic cpo is a non trivial one, as argued in [4]; we shall consider the solution proposed in [2].

Definition 3.2 The *intrinsic preorder* $\leq_R$ over $|R|$ is the binary relation:

$$d \leq_R e \Leftrightarrow \forall f \in |(R \to O)|.\ f \cdot d = \top \Rightarrow f \cdot e = \top,$$

where O is identified with the diagonal over O.

This defines a complete preorder over $|R|$ which includes both R and $\sqsubseteq$; because of completeness and uniformity of R, $\leq_R$ is the least complete preorder with such a property.

Theorem 3.3 *Suppose that R is an* antisymmetric *CUPER, that is for all $d, e \in |R|$:*

$$d \leq_R e \leq_R d \Rightarrow d\, R\, e.$$

Then the ordering $[d]_R \leq [e]_R \Leftrightarrow d \leq_R e$ is well defined, and turns D/R into an ω-algebraic cpo, where $[\bot]_R$ is the least element, $[\bigsqcup_r d^r]_R$ the sup of the $[d^r]_R$ (if the d^r form a $\sqsubseteq$-directed set in $|R|$), and compact elements are of the form $[a]_R$, with $a \in K(D) \cap |R|$.

Proof. See [2] theorem 1. □

Properties of elements in a domain D are basic opens of the Scott topology over D. We call sub-basis a subset of the basis of the latter topology, which is still a basis.

Definition 3.4 A *sub-basis* of a domain D is a subset $X \subseteq K(D)$ which is closed under finite sups of compatible elements (i.e. bounded in D) and such that $\bot \in X$. If D has a notion of approximation, we say that X is *closed under approximations* if for any $d \in D$ and $n \in \mathbb{N}$:

$$d\lceil X \neq \{\bot\} \wedge d_{[n]} \neq \bot \Rightarrow d_{[n]}\lceil X \neq \{\bot\},$$

where $d\lceil X = K(d) \cap X$ and $K(d) = \{a \in K(D) \mid a \sqsubseteq d\}$.

If X is a sub-basis then by the algebraicity of D, the set $\{a\uparrow \mid a \in X\}$ is the basis of a topology $\mathcal{T}_X$ over D which is coarser than the Scott topology of D. Similarly, if $X \subseteq Y \subseteq K(D)$ and both X and Y satisfy the above requirements, then $\mathcal{T}_X$ is coarser than $\mathcal{T}_Y$. If we define a binary relation $d \sim_X e \Leftrightarrow d\lceil X = e\lceil X$ (in other words $\sim_X$ is the equivalence induced by the specialization preorder of $\mathcal{T}_X$), then $\sim_Y \subseteq \sim_X$, so that, reasoning by analogy with PER inclusion, the finer topology $\mathcal{T}_Y$ should be a subtype of $\mathcal{T}_X$. The problem here is that $\sim_X$ is an equivalence relation, and not just a PER; therefore we refine the construction as follows.

Definition 3.5 Let X be a sub-basis of D: then $R_X \subseteq D \times D$ is the relation such that

$$dR_Xe \Leftrightarrow d = \bot = e \vee d\lceil X = e\lceil X \neq \{\bot\}.$$

We call R_X a *topological PER.*

Observe that $d \in |R_X|$ if and only if either $d = \bot$ or there exists $x \in X\setminus\{\bot\}$ such that $x \sqsubseteq d$.

Lemma 3.6 *If X is a sub-basis of D, then R_X is a pointed and complete PER.*

Proof. R_X is pointed by definition. Suppose that $d^r R_X e^r$ for all r and $\langle d, e\rangle = \bigsqcup_{r<\omega}\langle d^r, e^r\rangle$: then either $d^r = \bot = e^r$ for all r, in which case $d = \bot = e$ and we are done, or there exists t s.t. $d^t \neq \bot$, and therefore $e^t \neq \bot$. Indeed $d^t R_X e^t$, which implies that there exists $x \in X$ s.t. $x \sqsubseteq d^t, e^t$; it follows that $x \sqsubseteq d, e$, so that both $d\lceil X \setminus \{\bot\} \neq \emptyset$ and $e\lceil X \setminus \{\bot\} \neq \emptyset$. If $x \in d\lceil X \setminus \{\bot\}$ then $x \sqsubseteq d^s$ for some s since $d = \bigsqcup_r d^r$ and x is finite; by hypothesis $x \sqsubseteq e^s$ which implies $x \sqsubseteq e$: it follows that $d\lceil X \setminus \{\bot\} \subseteq e\lceil X \setminus \{\bot\}$; the opposite inclusion is proved similarly, and we conclude that R_X is complete.

□

Proposition 3.7 *Let $X \subseteq K(D)$ be a sub-basis: then R_X is a CUPER if and only if X is closed under approximations.*

Proof. By lemma 3.6 R_X is a pointed complete PER. To see that it is uniform let dR_Xe, where d and e are both different than $\bot$. If $n = 0$ then $d_{[n]} = \bot = e_{[n]}$ which immediately implies that $d_{[n]}R_Xe_{[n]}$. Otherwise let $n > 0$: by hypothesis $d\lceil X \neq \{\bot\}$ so that, if $d_{[n]} \neq \bot$ then $d_{[n]}\lceil X \setminus \{\bot\} \neq \emptyset$, by the hypothesis that X is closed under approximations. Let x be an element of the latter set: then $\bot \sqsubseteq x \sqsubseteq d_{[n]}$ so that $x \sqsubseteq d$: it follows that $x \sqsubseteq e$ hence $x = x_{[n]} \sqsubseteq e_{[n]}$ since

$x \in D_i$ and $(\_)_{[n]}$ is monotonic. We conclude that $\{\bot\} \neq d_{[i]}\lceil X \subseteq e_{[n]}\lceil X$: the opposite inclusion is symmetric, whence $d_{[n]} R_X e_{[n]}$.

Vice versa let $d\lceil X \neq \{\bot\}$ and $d_{[n]} \neq \bot$. It follows that $d\, R_X\, d$ so that $d_{[n]} R_X d_{[n]}$ for any $n \in \mathbb{N}$ since R_X is uniform which implies $d_{[n]}\lceil X \neq \{\bot\}$, namely X is closed under approximations.

□

To make reading easier in the following we write simply $(d \Rightarrow e)$ instead of $\psi(in_{[D\to D]}(d \Rightarrow e))$ and similarly $(\ell \Rightarrow d)$ in place of $\psi(in_{[L\to D]}(\ell \Rightarrow d))$. We also write functional application as $f(d)$ instead of $f \cdot e$.

Lemma 3.8 *If $d, e \in |R_X|$, where $X \subseteq K(D)$ is a sub-basis of D which is closed under approximations, then*

$$d \leq_{R_X} e \Leftrightarrow d\lceil X \subseteq e\lceil X.$$

It follows that R_X is antisymmetric.

Proof. Suppose $d \leq_{R_X} e$: if $a \in d\lceil X$ then $(a \Rightarrow \top) \in |(R_X \to O)|$ and $(a \Rightarrow \top)(d) = \top$: were $a \not\sqsubseteq e$ we had $(a \Rightarrow \top)(e) = \bot$, contradicting the hypothesis.

On the other hand suppose that $d\lceil X \subseteq e\lceil X$ and $f \in |(R_X \to O)|$ s.t. $f(d) = \top$. If $f(e) = \bot$ then, by observing that $e\, R_X(\bigsqcup e\lceil X)$, $f(\bigsqcup e\lceil X) = \bot$, namely $f(z) = \bot$ for all $z \in e\lceil X$ by continuity; but then $f(z) = \bot$ for all $z \in d\lceil X$ by hypothesis, which implies $\bot = f(\bigsqcup d\lceil X) = f(d)$.

□

Observe that in the above lemma the hypothesis $d, e \in |R_X|$ is essential, since otherwise $d\lceil X = e\lceil X$ is only necessary but not sufficient condition for $d\ R_X\ e$ to hold.

In definition 2.1 the arrow and record type constructors are considered; these are interpreted by the arrow an record functors over PER, according to proposition 2.5. We introduce arrow and record operators acting on sub-bases, and compare the resulting logical PERs to those obtained by applying the arrow and record functors.

Definition 3.9 If $X, Y, X_i \subseteq K(D)$ are sub-bases of D then define:

(i) $X \to Y = \{\bigsqcup_{i\in I}(d_i \Rightarrow e_i) \mid d_i \in X, e_i \in Y, \text{the sup exists}\}$,

(ii) $\{\ell_i : X_i\ ^{(i\in I)}\} = \{\bigsqcup_{i\in I}(\ell_i \Rightarrow d_i) \mid d_i \in X_i\}$

where I is always finite, and $d_i \Rightarrow e_i$, $\ell_i \Rightarrow d_i$ are step functions.

Proposition 3.10 *Let $X, Y, X_i \subseteq K(D)$ be sub-bases then $X \to Y$ and $\{\ell_i : X_i\ ^{(i\in I)}\}$ are such, and moreover:*

(i) $R_{X\to Y} \supseteq (R_X \to R_Y)$,

(ii) $R_{\{\ell_i : X_i\ ^{(i\in I)}\}} = \{\ell_i : R_{X_i}\ ^{(i\in I)}\}$.

Proof. That $R_{\{\ell_i : X_i\ ^{(i\in I)}\}} = \{\ell_i : R_{X_i}\ ^{(i\in I)}\}$ is immediate by definitions.

To see that $R_{X\to Y} \supseteq (R_X \to R_Y)$ let $f(R_X \to R_Y)g$ and suppose that $(a \Rightarrow b) \in f\lceil(X \to Y)$. Then $b = (a \Rightarrow b)(a) \sqsubseteq f(a)R_Y g(a)$; hence $b \sqsubseteq g(a)$ so that $(a \Rightarrow b) \sqsubseteq g$. This shows $f\lceil(X \to Y) \subseteq g\lceil(X \to Y)$ whence $f \leq_{R_{X\to Y}} g$: being $(R_X \to R_Y)$ symmetric this also shows that $g \leq_{R_{X\to Y}} f$ and we conclude being $R_{X\to Y}$ antysimmetric by lemma 3.8.

$\square$

Unfortunately $R_{X\to Y} \not\subseteq (R_X \to R_Y)$. As a matter of fact we can show that $(a \Rightarrow b) \in |R_X \to R_Y|$ if and only if both $a \in |R_X|$ and $b \in |R_Y|$: but $(X \to Y) \ni (x \Rightarrow y) \sqsubseteq (a \Rightarrow b)$ (where $y \neq \bot$) does not imply $x \sqsubseteq a$.

4 A logical interpretation

4.1 Intersection types and the filter model

In this section intersection types are called *properties* to emphasize that they are the formulas of some program logic, and to keep them distinct from types in the sense of definition 2.1.

Definition 4.1 The language $\mathcal{L}$ of *properties* is generated by the grammar:

$$\sigma, \tau ::= \alpha \mid \omega \mid \sigma{\to}\tau \mid \langle \ell : \sigma \rangle \mid \sigma \wedge \tau$$

where α ranges over a countable set of atomic properties.

The intended meaning of $\langle \ell : \sigma \rangle$ is: the property satisfied be a record having a field labeled by ℓ, whose entry satisfies σ. We abbreviate $\bigwedge_i \langle \ell_i : \sigma_i \rangle$ by $\langle \ell_i : \sigma_i \;^{(i\in I)} \rangle$; if $I = \emptyset$ then this intersection is ω.

Definition 4.2 Over the set $\mathcal{L}$ of properties it is defined a binary relation $\leq$ (the implication) such that the following axioms are satisfied:

(i) axioms making $\leq$ reflexive and transitive, $\sigma \wedge \tau$ the meet and ω the top;

(ii) $\omega \leq \omega \to \omega$,

(iii) $(\sigma \to \tau) \wedge (\sigma{\to}\tau') \leq \sigma \to (\tau \wedge \tau')$,

(iv) $\sigma \geq \sigma', \tau \leq \tau' \Rightarrow \sigma \to \tau \leq \sigma' \to \tau'$,

(v) $\omega \leq \langle \ell : \omega \rangle$,

(vi) $\sigma \leq \tau \Rightarrow \langle \ell : \sigma \rangle \leq \langle \ell : \tau \rangle$,

(vii) $\langle \ell : \sigma \rangle \wedge \langle \ell : \tau \rangle \leq \langle \ell : \sigma \wedge \tau \rangle$.

When restricted to arrow and intersection constructors, these are the axioms for intersection types of [5]; as far as the record properties are concerned, these are the same as those found in [9] but for $\omega \leq \langle \ell : \omega \rangle$: this is analogous to $\omega \leq \omega \to \omega$ and says that $\lambda l.\bot = \bot \in [L \to D]$ (see below). Finally it is easy to see that, if $\sigma_i \leq \tau_i$ for all $i \in I \supseteq J$ then $\langle \ell_i : \sigma_i \;^{(i\in I)} \rangle \leq \langle \ell_j : \tau_j \;^{(j\in J)} \rangle$.

A *filter* is a subset $F \subseteq \mathcal{L}$ which is upward closed w.r.t. $\leq$ and closed under finite intersections. The set of filters $\mathcal{F}$ ordered by set inclusion is an algebraic complete lattice which provides a solution of the equation (1) in this category. Without spelling this out in detail, we only remark that compacts (finite) elements of $\mathcal{F}$ are principal filters $\sigma\uparrow$: $\omega\uparrow$ is the least element and $\sigma\uparrow \sqcup \tau\uparrow = (\sigma \wedge \tau)\uparrow$. The atomic properties are used to describe the compacts of E, while the step functions in $[L \to D]$ and $[D \to D]$ are described by properties of the shape $\langle \ell : \sigma \rangle$ and $\sigma \to \tau$ respectively.

Even if $\mathcal{F}$ is a solution to equality (not just up to isomorphism) of the domain equation (1), there is a correspondence between the structure of $\mathcal{F}$ and the inverse limit construction: let us stratify the definition of $\mathcal{L}$ by

(i) $\forall i \in I.\, \sigma_i \in \mathcal{L}^{(n)} \Rightarrow \bigwedge_{i\in I} \sigma_i \in \mathcal{L}^{(n)}$,

(ii) $\sigma, \tau \in \mathcal{L}^{(n)} \Rightarrow \alpha, \sigma \to \tau, \langle \ell : \sigma \rangle \in \mathcal{L}^{(n+1)}$.

Clearly $\mathcal{L} = \bigcup_n \mathcal{L}^{(n)}$ (remember that $\bigwedge_{i\in\emptyset} \sigma_i \equiv \omega$). Setting $\leq^{(n)} = \leq \cap \mathcal{L}^{(n)} \times \mathcal{L}^{(n)}$ we define $\mathcal{F}^{(n)}$ as the set of filters w.r.t. $\leq^{(n)}$: it turns out that compacts of $\mathcal{F}^{(n)}$ have the shape $\sigma \uparrow^{(n)}$ for $\sigma \in \mathcal{L}^{(n)}$ (where $\uparrow^{(n)}$ is the upward closure w.r.t. $\leq^{(n)}$) and that $\mathcal{F} = \lim_{\leftarrow} \mathcal{F}^{(n)}$. Moreover the projections $\pi_n : \mathcal{F} \to \mathcal{F}$ are $\pi_n(F) = F_{[n]} = F \cap \mathcal{L}^{(n)}$, and their collection induces a notion of approximation.

The next step is to show in more detail that $\mathcal{F}$ is a model of the (untyped) λ-calculus of records. Strictly speaking this could be derived e.g. by exploiting the above remarks and by using proposition 2.3, but we prefer a more direct and concrete approach. (In the following some proofs are omitted or just sketched).

Lemma 4.3

(i) *For all finite I, $\langle \ell_i : \sigma_i \ ^{(i\in I)} \rangle \neq \omega$;*

(ii) *if $\langle \ell_i : \sigma_i \ ^{(i\in I)} \rangle \leq \tau$ and $\tau \neq \omega$ then there exist $J \subseteq I$ and a family $\{\tau_j \mid j \in J\}$ such that $\tau = \langle \ell_j : \tau_j \ ^{(j\in J)} \rangle$ and $\sigma_j \leq \tau_j$ for all $j \in J$;*

(iii) *if $\bigwedge_{i\in I}(\sigma_i \to \tau_i) \leq \mu \neq \omega$ then $\mu = \bigwedge_{j\in J}(\phi_i \to \psi_i)$, for some property $\bigwedge_{j\in J}(\phi_i \to \psi_i)$, and for all $j \in J$ there exists $I' \subseteq I$ s.t. $\phi_j \leq \bigwedge_{i\in I'} \sigma_i$ and $\bigwedge_{i\in I'} \tau_i \leq \psi_j$.*

Lemma 4.4 *If $F, G \in \mathcal{F}$ then the following are filters:*

(i) *(application) $(F \cdot G) = \{\tau \mid \exists \sigma \in G.\, \sigma \to \tau \in F\}$*

(ii) *(selection) $(F.\ell) = \{\sigma \mid \langle \ell : \sigma \rangle \in F\}$*

(iii) *(empty record) $emp = \omega\uparrow$*

(iv) *(label conditional)*

$$\begin{aligned}(F.\ell := G) = \{\tau \mid \exists I, \sigma_i.\ \tau = &\bigwedge_{i\in I} \langle \ell_i : \sigma_i \rangle \wedge \\ &\forall i \in I.\ (\ell \neq \ell_i \Rightarrow \langle \ell_i : \sigma_i \rangle \in F) \wedge (\ell = \ell_i \Rightarrow \sigma_i \in G)\}\end{aligned}$$

Proof. If $\tau \in F \cdot G$, $\tau \leq \tau'$ and $\tau \neq \omega$ then $\sigma \to \tau \in F$ for some $\sigma \in G$; since $\sigma \to \tau \leq \sigma \to \tau'$ we have $\sigma \to \tau' \in F$ as F is upward closed, then $\tau' \in F \cdot G$.

If $\tau, \tau' \in F \cdot G$ and both are $\neq \omega$ (otherwise $\tau \wedge \tau'$ is trivially in $F \cdot G$), then $\sigma \rightarrow \tau, \sigma' \rightarrow \tau' \in F$ for some $\sigma, \sigma' \in G$. Now, being both F and G closed under meets, $\sigma \rightarrow \tau \wedge \sigma' \rightarrow \tau' \in F$ and $\sigma \wedge \sigma' \in G$. The thesis follows since $\sigma \rightarrow \tau \wedge \sigma' \rightarrow \tau' \leq (\sigma \wedge \sigma') \rightarrow (\tau \wedge \tau')$, and F is upward closed.

If $\tau \in F.\ell$ is $\neq \omega$ and $\tau \leq \tau'$ then $\langle \ell : \tau \rangle \in F$ and $\langle \ell : \tau \rangle \leq \langle \ell : \tau' \rangle \in F$ which is upward closed; then $\tau' \in F.\ell$. If $\tau, \tau' \in F.\ell$ and both are $\neq \omega$, then $\langle \ell : \tau \rangle, \langle \ell : \tau' \rangle \in F$, so that $\langle \ell : \tau \rangle \wedge \langle \ell : \tau' \rangle = \langle \ell : \tau \wedge \tau' \rangle \in F$, hence $\tau \wedge \tau' \in F.\ell$.

If $\tau = \bigwedge_{i \in I} \langle \ell_i : \sigma_i \rangle \in F.\ell := G$ and $\tau \leq \tau' \neq \omega$, then, by lemma 4.3, (ii) $\tau' = \bigwedge_{j \in J} \langle \ell_j : \tau_j \rangle$ for some $J \subseteq I$, where $\sigma_j \leq \tau_j$ for all j; this implies that $\langle \ell_j : \sigma_j \rangle \leq \langle \ell_j : \tau_j \rangle$ which in turn implies that $\langle \ell_j : \tau_j \rangle \in F$ if $\ell_j \neq \ell$ and $\tau_j \in G$ otherwise, being F and G upward closed. Then $\bigwedge_{j \in J} \langle \ell_j : \tau_j \rangle \in F.\ell := G$ by definition. If $\sigma, \tau \in F \cdot \ell := G$ and $\sigma = \bigwedge_{i \in I} \langle \ell_i : \sigma_i \rangle$, $\tau = \bigwedge_{j \in j} \langle \ell_j : \tau_j \rangle$ then $\sigma \wedge \tau = \bigwedge_{i \in I \cup J} \langle \ell_i : \sigma_i \wedge \tau_i \rangle$ which is in $F.\ell := G$ by the closure of F and G under $\wedge$ and by definition.

$\square$

The actual content of the last lemma is that $\mathcal{F}$ is an applicative structure which is closed under application, record selection and update; more precisely:

Theorem 4.5 *$\mathcal{F}$ is a model of the type-free λ-calculus with records.*

Proof. By propostion 2.3 and lemma 4.4. $\square$

4.2 The logical interpretation of types

In the standard semantics types are interpreted as PERs; we show that, if the PERs we choose are CUPERs of the shape R_X, then these give rise to the same domain theoretic interpretation than a filter interpretation.

Definition 4.6 Let $\mathcal{A} = \{\mathcal{A}_G\}_G$ be a collection of subsets of $\{\alpha \mid \alpha \text{ atomic}\}$ indexed by ground types; then $\mathcal{A}$ induces a *hierarchy of languages* for $\mathcal{L}$ which is the family $\{\mathcal{L}_A\}_A$ of subsets of $\mathcal{L}$ indexed by the set of types, such that each $\mathcal{L}_A$ is the least set which:

(i) $\mathcal{A}_G \subseteq \mathcal{L}_G$, for ground G,

(ii) $\omega \in \mathcal{L}_A$ and if $\sigma, \tau \in \mathcal{L}_A$ then $\sigma \wedge \tau \in \mathcal{L}_A$,

(iii) if $\sigma \in \mathcal{L}_A$ and $\tau \in \mathcal{L}_B$ then $\sigma \rightarrow \tau \in \mathcal{L}_{A \rightarrow B}$,

(iv) if $\sigma \in \mathcal{L}_{B_j}$, $A \equiv \{\ell_i : B_i\ ^{(i \in I)}\}$ and $j \in I$ then $\langle \ell_j : \sigma \rangle \in \mathcal{L}_A$.

Provided that any constant α belongs to some $\mathcal{A}_G$, it is easy to show that $\mathcal{L} = \bigcup_A \mathcal{L}_A$: we shall indeed assume this in the sequel.

By definition, if $A \equiv \{\ell_i : B_i\ ^{(i \in I)}\}$ and $j \in I$ then $\langle \ell_j : \omega \rangle \in \mathcal{L}_A$; similarly if $\sigma, \tau \in \mathcal{L}_{B_j}$ then $\langle \ell_j : \sigma \wedge \tau \rangle \in \mathcal{L}_A$.

Note that languages are not upward closed w.r.t. the $\leq$ relation: take $\sigma \equiv \langle \ell_1 : \sigma_1 \rangle$ and $\sigma' \equiv \langle \ell_1 : \sigma_1, \ell_2 : \sigma_2 \rangle$ with $\sigma_i \in \mathcal{L}_{B_i}$, then $\sigma \in \mathcal{L}_{\{\ell_1 : B_1\}}$ and $\sigma' \in \mathcal{L}_{\{\ell_1 : B_1, \ell_2 : B_2\}}$; on the other hand $\sigma' \leq \sigma$ so that $\sigma \rightarrow \tau \leq \sigma' \rightarrow \tau$ for any $\tau \in \mathcal{L}_C$ and type C; now $\sigma \to \tau \in \mathcal{L}_{\{\ell_1:B_1\}\to C}$ but this is not the case for $\sigma' \to \tau$.

Let $\leq_A$ be the restriction of $\leq$ to $\mathcal{L}_A$ and $\mathcal{F}_A$ be the set of filters over $(\mathcal{L}_A, \leq_A)$. We call $\mathcal{F}_A$ the *filter interpretation* of the type A. The subset $\{\sigma{\uparrow_A} | \ \sigma \in \mathcal{L}_A\}$ (where $\uparrow_A$ is the upward closure w.r.t. $\leq_A$) of principal filters over $\mathcal{L}_A$ is a sub-basis w.r.t. $\subseteq$, and in fact it is $K(\mathcal{F}_A)$: it follows that $R_A = R_{\{\sigma\uparrow_A | \sigma \in \mathcal{L}_A\}}$ is well defined. The main theorem of this section shows that filter interpretation and relational interpretation using logical PERs give rise to the same domain theoretic interpretation of types.

Theorem 4.7 *For all type A, $\mathcal{F}_A \simeq \mathcal{F}/R_A$.*

Proof. Let $\Phi_A : \mathcal{F}_A \to \mathcal{F}/R_A$ be defined as $F \mapsto [\widehat{F}]_{R_A}$, where $\widehat{F} = \{\sigma \in \mathcal{L} \mid \exists \tau \in F.\ \sigma \leq \tau\}$. Further define $\Psi_A : \mathcal{F}/R_A \to \mathcal{F}_A$ by $[P]_{R_A} \mapsto P \cap \mathcal{L}_A$ which is well defined. Then $\Phi_A \circ \Psi_A = \mathrm{Id}_{\mathcal{F}/R_A}$ and $\Psi_A \circ \Phi_A = \mathrm{Id}_{\mathcal{F}_A}$.

Suppose that $F, G \in \mathcal{F}_A$ are such that $F \subseteq G$: then $\widehat{F} \subseteq \widehat{G}$, and hence $K(\widehat{F}) \cap \{\sigma \mid \sigma \in \mathcal{L}_A\} \subseteq K(\widehat{G}) \cap \{\sigma \mid \sigma \in \mathcal{L}_A\}$; therefore, by lemma 3.8, $\widehat{F} \leq_{R_A} \widehat{G}$, that is $[\widehat{F}]_{R_A} \leq [\widehat{F}]_{R_A}$.

Vice versa if $P, Q \in \mathcal{F}$ with $P \leq_{R_A} Q$ then by lemma 3.8 $K(P) \cap \{\sigma{\uparrow_A} | \ \sigma \in \mathcal{L}_A\} \subseteq K(Q) \cap \{\sigma{\uparrow_A} | \ \sigma \in \mathcal{L}_A\}$; therefore $P \cap \mathcal{L}_A \subseteq Q \cap \mathcal{L}_A$.

$\square$

4.3 A program logic of the λ-calculus with records

Let us introduce a *program logic*, namely an assignment system of properties to typed terms which is an instance of *intersection type assignment system* and of (though simpler than) *endogenous logic* [3]. A *typed basis* is a set $\Delta = \{x_1 : B_1 : \sigma_1, \ldots, x_n : B_n : \sigma_n\}$ where $\sigma_i \in \mathcal{L}_{B_i}$. Each typed basis Δ determines a context Γ_Δ which is obtained form Δ by forgetting about properties. Then we derive judgments of the form $\Delta \vdash M : A : \sigma$ from the rules in Figure 3.

Assuming that $\alpha \in \mathcal{L}_G$ if $c : G : \alpha$, it is easy to see that if $\Delta \vdash_\Sigma M : A : \sigma$ then $\sigma \in \mathcal{L}_A$ (which is the reason for the third hypothesis in the subsumption rule). Moreover under a restricted use of (ω), namely by checking that $\Gamma_\Delta \vdash_\Sigma M : A$ to deduce $\Delta \vdash M : A : \omega$, we clearly have that $\Delta \vdash_\Sigma M : A : \sigma$ implies $\Gamma_\Delta \vdash_\Sigma M : A$. We henceforth fix a set of subtyping axioms Σ.

The logical interpretation of a term M w.r.t. a type A and an environment ρ is the set of properties in $\mathcal{L}_A$ that can be deduced for M under a typed basis which is consistent with ρ. We might think of a term as a model of its properties, and of the set of these properties as the theory of this model.

Definition 4.8 Let ρ be a mapping from term variables to pairs (A', F) where A' is a type, and $F \in \bigcup_A \mathcal{F}_A$: we say that ρ is a *typed environment* if

$$\forall x.\ \rho(x) = (A, F) \Rightarrow F \in \mathcal{F}_A.$$

$$\frac{x : A : \sigma \in \Delta}{\Delta \vdash x : A : \sigma} \qquad \frac{c : G : \alpha}{\Delta \vdash c : G : \alpha}$$

$$\frac{\Delta, x : A : \sigma \vdash M : B : \tau}{\Delta \vdash \lambda x : A.M : A \to B : \sigma \to \tau} \qquad \frac{\Delta \vdash M : A \to B : \sigma \to \tau \quad \Delta \vdash N : A : \sigma}{\Delta \vdash MN : B : \tau}$$

$$\frac{\Delta \vdash M_i : B_i : \sigma_i \quad \forall i \in I}{\Delta \vdash \{\ell_i = M_i\ ^{(i \in I)}\} : \{\ell_i : B_i\ ^{(i \in I)}\} : \langle \ell_i : \sigma_i\ ^{(i \in I)}\rangle}$$

$$\frac{\Delta \vdash M : \{\ell_i : B_i\ ^{(i \in I)}\} : \langle \ell_j : \sigma \rangle \quad j \in I}{\Delta \vdash M.\ell_j : B_j : \sigma}$$

$$\frac{}{\Delta \vdash M : A : \omega}\ (\omega) \qquad \frac{\Delta \vdash M : A : \sigma \quad \Delta \vdash M : A : \tau}{\Delta \vdash M : A : \sigma \wedge \tau} \qquad \frac{\Delta \vdash M : A : \sigma \quad \sigma \leq_A \tau}{\Delta \vdash M : A : \tau}$$

$$\frac{\Delta \vdash M : A : \sigma \quad \Sigma \vdash A <: B \quad \sigma \in \mathcal{L}_B}{\Delta \vdash M : B : \sigma}$$

Fig. 3. The program logic

If Δ is a typed basis and ρ a typed environment then: $\rho \models \Delta$ if and only if

$$\forall x : B : \tau \in \Delta\ \exists F.\ \rho(x) = (B, F) \wedge \sigma \in F.$$

Then we define the *logical interpretation* of M in type A w.r.t. ρ as the set

$$[\![M : A]\!]^{\mathcal{L}}_{\rho} = \{\sigma \mid \exists \Delta.\ \rho \models \Delta\ \&\ \Delta \vdash_{\Sigma} M : A : \sigma\}.$$

If ρ is a typed environment, then $\widehat{\rho}$ defined by $\widehat{\rho}(x) = \widehat{F}$ whenever $\rho(x) = (A, F)$, is a mapping from term variables to $\mathcal{F}$, namely an environment for the type free calculus. The following lemma relates the logical interpretation of a typed term to the interpretation of its erasure in the filter model of the type free λ-calculus with records.

Lemma 4.9 *For all M and A, if ρ is an environment over $\bigcup_A \mathcal{F}_A$, then*

$$[\![erase(M)]\!]^{\mathcal{F}}_{\widehat{\rho}} \cap \mathcal{L}_A = [\![M : A]\!]^{\mathcal{L}}_{\rho}.$$

Proof. To prove $[\![erase(M)]\!]^{\mathcal{F}}_{\widehat{\rho}} \cap \mathcal{L}_A \subseteq [\![M : A]\!]^{\mathcal{L}}_{\rho}$ we reason by induction on M. A non trivial case is when $M \equiv LN$. If $\tau \in [\![erase(LN)]\!]^{\mathcal{F}}_{\widehat{\rho}}$ then there exists some $\sigma \in [\![erase(N)]\!]^{\mathcal{F}}_{\widehat{\rho}}$ s.t. $\sigma \to \tau \in [\![erase(L)]\!]^{\mathcal{F}}_{\widehat{\rho}}$; since $\sigma \in \mathcal{L}_B$ for some B, then $\sigma \to \tau \in \mathcal{L}_{B \to A}$, so that by induction $\sigma \to \tau \in [\![L : B \to A]\!]^{\mathcal{L}}_{\rho}$ and $\sigma \in [\![N : B]\!]^{\mathcal{L}}_{\rho}$. It follows that there are Δ_0, Δ_1 s.t. $\rho \models \Delta_0, \Delta_1$ and both $\Delta_0 \vdash L : B \to A : \sigma \to \tau$ and $\Delta_1 \vdash N : B : \sigma$. The fact that $\rho \models \Delta_0, \Delta_1$ implies that the type assumed for each variable declared in both of them is the same, hence if we set $x : C : \mu$ to be $x : C : \varphi \wedge \psi$ if $x : C : \varphi \in \Delta_0$ and $x : C : \psi \in \Delta_1$; $x : C : \varphi$ if $x : C : \varphi \in \Delta_0$ and $x \notin \Delta_1$; $x : C : \psi$ if $x : C : \psi \in \Delta_1$ and $x \notin \Delta_0$. Then $\Delta \models \rho$ and $\Delta \vdash L : B \to A : \sigma \to \tau$, $\Delta \vdash N : B : \sigma$. From this we conclude $\Delta \vdash LN : A : \tau$.

To prove $[\![erase(M)]\!]^{\mathcal{F}}_{\widehat{\rho}} \cap \mathcal{L}_A \supseteq [\![M : A]\!]^{\mathcal{L}}_{\rho}$ we show, by induction on derivations, that if $\Delta \models \rho$ and $\Delta \vdash_{\Sigma} M : A : \sigma$ then $\sigma \in [\![erase(M)]\!]^{\mathcal{F}}_{\widehat{\rho}}$ (which is enough, since $\sigma \in \mathcal{L}_A$ by a previous remark about the logical system).

□

Definition 4.10 For any terms M, N, type A and environment ρ define the predicate:

$$[\![M = N : A]\!]^{\mathcal{L}}_{\rho} \Leftrightarrow [\![M : A]\!]^{\mathcal{L}}_{\rho} = [\![M : A]\!]^{\mathcal{L}}_{\rho}.$$

Then we say that M, N are *logically equivalent* w.r.t. A and ρ.

In words, two typed terms are logically the same w.r.t. some type if and only if they cannot be taken apart by any predicate in the language associated to the type, which is deducible for one them. We end this section by stating that the latter model is the same as the PER model determined by the erasure map and the hierarchy $\{R_A\}_A$.

Theorem 4.11 *For all M, N, A and typed environment ρ:*

$$[\![M = N : A]\!]^{\mathcal{L}}_{\rho} \Leftrightarrow [\![erase(M)]\!]^{\mathcal{F}}_{\widehat{\rho}} \; R_A \; [\![erase(M)]\!]^{\mathcal{F}}_{\widehat{\rho}}.$$

Proof. Let $\Phi_A : \mathcal{F}_A \to \mathcal{F}/R_A$ be the isomorphism of theorem 4.7, and Ψ_A its inverse. By lemma 4.9 both $[\![erase(M)]\!]^{\mathcal{F}}_{\widehat{\rho}} R_A \widehat{[\![M : A]\!]^{\mathcal{L}}_{\rho}}$ and $[\![erase(N)]\!]^{\mathcal{F}}_{\widehat{\rho}} R_A \widehat{[\![N : A]\!]^{\mathcal{L}}_{\rho}}$, so that if $[\![M = N : A]\!]^{\mathcal{L}}_{\rho}$ then:

$$\begin{aligned}[][\![erase(M)]\!]^{\mathcal{F}}_{\widehat{\rho}}]_{R_A} &= [\widehat{[\![M : A]\!]^{\mathcal{L}}_{\rho}}]_{R_A} = \Phi_A([\![M : A]\!]^{\mathcal{L}}_{\rho}) \\ &= \Phi_A([\![N : A]\!]^{\mathcal{L}}_{\rho}) = [[\![erase(N)]\!]^{\mathcal{F}}_{\widehat{\rho}}]_{R_A}.\end{aligned}$$

Vice versa, if $[\![erase(M)]\!]^{\mathcal{F}}_{\widehat{\rho}} \; R_A \; [\![erase(M)]\!]^{\mathcal{F}}_{\widehat{\rho}}$ then we have $\Psi_A([[\![erase(M)]\!]^{\mathcal{F}}_{\widehat{\rho}}]_{R_A}) = \Psi_A([[\![erase(N)]\!]^{\mathcal{F}}_{\widehat{\rho}}]_{R_A})$; now $\Psi_A([[\![erase(M)]\!]^{\mathcal{F}}_{\widehat{\rho}}]_{R_A}) = \Psi_A \circ \Phi_A([\![M : A]\!]^{\mathcal{L}}_{\rho}) = [\![M : A]\!]^{\mathcal{L}}_{\rho}$, and similarly $\Psi_A([[\![erase(N)]\!]^{\mathcal{F}}_{\widehat{\rho}}]_{R_A}) = [\![N : A]\!]^{\mathcal{L}}_{\rho}$ so that $[\![M = N : A]\!]^{\mathcal{L}}_{\rho}$ holds.

□

5 Conclusion and further work

In [10] one finds a derivation system of equations $\Gamma \vdash M = N : A$. It is remarked that derivable equations do depend on the type: if $\Gamma \vdash M = N : A$ and $A <: B$ then $\Gamma \vdash M = N : B$ but not vice versa, in general. This is nicely mirrored by interpreting equality A as being related by the PER associated to A, and subtyping by PER inclusion.

By establishing the invariance of predicates under equality we can prove that the logical semantics $[\![M = N : A]\!]_\rho$ provides a sound interpretation of the system.

Similar results are expected when moving to more complex languages of terms and types, like object calculi. These admit an interpretation based on CUPERs (see [1] ch. 14). In this case the complexity of the standard PER description of the object types strongly calls for an alternative treatment of types and subtyping, for which we propose an approach based on domain logic.

Acknowledgments

I wish to thank Felice Cardone for drawing my attention to some key papers on the subject of PERs over realizability structures. My gratitude is also expressed to Steffen van Bakel for useful discussions, and to Mariangiola Dezani for her comments on the final version of the paper.

References

[1] M. Abadi, L. Cardelli, *A Theory of Objects*, Springer 1996.

[2] M. Abadi, G.D. Plotkin, "A Per Model of Polymorphism and Recursive Types", proc. of *IEEE-LICS* 1990, 3355-365.

[3] S. Abramsky, "Domain Theory in Logical Form", *Ann. Pure Appl. Log.* 51, 1991, 1-77.

[4] R. Amadio, "Recursion over Realizability Structures", *Info. Comp.* 91, 1991, 55-85.

[5] H. Barendregt, M. Coppo, M. Dezani-Ciancaglini "A Filter Lambda-Model and the Completeness of Type Assignment", *J. Symb. Log.* 48, 1983, 931-940.

[6] K.B. Bruce, G. Longo, "A Modest Model of Records, Inheritance, and Bounded Quantification", *Info. Comp.* 87, 1990, 1964-240.

[7] K.B. Bruce, J.C. Mitchell, "PER models of subtyping, recursive types and higher-order polymorphism", proc. of *ACM-POPL* 1992.

[8] F. Cardone, "Relational semantics for recursive types and bounded quantification", *LNCS* 372, 1989, 164-178.

[9] U. de'Liguoro, "Characterizing convergent terms in object calculi via intersection types", *LNCS* 2044, 2001.

[10] J.C. Mitchell, *Foundations for Programming Languages*, MIT Press, 1996.