Studying the Fully Abstract Model of PCF within its Continuous Function Model [1]

Achim Jung[2] and Allen Stoughton[3]

Abstract. We give a concrete presentation of the inequationally fully abstract model of PCF as a continuous projection of the inductively reachable subalgebra of PCF's continuous function model.

1 Introduction

As is well known, the continuous function model $\mathcal{E}$ of the applied typed lambda calculus PCF fails to be inequationally fully abstract [Plo77], but PCF has a unique inequationally fully abstract, order-extensional model $\mathcal{F}$ [Mil77, Sto90], where models are required to interpret the ground type ι as the flat cpo of natural numbers. Two attempts at finding connections between $\mathcal{E}$ and $\mathcal{F}$ have been made in the literature.

Mulmuley's idea was to connect complete lattice versions of $\mathcal{E}$ and $\mathcal{F}$ [Mul87]. Using a syntactically defined inductive (inclusive) predicate, he defines an inequationally fully abstract, order-extensional model $\mathcal{F}'$ as the image of a continuous closure (retraction that is greater than the identity function) of the complete lattices version $\mathcal{E}'$ of the continuous function model. The use of complete lattices is essential in this construction, and, e.g., parallel or is mapped to $\top$. Very pleasingly, $\mathcal{F}'$ inherits both its ordering relation and function application operation from $\mathcal{E}'$. Thus some of PCF's operations must be sent by the closure to strictly greater functions. Although the closure isn't a homomorphism between $\mathcal{E}'$ and $\mathcal{F}'$ (since it doesn't preserve application in general), it does preserve the meaning of terms. $\mathcal{F}'$ isn't a combinatory algebra, since all functions of $\mathcal{F}'$ preserve $\top$, and thus the usual axiom for the K combinator cannot hold. Finally, Mulmuley is able to recover $\mathcal{F}$ from $\mathcal{F}'$ simply by removing $\top$ at all types.

A more algebraic connection between $\mathcal{E}$ and $\mathcal{F}$ was subsequently developed by the second author [Sto88]. Here one begins by forming the inductively reachable subalgebra $R(\mathcal{E})$ of $\mathcal{E}$, which in this case simply consists of those elements of $\mathcal{E}$ that are lub's of directed sets of denotable elements. $R(\mathcal{E})$ is then continuously quotiented by a syntactically defined inductive pre-ordering, producing $\mathcal{F}$. Furthermore, in contrast to the situation with $\mathcal{E}'$ and $\mathcal{F}'$ above, there is a continuous homomorphism from $R(\mathcal{E})$ to $\mathcal{F}$.

[1] Appears in M. Bezem and J. F. Groote, editors, *International Conference on Typed Lambda Calculi and Applications*, Lecture Notes in Computer Science, vol. 664, pp. 230–244, Springer-Verlag, 1993.

[2] Fachbereich Mathematik, Technische Hochschule Darmstadt, Schloßgartenstraße 7, D-6100 Darmstadt, Germany, e-mail:`jung@mathematik.th-darmstadt.de`.

[3] School of Cognitive and Computing Sciences, University of Sussex, Falmer, Brighton BN1 9QH, UK, e-mail:`allen@cogs.sussex.ac.uk`.

The purpose of this paper is to give a concrete presentation of this construction of $\mathcal{F}$ from $R(\mathcal{E})$. We define a model $N(\mathcal{E})$ as the image of a syntactically defined continuous projection over $R(\mathcal{E})$, and show that $N(\mathcal{E})$ is inequationally fully abstract and order-extensional, and is thus order-isomorphic to $\mathcal{F}$. As in Mulmuley's construction, the ordering relation of $N(\mathcal{E})$ is inherited from $\mathcal{E}$ (and $R(\mathcal{E})$). On the other hand we prove that the application operation of $N(\mathcal{E})$ cannot be inherited from $\mathcal{E}$. There is, of course, a continuous homomorphism from $R(\mathcal{E})$ to $N(\mathcal{E})$.

In the final section of the paper, we consider the relationship between the full abstraction problem, lambda definability and our presentation of $\mathcal{F}$, and propose a minimal condition that any "solution" to the full abstraction problem should satisfy.

2 Background

The reader is assumed to be familiar with such standard domain-theoretic concepts as (directed) complete partial orders (cpo's), (directed) continuous functions, and ω-algebraic, strongly algebraic (SFP) and consistently complete cpo's.

If X is a subset of a poset P, then we write $\bigsqcup X$ and $\bigsqcap X$ for the lub and glb, respectively, of X in P, when they exist. We abbreviate $\bigsqcup\{x, y\}$ and $\bigsqcap\{x, y\}$ to $x \sqcup y$ and $x \sqcap y$, respectively. We write $\omega_\perp$ for the flat cpo of natural numbers. Given cpo's P and Q, we write $P \xrightarrow[c]{} Q$ for the cpo of all continuous functions from P to Q, ordered pointwise. A cpo P is a *subcpo* of a cpo Q iff $P \subseteq Q$, $\sqsubseteq_P$ is the restriction of $\sqsubseteq_Q$ to P, $\perp_P = \perp_Q$ and $\bigsqcup_P D = \bigsqcup_Q D$ for all directed $D \subseteq P$. A pre-ordering $\leq$ over a cpo P is *inductive* iff $\sqsubseteq_P \subseteq \leq$ and, whenever D is a directed set in $\langle P, \sqsubseteq_P \rangle$ and p is an ub of D in $\langle P, \leq \rangle$, the lub of D in $\langle P, \sqsubseteq_P \rangle$ is $\leq p$.

In the remainder of this section, we briefly recall the definitions and results from [Sto88] that will be required in the sequel.

The reader is assumed to be familiar with many-sorted signatures Σ over sets of sorts S, as well as algebras over such signatures, i.e., Σ-algebras. Signatures are assumed to contain distinguished constants Ω_s at each sort s, which intuitively stand for divergence. Many operations and concepts extend naturally from sets to S-indexed families of sets, in a pointwise manner. For example, if A and B are S-indexed families of sets, then a function $f: A \to B$ is an S-indexed family of functions $f_s: A_s \to B_s$. We will make use of this and other such extensions without explicit comment. We use uppercase script letters ($\mathcal{A}$, $\mathcal{B}$, etc.) to denote algebras and the corresponding italic letters (A, B, etc.) to stand for their carriers.

We write $\mathcal{T}_\Sigma$ (or just $\mathcal{T}$) for the initial (term) algebra, so that T_s is the set of terms of sort s. Given an algebra $\mathcal{A}$ and a term t of sort s, $[\![t]\!]_\mathcal{A}$ (or just $[\![t]\!]$) is the meaning of t in A_s, i.e., the image of t under the unique homomorphism from $\mathcal{T}$ to $\mathcal{A}$. Sometimes we write $t_\mathcal{A}$ (or even just t) for $[\![t]\!]_\mathcal{A}$.

An algebra is *reachable* iff all of its elements are denotable (definable) by terms. A pre-ordering over an algebra is *substitutive* iff it is respected by all of the operations of that algebra. Substitutive equivalence relations are called *congruences*, as usual. The congruence over $\mathcal{T}$ that is induced by an algebra $\mathcal{A}$ is called $\approx_\mathcal{A}$: two terms are congruent when they are mapped to the same element of A. When we say that $c[v_1, \ldots, v_n]$ is a *derived operator* of type $s_1 \times \cdots \times s_n \to s'$, this means that c is a context of sort s' over context variables v_i of sort s_i. We write $c_\mathcal{A}$ for the corresponding *derived operation* over an algebra $\mathcal{A}$.

The reader is also assumed to be familiar with *ordered algebras*, i.e., algebras $\mathcal{A}$ whose carriers are S-indexed families of posets $A_s = \langle A_s, \sqsubseteq_s \rangle$ with least elements $\perp_s$ denoted by the Ω_s constants, and whose operations are monotone functions. Such an algebra is called *complete* when its carrier is a cpo and operations are continuous. A homomorphism over complete ordered algebras is called *continuous* when it is continuous on the underlying cpo's. We write $\mathcal{OT}_\Sigma$ (or just $\mathcal{OT}$) for the initial ordered algebra, which consists of $\mathcal{T}$ with the "Ω-match" ordering: one term is less than another when the second can be formed by replacing occurrences of Ω in the first by terms. The substitutive pre-ordering over $\mathcal{T}$ that is induced by an ordered algebra $\mathcal{A}$ is called $\preceq_{\mathcal{A}}$: one term is less than another when the meaning of the first is less than that of the second in A.

Given complete ordered algebras $\mathcal{A}$ and $\mathcal{B}$, we say that $\mathcal{A}$ is an *inductive subalgebra* of $\mathcal{B}$ (written $\mathcal{A} \preceq \mathcal{B}$) iff $\mathcal{A}$ is a subalgebra of $\mathcal{B}$ and A is a subcpo of B. Given a complete ordered algebra $\mathcal{A}$, we write $R(\mathcal{A})$ for the $\preceq$-least inductive subalgebra of $\mathcal{A}$. Its carrier $R(A)$ is the least set containing all denotable elements and closed under lub's of directed sets; thus we are able to carry out proofs by induction on $R(A)$. A complete ordered algebra $\mathcal{A}$ is *inductively reachable* iff $\mathcal{A} = R(\mathcal{A})$. Hence $R(\mathcal{A})$ is inductively reachable for all complete ordered algebras $\mathcal{A}$. Complete ordered algebras whose carriers are ω-algebraic are inductively reachable iff all of their isolated elements are denotable.

If $\mathcal{A}$ is an algebra and R is a pre-ordering over A, then R is *unary-substitutive* iff all unary-derived operations respect R: for all derived operators $c[v]$ of type $s \rightarrow s'$ and $a, a' \in A_s$, if aR_sa', then $c\langle a\rangle R_{s'} c\langle a'\rangle$. Unary-substitutive pre-orderings can fail to be substitutive; see Lemma 2.2.27 of [Sto88] and Counterexample 4.7.

If $P \subseteq S$, $\mathcal{A}$ is an algebra and R is a pre-ordering over $A|P$, then R^c, the *contextualization* of R, is the relation over A defined by: $a\, R^c_s\, a'$ iff $c\langle a\rangle\, R_p\, c\langle a'\rangle$, for all derived operators $c[v]$ of type $s \rightarrow p$, $p \in P$.

Lemma 2.1 *If $P \subseteq S$, $\mathcal{A}$ is an algebra and R is a pre-ordering (respectively, equivalence relation) over $A|P$, then R^c is the greatest unary-substitutive pre-ordering (respectively, equivalence relation) over $\mathcal{A}$ whose restriction to P is included in R.*

Proof. See Lemma 2.2.25 of [Sto88]. □

Lemma 2.2 *If $P \subseteq S$, $\mathcal{A}$ is a complete ordered algebra and $\leq$ is an inductive pre-ordering over $A|P$, then $\leq^c$ is a unary-substitutive, inductive pre-ordering over $\mathcal{A}$.*

Proof. See Lemma 2.3.14 of [Sto88]. □

Lemma 2.3 (i) *Unary substitutive pre-orderings over reachable algebras are substitutive.*

(ii) *Unary substitutive, inductive pre-orderings over inductively reachable, complete ordered algebras are substitutive.*

Proof. See Lemmas 2.2.29 and 2.3.35 of [Sto88]. □

3 Syntax and Semantics of PCF

For technical simplicity, we have chosen to work with a combinatory logic version of PCF with a single ground type ι, whose intended interpretation is the natural numbers. From the viewpoint of the conditional operations, non-zero and zero are interpreted as true and false, respectively.

The syntax of PCF is specified by a signature, the sorts of which consist of PCF's types. The set of sorts S is least such that

(i) $\iota \in S$, and

(ii) $s_1 \to s_2 \in S$ if $s_1 \in S$ and $s_2 \in S$.

As usual, we let $\to$ associate to the right. Define s^n, for $n \in \omega$, by: $s^0 = s$ and $s^{n+1} = s \to s^n$. The signature Σ over S has binary (application) operators $\cdot_{s_1,s_2}$ of type $(s_1 \to s_2) \times s_1 \to s_2$ for all $s_1, s_2 \in S$, as well as the following constants (nullary operators) for all $s_1, s_2, s_3 \in S$:

(i) Ω_s of sort s,

(ii) K_{s_1,s_2} of sort $s_1 \to s_2 \to s_1$,

(iii) S_{s_1,s_2,s_3} of sort $(s_1 \to s_2 \to s_3) \to (s_1 \to s_2) \to s_1 \to s_3$,

(iv) Y_s of sort $s^1 \to s$,

(v) n of sort ι, for $n \in \omega$,

(vi) Succ and Pred of sort ι^1, and

(vii) If_s of sort $\iota \to s^2$.

We usually abbreviate $x \cdot y$ to $x\,y$, and let application associate to the left.

A *model* $\mathcal{A}$ is a complete ordered algebra such that the following conditions hold:

(i) $A_\iota = \{\bot_\iota, 0_\mathcal{A}, 1_\mathcal{A}, \ldots\}$, where $\bot_\iota \sqsubseteq n_\mathcal{A}$ for all $n \in \omega$ and $n_\mathcal{A}$ and $m_\mathcal{A}$ are incomparable whenever $n \neq m$ (we often confuse A_ι with $\omega_\bot$ below);

(ii) For all $x \in A_{s_1}$ and $y \in A_{s_2}$, $\mathrm{K}_{s_1,s_2}\,x\,y = x$;

(iii) For all $x \in A_{s_1 \to s_2 \to s_3}$, $y \in A_{s_1 \to s_2}$ and $z \in A_{s_1}$, $\mathrm{S}_{s_1,s_2,s_3}\,x\,y\,z = x\,z\,(y\,z)$;

(iv) For all $x \in A_{s^1}$, $\mathrm{Y}_s\,x$ is the least fixed point of the continuous function over A_s that x represents;

(v) For all $x \in A_\iota$, $\mathrm{Succ}\,x$ is equal to $\bot$, if $x = \bot$, and is equal to $x+1$, if $x \in \omega$;

(vi) For all $x \in A_\iota$, $\mathrm{Pred}\,x$ is equal to $\bot$, if $x = \bot$, is equal to 0, if $x = 0$, and is equal to $x - 1$, if $x \in \omega - \{0\}$; and

(vii) For all $x \in A_\iota$ and $y, z \in A_s$, $\mathrm{If}_s\,x\,y\,z$ is equal to $\bot$, if $x = \bot$, is equal to y, if $x \in \omega - \{0\}$, and is equal to z, if $x = 0$.

A model $\mathcal{A}$ is *extensional* iff, for all $x_1, x_2 \in A_{s_1 \to s_2}$, if $x_1\,y = x_2\,y$ for all $y \in A_{s_1}$, then $x_1 = x_2$, and *order-extensional* iff, for all $x_1, x_2 \in A_{s_1 \to s_2}$, if $x_1\,y \sqsubseteq x_2\,y$ for all $y \in A_{s_1}$, then $x_1 \sqsubseteq x_2$. Finally, *morphisms* between models are simply continuous homomorphisms between the complete ordered algebras.

Application is left-strict in all models $\mathcal{A}$ since $\bot_{s_1 \to s_2} \sqsubseteq_{s_1 \to s_2} \mathrm{K}_{s_2,s_1}\,\bot_{s_2}$, and thus $\bot_{s_1 \to s_2}\,x \sqsubseteq_{s_2} \mathrm{K}_{s_2,s_1}\,\bot_{s_2}\,x = \bot_{s_2}$, for all $x \in A_{s_1}$.

The *continuous function model* $\mathcal{E}$ is the unique model $\mathcal{E}$ such that $E_\iota = \omega_\bot$, $E_{s_1 \to s_2} = E_{s_1} \xrightarrow{c} E_{s_2}$ for all $s_1, s_2 \in S$, application is function application and $n_\mathcal{A} = n$ for all $n \in \omega$. $\mathcal{E}$ is clearly order-extensional. The parallel or operation $\mathrm{por} \in E_{\iota^2}$ is defined by: $\mathrm{por}\,x\,y = 1$, if $x \in \omega - \{0\}$ or $y \in \omega - \{0\}$, $\mathrm{por}\,x\,y = 0$, if $x = 0$ and $y = 0$, and $\mathrm{por}\,x\,y = \bot$, otherwise.

Lemma 3.1 *If $\mathcal{A}$ is a model, then so is $R(\mathcal{A})$.*

Proof. Follows easily from the fact that $R(\mathcal{A})$ is an inductive subalgebra of $\mathcal{A}$. $\square$

For $s \in S$, we write I_s for the term $\mathrm{S}_{s,s^1,s}\,\mathrm{K}_{s,s^1}\,\mathrm{K}_{s,s}$ of sort s^1. I is the identity operation in all models. We code lambda abstractions in terms of the S, K and I combinators, in the standard way.

For $s \in S$, define approximations Y_s^n to Y_s of sort $s^1 \to s$ by $\mathrm{Y}_s^0 = \Omega_{s^1 \to s}$ and $\mathrm{Y}_s^{n+1} = \mathrm{S}_{s^1,s,s}\,\mathrm{I}_{s^1}\,\mathrm{Y}_s^n$, so that Y_s^n is an ω-chain in the initial ordered algebra, and thus in all models.

Following [Mil77, BCL85], we can define syntactic projections Ψ^n_s of sort s^1, for all $n \in \omega$ and $s \in S$, by $\Psi^n_\iota = \mathrm{Y}^n_{\iota^1} F$ and $\Psi^n_{s_1 \to s_2} = \lambda xy.\, \Psi^n_{s_2}(x(\Psi^n_{s_1}\, y))$, where F of sort $\iota^1 \to \iota^1$ is $\lambda xy.\, \mathrm{If}\, y\, (\mathrm{Succ}(x(\mathrm{Pred}\, y)))\, 0$. Expanding the abstractions, one can see that the Ψ^n_s form an ω-chain in the initial ordered algebra, and thus in all models. Given a model $\mathcal{A}$, we write A^n_s for the subposet of A_s whose elements are $\{\, \Psi^n_s\, x \mid x \in A_s \,\}$. Clearly $A^n_\iota = \{\perp, 0, 1, \ldots, n-1\}$ for all $n \in \omega$.

Lemma 3.2 (Milner/Berry) *Suppose $\mathcal{A}$ is an extensional model and $s \in S$. The Ψ^n_s represent an ω-chain of continuous projections with finite image over A_s whose lub is the identify function. Hence $x \in A^n_s$ iff $x = \Psi^n_s\, x$, $A^n_s \subseteq A^m_s$ whenever $n \leq m$, and A_s is a strongly algebraic cpo whose set of isolated elements is $\bigcup_{n\in\omega} A^n_s$.*

Proof. The Ψ^n_s obviously represent an ω-chain of continuous functions. Inductions on S suffice to show that they are retractions, have finite image and that their lub is the identity function. But then each Ψ^n_s is less than the identity function. The rest follows easily. □

Lemma 3.3 *Suppose $\mathcal{A}$ is an extensional model and $s \in S$. The Ψ^n_s also represent an ω-chain of continuous projections with finite image over $R(A)_s$ whose lub is the identity function. Hence $R(A)_s$ is a strongly algebraic cpo and, for all $x \in A_s$,*

(i) *if $x \in R(A)_s$, then x is isolated in $R(A)_s$ iff x is isolated in A_s;*
(ii) *if $x \in R(A)_s$ is isolated, then x is denotable; and*
(iii) *$x \in R(A)_s$ iff $\Psi^n\, x$ is denotable for all $n \in \omega$.*

Proof. Everything except (ii) and the "only if" direction of (iii) follows by Lemma 3.2 and the fact that $R(\mathcal{A})$ is an inductive subalgebra of $\mathcal{A}$. (ii) follows by induction on $R(A)_s$, and the "only if" direction of (iii) follows from (ii). □

We write por^2 for $\Psi^2\, \mathrm{por}$. It is easy to see that por^2 and por are interdefinable elements of E_{ι^2}.

Let the equality test Eq of sort ι^2 be

$$\mathrm{Y}(\lambda zxy.\, \mathrm{If}\, x\, (\mathrm{If}\, y\, (z(\mathrm{Pred}\, x)(\mathrm{Pred}\, y))\, 0)\, (\mathrm{Not}\, y)),$$

where Not of sort ι^1 is $\lambda x.\, \mathrm{If}\, x\, 0\, 1$.

For $n \in \omega$, define operators And_n of sort ι^n by: $\mathrm{And}_0 = 1$ and

$$\mathrm{And}_{n+1} = \lambda x y_1 \cdots y_n .\, \mathrm{If}\, x\, (\mathrm{And}_n\, y_1\, \cdots\, y_n)\, 0.$$

Also following [Mil77, BCL85], define glb operators Inf^n_s of sort s^n, for $n \geq 1$, by:

$$\begin{aligned} \mathrm{Inf}^n_\iota &= \lambda x_1 \cdots x_n .\, \mathrm{If}\, (\mathrm{And}_{n-1}\, (\mathrm{Eq}\, x_1\, x_2)\, \cdots\, (\mathrm{Eq}\, x_1\, x_n))\, x_1\, \Omega \\ \mathrm{Inf}^n_{s_1 \to s_2} &= \lambda x_1 \cdots x_n y.\, \mathrm{Inf}^n_{s_2}\, (x_1\, y)\, \cdots\, (x_n\, y). \end{aligned}$$

Lemma 3.4 (Milner) *If $\mathcal{A}$ is an order-extensional model, $x_1, \ldots, x_n \in A_s$, $n \geq 1$ and $s \in S$, then $\mathrm{Inf}^n\, x_1\, \cdots\, x_n$ is the glb of $\{x_1, \ldots, x_n\}$ in A_s—and also in $R(A)_s$, if the x_i are in $R(A)_s$.*

Proof. By induction on S. □

Lemma 3.5 *Suppose $\mathcal{A}$ is an order-extensional model, $X \subseteq A_s$ is nonempty and $s \in S$. Then $\bigsqcup_{n\in\omega}(\bigsqcap(\Psi^n\, X))$ is the glb of X in A_s—and also in $R(A)_s$, if $X \subseteq R(A)_s$. Hence A_s and $R(A)_s$ are consistently complete, ω-algebraic cpo's.*

Proof. Suppose $X \subseteq A_s$ is nonempty. Then the $\bigsqcap(\Psi^n X)$ are well-defined and form an ω-chain by Lemmas 3.2 and 3.4. Let $x \in X$. Then $\bigsqcap(\Psi^n X) \sqsubseteq x$ for all $n \in \omega$, and thus $\bigsqcup_{n\in\omega}(\bigsqcap(\Psi^n X)) \sqsubseteq x$. Now, let y be a lb of X. Then $\Psi^n y \sqsubseteq \bigsqcap(\Psi^n X)$ for all $n \in \omega$, so that $y = \bigsqcup_{n\in\omega}(\Psi^n y) \sqsubseteq \bigsqcup_{n\in\omega}(\bigsqcap(\Psi^n X))$, completing the proof that $\bigsqcup_{n\in\omega}(\bigsqcap(\Psi^n X))$ is the glb of X in A_s. But, if $X \subseteq R(A)_s$, then each $\bigsqcap(\Psi^n X) \in R(A)_s$ by Lemma 3.4, so that $\bigsqcup_{n\in\omega}(\bigsqcap(\Psi^n X)) \in R(A)_s$, as required. The rest follows by Lemmas 3.2 and 3.3. □

The preceding lemma allows us to conclude that both E and $R(E)$ are consistently complete, ω-algebraic cpo's.

Lemma 3.6 *If $\mathcal{A}$ is an order-extensional model, then $\Psi^n(\bigsqcap X) = \bigsqcap(\Psi^n X)$, for all $n \in \omega$ and finite, nonempty $X \subseteq A_s$.*

Proof. By induction on S, using the fact (Lemma 3.4) that finite, nonempty glb's are determined pointwise. □

Since glb's of infinite subsets of E are not always determined pointwise, it is somewhat surprising that we have an infinitary version of the preceding lemma.

Lemma 3.7 *If $\mathcal{A}$ is an order-extensional model, then $\Psi^n(\bigsqcap X) = \bigsqcap(\Psi^n X)$, for all $n \in \omega$ and nonempty $X \subseteq A_s$.*

Proof. For all $x \in X$, we have that $\Psi^n(\bigsqcap X) \sqsubseteq \Psi^n x$. Thus $\Psi^n(\bigsqcap X) \sqsubseteq \bigsqcap(\Psi^n X)$. For the other direction, $\bigsqcap(\Psi^n X) = \bigsqcap(\Psi^n(\Psi^n X)) = \Psi^n(\bigsqcap(\Psi^n X)) \sqsubseteq \Psi^n(\bigsqcap X)$ by Lemma 3.6 and the fact that $\bigsqcap(\Psi^n X) \sqsubseteq \bigsqcap X$. □

Following [Plo80], we say that an n-ary *logical relation* L over a model $\mathcal{A}$, for $n \in \omega$, is an n-ary relation over A such that $\langle x_1, \ldots, x_n\rangle \in L_{s_1 \to s_2}$ iff $\langle x_1\, y_1, \ldots, x_n\, y_n\rangle \in L_{s_2}$ for all $\langle y_1, \ldots, y_n\rangle \in L_{s_1}$. Given such an L and $\mathcal{A}$, we say that an element $x \in A_s$ *satisfies* L iff $\langle x, \ldots, x\rangle \in L_s$.

Lemma 3.8 *Suppose L is an n-ary logical relation over a model $\mathcal{A}$, $s \in S$ and $D_1, \ldots, D_n \subseteq A_s$ are directed sets such that, for all $x_i \in D_i$, $1 \le i \le n$, there are $y_i \in D_i$, $1 \le i \le n$, such that $x_i \sqsubseteq y_i$ for all i and $\langle y_1, \ldots, y_n\rangle \in L_s$. Then $\langle \bigsqcup D_1, \ldots, \bigsqcup D_n\rangle \in L_s$.*

Proof. By induction on S. □

Lemma 3.9 *Suppose L is an n-ary logical relation over a model $\mathcal{A}$. If L is satisfied by Ω_ι, n, for all $n \in \omega$,* Succ, Pred *and* If_ι*, then all elements of $R(A)$ satisfy L.*

Proof. First we must show that the remaining constants satisfy L. The satisfaction of L by K and S at all sorts follows as usual. One shows that Ω satisfies L at all sorts by induction on S, using the fact that application is strict in its left argument. The proof that If satisfies L at all sorts also proceeds by induction on S, using the fact that $\mathrm{If}_{s_1 \to s_2}\, x\, y\, z\, w = \mathrm{If}_{s_2}\, x\, (y\, w)\, (z\, w)$ for all $x \in A_\iota$, $y, z \in A_{s_1 \to s_2}$ and $w \in A_{s_1}$. Finally, the satisfaction of L by Y at all sorts follows using Lemma 3.8.

A simple induction on T then shows that all denotable elements of A satisfy L, following which we use Lemma 3.8 again to show, by induction on $R(A)$, that L is satisfied by all elements of $R(A)$. □

Lemma 3.10 (Plotkin) *There is no* $f \in R(E)_{\iota^2}$ *such that* $f \sqsupseteq \mathrm{por}^2$.

Proof. Following [Sie92], let L be the ternary logical relation over $\mathcal{E}$ such that $\langle x_1, x_2, x_3\rangle \in L_\iota$ iff either $x_i = \bot$ for some i or $x_1 = x_2 = x_3$. It is easy to see that L satisfies the hypotheses of Lemma 3.9, and thus all elements of $R(E)$ satisfy L. Clearly, $\langle 1, \bot, 0\rangle \in L_\iota$ and $\langle \bot, 1, 0\rangle \in L_\iota$. Thus, if there were such an f, then we would have that $\langle x_1, x_2, x_3\rangle \in L_\iota$, where $x_1 = f\,1\,\bot$, $x_2 = f\,\bot\,1$ and $x_3 = f\,0\,0$. But $x_1 = 1$, $x_2 = 1$ and $x_3 = 0$—contradicting the definition of L. □

The following theorem allows us to define the meaning $[\![M]\!] \in \omega_\bot$ of a term M of sort ι to be $[\![M]\!]_{\mathcal{A}}$, for an arbitrary model $\mathcal{A}$.

Theorem 3.11 (Plotkin) *For all models* $\mathcal{A}$ *and* $\mathcal{B}$ *and terms* M *of sort* ι, $[\![M]\!]_{\mathcal{A}} = [\![M]\!]_{\mathcal{B}}$.

Proof. See Theorem 3.1 of [Plo77]. □

We now define notions of program ordering and equivalence for PCF. Define a pre-ordering $\precsim$ over $T|\{\iota\}$ by $M \precsim_\iota N$ iff $[\![M]\!] \sqsubseteq [\![N]\!]$, and let $\approx$ be the equivalence relation over $T|\{\iota\}$ induced by $\precsim$. By Lemmas 2.1 and 2.3(i), $\precsim^c$ is a substitutive pre-ordering over $\mathcal{T}$ and $\approx^c$ is a congruence over $\mathcal{T}$. It is easy to see that $\precsim^c$ induces $\approx^c$. We say that a model $\mathcal{A}$ is *inequationally fully abstract* iff $\preceq_{\mathcal{A}} \; = \; \precsim^c$. From [Plo77], we know that $\mathcal{E}$ is not inequationally fully abstract. On the other hand, by [Mil77], there exists a unique (up to order-isomorphism) inequationally fully abstract, order-extensional model.

Finally, we recall Milner's important result concerning the order-extensional nature of $\precsim^c$ and the extensional nature of $\approx^c$ [Mil77].

Lemma 3.12 (Milner) (i) $\precsim^c_\iota = \precsim_\iota$ *and* $\approx^c_\iota = \approx_\iota$.

(ii) *If* $M_1\,N \precsim^c_{s_2} M_2\,N$ *for all* $N \in T_{s_1}$, *then* $M_1 \precsim^c_{s_1 \to s_2} M_2$.

(iii) *If* $M_1\,N \approx^c_{s_2} M_2\,N$ *for all* $N \in T_{s_1}$, *then* $M_1 \approx^c_{s_1 \to s_2} M_2$.

Proof. See Lemma 4.1.11 of [Cur86]. □

From Lemma 3.12(i), we know that, for all terms M of sort ι, either $M \approx^c_\iota \Omega$ or $M \approx^c_\iota n$ for some $n \in \omega$.

4 Normalization of $R(\mathcal{E})$

In this section, we focus on $\mathcal{E}$. Apart from the counterexamples, however, we could just as well work with any other order-extensional model, such as the bidomains model [BCL85]. We begin by defining semantic analogues of $\precsim^c$ and $\approx^c$.

Definition 4.1 Define an inductive pre-ordering $\leq$ over $E|\{\iota\}$ by $x \leq_\iota y$ iff $x \sqsubseteq y$, and let $\equiv$ be the equivalence relation over $E|\{\iota\}$ induced by $\leq$.

Clearly, $\equiv$ is just the identity relation over $E|\{\iota\}$.

Lemma 4.2 (i) $\leq^c$ *is a unary-substitutive, inductive pre-ordering over* $\mathcal{E}$.

(ii) $\equiv^c$ *is the unary-substitutive equivalence relation over* $\mathcal{E}$ *induced by* $\leq^c$.

(iii) *For all* $M, N \in T_s$, $M \precsim^c_s N$ *iff* $[\![M]\!] \leq^c_s [\![N]\!]$.

(iv) *For all* $M, N \in T_s$, $M \approx^c_s N$ *iff* $[\![M]\!] \equiv^c_s [\![N]\!]$.

Proof. (i) follows from Lemma 2.2, (ii) is by Lemma 2.1 and an easy calculation, (iii) can be shown by another simple calculation, and (iv) follows from (ii) and (iii). □

Lemma 4.3 (i) *The restriction of* $\leq^c$ *to* $R(E)$ *is a substitutive, inductive pre-ordering over* $R(\mathcal{E})$.
(ii) *The restriction of* $\equiv^c$ *to* $R(E)$ *is a congruence over* $R(\mathcal{E})$.

Proof. (i) follows by Lemma 2.3(ii) and the fact that $R(\mathcal{E})$ is an inductive subalgebra of $\mathcal{E}$, and (ii) follows from (i). □

Lemma 4.4 (i) $\leq^c_\iota = \leq_\iota$ *and* $\equiv^c_\iota = \equiv_\iota$.
(ii) *For all* $x_1, x_2 \in R(E)_{s_1 \to s_2}$, *if* $x_1\, y \leq^c x_2\, y$ *for all* $y \in R(E)_{s_1}$, *then* $x_1 \leq^c x_2$.
(iii) *For all* $x_1, x_2 \in R(E)_{s_1 \to s_2}$, *if* $x_1\, y \equiv^c x_2\, y$ *for all* $y \in R(E)_{s_1}$, *then* $x_1 \equiv^c x_2$.

Proof. (i) follows from Lemma 3.12(i). For (ii), it suffices to show that $\Psi^n\, x_1 \leq^c \Psi^n\, x_2$ for all $n \in \omega$, since $\leq^c$ is inductive. But isolated elements of $R(E)$ are denotable, and thus, by Lemma 3.12(ii), it is sufficient to show that $\Psi^n\, x_1\, y \leq^c \Psi^n\, x_2\, y$ for all $y \in R(E)_{s_1}$. But $\Psi^n(x_1(\Psi^n\, y)) \leq^c \Psi^n(x_2(\Psi^n\, y))$ follows from the hypothesis and Lemma 4.3. Finally, (iii) follows from (ii). □

The following term features prominently below and is a generalization of the parallel or tester introduced in [Plo77].

Definition 4.5 Let the term Test of sort $\iota \to \iota \to \iota^2 \to \iota$ be

$$\begin{aligned}\lambda xyf.\,&\mathrm{If}\,(\mathrm{Eq}\,(f\,1\,y)\,1)\\ &(\mathrm{If}\,(\mathrm{Eq}\,(f\,x\,1)\,1)\\ &\quad(\mathrm{If}\,(f\,0\,0)\,\Omega\,0)\\ &\quad\Omega)\\ &\Omega.\end{aligned}$$

Lemma 4.6 *For all* $f \in E_{\iota^2}$, $\mathrm{Test}\perp\perp f$ *is* 0, *if* $f \sqsupseteq \mathrm{por}^2$, *and* $\perp$, *otherwise.* □

The following is a counterexample to $\equiv^c$ (and thus $\leq^c$) being substitutive.

Counterexample 4.7 $\mathrm{Test}\perp\perp \equiv^c \perp$, *but* $\mathrm{Test}\perp\perp\mathrm{por} \not\equiv^c \perp\mathrm{por}$.

Proof. By Lemmas 4.4, 4.6 and 3.10, we have $\mathrm{Test}\perp\perp \equiv^c \perp$. But $\mathrm{Test}\perp\perp\mathrm{por} = 0$, and thus $\mathrm{Test}\perp\perp\mathrm{por} \not\equiv^c \perp\mathrm{por}$. □

We do, however, have:

Lemma 4.8 (i) *For all* $x_1, x_2 \in E_{s_1 \to s_2}$ *and* $y \in R(E)_{s_1}$, *if* $x_1 \leq^c x_2$, *then* $x_1\, y \leq^c x_2\, y$.
(ii) *For all* $x \in R(E)_{s_1 \to s_2}$ *and* $y_1, y_2 \in E_{s_2}$, *if* $y_1 \leq^c y_2$, *then* $x\, y_1 \leq^c x\, y_2$.

Proof. For (i), since application is continuous and $\leq^c$ is inductive, it suffices to show $x_1\, y \leq^c x_2\, y$ when y is isolated. But this follows since all isolated elements of $R(E)_{s_1}$ are denotable and $\leq^c$ is unary-substitutive. (ii) follows similarly. □

The following result shows that we cannot allow x_1, x_2 to range over $E_{s_1 \to s_2}$ in parts (ii) and (iii) of Lemma 4.4. This raises the question (which we leave unanswered) of when nondenotable elements are related by $\leq^c$ and $\equiv^c$.

Counterexample 4.9 *Define* $G_1, G_2 \in E_{\iota^2 \to \iota^2}$ *by*

$$G_1 = \lambda f.\mathrm{If}\,(f\,0\,0)\,\mathrm{por}\,(\lambda xy.\,\mathrm{Test}\,\Omega\,\Omega\,f), \qquad G_2 = \lambda f.\mathrm{If}\,(f\,0\,0)\,\mathrm{por}\,\Omega.$$

Then $G_1\,f \equiv^c G_2\,f$ *for all* $f \in R(E)_{\iota^2}$*, but* $G_1 \not\equiv^c G_2$*.*

Proof. It is easy to see that $G_1\,f \equiv^c G_2\,f$ for all $f \in R(E)_{\iota^2}$. But $c\langle G_1 \rangle = 0$ and $c\langle G_2 \rangle = \bot$, where the derived operator $c[v]$ of type $(\iota^2 \to \iota^2) \to \iota$ is $v\,(v(\lambda xy.\,1))\,\Omega\,\Omega$. Thus $G_1 \not\equiv^c G_2$. □

We are now ready to define our continuous projection over $R(E)$.

Definition 4.10 The function norm: $R(E) \to R(E)$ is defined by

$$\mathrm{norm}_s\,x = \bigsqcap\{\,x' \in R(E)_s \mid x' \equiv^c x\,\}.$$

By Lemma 3.7, $\Psi^n(\mathrm{norm}\,x) = \bigsqcap\{\,\Psi^n\,x' \mid x' \equiv^c x \text{ and } x' \in R(E)_s\,\}$ for all $n \in \omega$ and $x \in R(E)_s$, $s \in S$. We write $x \sqsubseteq\!\equiv^c y$ for $x \sqsubseteq y$ and $x \equiv^c y$.

Lemma 4.11 *If* X *is a finite subset of* $R(E)_s$ *and* $x' \in X$ *is such that* $x' \leq^c x$ *for all* $x \in X$*, then* $x' \equiv^c \bigsqcap X$*.*

Proof. By induction on S. □

Lemma 4.12 *Let* $x, y \in R(E)_s$*,* $s \in S$*, and* $n \in \omega$*.*

(i) $\mathrm{norm}\,x \sqsubseteq x$.
(ii) $\mathrm{norm}\,x \equiv^c x$.
(iii) *If* $x \sqsubseteq\!\equiv^c \mathrm{norm}\,y$*, then* $x = \mathrm{norm}\,y$.
(iv) $x \leq^c y$ *iff* $\mathrm{norm}\,x \sqsubseteq \mathrm{norm}\,y$.
(v) $x \equiv^c y$ *iff* $\mathrm{norm}\,x = \mathrm{norm}\,y$.
(vi) *If* $x \sqsubseteq y$*, then* $\mathrm{norm}\,x \sqsubseteq \mathrm{norm}\,y$.
(vii) $\mathrm{norm}(\mathrm{norm}\,x) = \mathrm{norm}\,x$.
(viii) $\mathrm{norm}(\Psi^n\,x) \sqsubseteq\!\equiv^c \Psi^n(\mathrm{norm}\,x)$.
(ix) $\Psi^n(\mathrm{norm}(\Psi^n\,x)) = \mathrm{norm}(\Psi^n\,x)$.
(x) $\mathrm{norm}\,x = \bigsqcup_{n \in \omega} \mathrm{norm}(\Psi^n\,x)$.
(xi) norm_s *is continuous.*

Proof. (i) Immediate from the reflexivity of $\equiv^c$.

(ii) By Lemma 4.11, we have that $\Psi^n\,x \equiv^c \bigsqcap\{\,\Psi^n\,x' \mid x' \equiv^c x \text{ and } x' \in R(E)_s\,\} \sqsubseteq \mathrm{norm}\,x$, for all $n \in \omega$. Thus $x \leq^c \mathrm{norm}\,x$, since $\leq^c$ is inductive. The result then follows by (i).

(iii) If $x \sqsubseteq\!\equiv^c \mathrm{norm}\,y$, then $x \equiv^c \mathrm{norm}\,y \equiv^c y$ by (ii), so that $\mathrm{norm}\,y \sqsubseteq x$. But then $x = \mathrm{norm}\,y$, since $x \sqsubseteq \mathrm{norm}\,y$.

(iv) The "if" direction follows from (ii) and the fact that $\sqsubseteq_s \subseteq \leq^c_s$. For the "only if" direction, suppose that $x \leq^c y$. Let $y' \in R(E)_s$ be such that $y' \equiv^c y$. Then $x \leq^c y'$, so that $x \sqcap y' \equiv^c x$ by Lemma 4.11. But then $\mathrm{norm}\,x \sqsubseteq x \sqcap y' \sqsubseteq y'$. Thus $\mathrm{norm}\,x \sqsubseteq \mathrm{norm}\,y$.

(v) Immediate from (iv).

(vi) Follows from (iv), since $\sqsubseteq_s \subseteq \leq^c_s$.

(vii) Follows by (ii) and (v).

(viii) Follows by (i), (ii) and (v).

(ix) Since $\mathrm{norm}(\Psi^n\, x) \equiv^c \Psi^n\, x$, we have $\Psi^n(\mathrm{norm}(\Psi^n\, x)) \equiv^c \Psi^n(\Psi^n\, x) = \Psi^n\, x \equiv^c \mathrm{norm}(\Psi^n\, x)$, and thus $\Psi^n(\mathrm{norm}(\Psi^n\, x)) \equiv^c \mathrm{norm}(\Psi^n\, x)$. The result then follows by (iii), since $\Psi^n(\mathrm{norm}(\Psi^n\, x)) \sqsubseteq \mathrm{norm}(\Psi^n\, x)$.

(x) By (vi) and (viii), $\mathrm{norm}(\Psi^n\, x) \sqsubseteq \mathrm{norm}\, x$ and $\bigsqcup_{n\in\omega} \mathrm{norm}(\Psi^n\, x) \geq^c \Psi^n(\mathrm{norm}\, x)$, for all $n \in \omega$. Thus $\bigsqcup_{n\in\omega} \mathrm{norm}(\Psi^n\, x) \sqsubseteq\equiv^c \mathrm{norm}\, x$, since $\leq^c$ is inductive. The result then follows by (iii).

(xi) Follows from (x). □

Lemma 4.13 $\mathrm{norm}(\mathrm{Test}\perp\perp) = \perp$.

Proof. Follows from Counterexample 4.7. □

Lemma 4.14 norm_ι *is the identity function on* $R(E)_\iota$.

Proof. Immediate by Lemma 4.4(i). □

The following counterexample shows that Lemma 4.12(viii) cannot be strengthened to an identity.

Counterexample 4.15 $\mathrm{norm}(\Psi^2(\mathrm{Test}\,2\,2)) \neq \Psi^2(\mathrm{norm}(\mathrm{Test}\,2\,2))$.

Proof. Let the term A of sort ι^2 be

$$
\begin{array}{l}
\lambda xy.\, \mathrm{If}\,(\mathrm{And}_2\,(\mathrm{Eq}\,x\,1)\,(\mathrm{Eq}\,y\,2)) \\
\qquad 1 \\
\qquad (\mathrm{If}\,(\mathrm{And}_2\,(\mathrm{Eq}\,x\,2)\,(\mathrm{Eq}\,y\,1)) \\
\qquad\quad 1 \\
\qquad\quad (\mathrm{If}\,(\mathrm{And}_2\,(\mathrm{Eq}\,x\,0)\,(\mathrm{Eq}\,y\,0))\,0\,\Omega)),
\end{array}
$$

so that $A \sqsubseteq \mathrm{por}^2$. Since $\mathrm{Test}\,2\,2\,A = 0$, it follows that $(\mathrm{norm}(\mathrm{Test}\,2\,2))A = 0$, and thus that $(\mathrm{norm}(\mathrm{Test}\,2\,2))\mathrm{por}^2 = 0$. But then

$$\Psi^2\,(\mathrm{norm}(\mathrm{Test}\,2\,2))\,\mathrm{por} = \Psi^2((\mathrm{norm}(\mathrm{Test}\,2\,2))\mathrm{por}^2) = \Psi^2\,0 = 0,$$

showing that $\Psi^2(\mathrm{norm}(\mathrm{Test}\,2\,2)) \neq \perp$. On the other hand, it is easy to show that $\Psi^2(\mathrm{Test}\,2\,2) = \mathrm{Test}\perp\perp$, and thus $\mathrm{norm}(\Psi^2(\mathrm{Test}\,2\,2)) = \perp$ by Lemma 4.13. □

In preparation for three key counterexamples, we now define the following or operations of sort ι^2, where the "L", "R" and "D" stand for "Left", "Right" and "Divergent", respectively:

$$
\begin{aligned}
\mathrm{LOr} &= \lambda xy.\, \mathrm{If}\,x\,1\,(\mathrm{If}\,y\,1\,0) \\
\mathrm{ROr} &= \lambda xy.\, \mathrm{If}\,y\,1\,(\mathrm{If}\,x\,1\,0) \\
\mathrm{DOr} &= \lambda xy.\, \mathrm{If}\,x\,(\mathrm{If}\,y\,\Omega\,1)\,(\mathrm{If}\,y\,1\,0).
\end{aligned}
$$

Lemma 4.16 *There is no* $h \in R(E)_{\iota\to\iota\to\iota^2\to\iota}$ *such that*

$$h\perp\perp\mathrm{por} = \perp, \qquad h\,0\perp\mathrm{LOr} = 0, \qquad h\perp 0\,\mathrm{ROr} = 0, \qquad h\,0\,0\,\mathrm{DOr} = 0.$$

Proof. Suppose, toward a contradiction, that such an h does exist.

Let L be the 4-ary logical relation over $\mathcal{E}$ such that $\langle x_1, x_2, x_3, x_4\rangle \in L_\iota$ iff $\{x_1, x_2, x_3, x_4\} \subseteq \{\bot, n\}$ for some $n \in \omega$ and, if $x_1 = \bot$, then one of x_2, x_3, x_4 is also $\bot$. Clearly, Ω_ι and all $n \in \omega$ satisfy L. Furthermore, Succ and Pred satisfy L since it is satisfied by all elements of E_{ι^1}. Finally, it is easy to show that L is satisfied by If_ι. Hence h satisfies L, by Lemma 3.9.

Next, we show that $\langle \mathrm{por}, \mathrm{LOr}, \mathrm{ROr}, \mathrm{DOr}\rangle \in L_{\iota^2}$. Suppose that $\langle x_1, x_2, x_3, x_4\rangle \in L_\iota$ and $\langle y_1, y_2, y_3, y_4\rangle \in L_\iota$. We must show that $\langle z_1, z_2, z_3, z_4\rangle \in L_\iota$, where $z_1 = \mathrm{por}\, x_1\, y_1$, $z_2 = \mathrm{LOr}\, x_2\, y_2$, $z_3 = \mathrm{ROr}\, x_3\, y_3$ and $z_4 = \mathrm{DOr}\, x_4\, y_4$. Clearly, each $z_i \in \{\bot, 0, 1\}$. Furthermore, if $z_i = 0$ for some i, then both x_i and y_i must be 0, so that no x_j or y_j is a nonzero element of ω, and thus no $z_j = 1$. Now, suppose that $z_1 = \bot$. We must show that one of z_2, z_3, z_4 is $\bot$. Either x_1 or y_1 must be $\bot$, and we consider the case when $x_1 = \bot$, the other case being dual. Since LOr and DOr are strict in their first arguments, if $x_i = \bot$ for some $i \in \{2, 4\}$, then $z_i = \bot$. Otherwise, we must have that $x_3 = \bot$ and $x_2 = x_4 \neq \bot$. Now, if $y_3 \in \{\bot, 0\}$, then $z_3 = \bot$. Otherwise, $y_3 \in \omega - \{0\}$ and $y_1, y_2, y_4 \in \{\bot, y_3\}$. But then $y_1 = \bot$ (otherwise $z_1 = 1$), and thus either $y_2 = \bot$ or $y_4 = \bot$. Since DOr is also strict in its second argument, if $y_4 = \bot$, then $z_4 = \bot$. Otherwise, $y_2 = \bot$ and $y_4 = y_3$. Now, if $x_2 = 0$, then $z_2 = \bot$. Otherwise, we have that $x_2 = x_4 \in \omega - \{0\}$. But then $z_4 = \bot$, since both $x_4, y_4 \in \omega - \{0\}$.

Summarizing, we have that h satisfies L and $\langle \mathrm{por}, \mathrm{LOr}, \mathrm{ROr}, \mathrm{DOr}\rangle \in L_{\iota^2}$. Furthermore, $\langle \bot, 0, \bot, 0\rangle \in L_\iota$ and $\langle \bot, \bot, 0, 0\rangle \in L_\iota$, so that $\langle z_1, z_2, z_3, z_4\rangle \in L_\iota$, where $z_1 = h\,\bot\,\bot\,\mathrm{por}$, $z_2 = h\,0\,\bot\,\mathrm{LOr}$, $z_3 = h\,\bot\,0\,\mathrm{ROr}$ and $z_4 = h\,0\,0\,\mathrm{DOr}$. But $z_1 = \bot$ and $z_2 = z_3 = z_4 = 0$, contradicting the definition of L. □

The following counterexample shows that application is not preserved by norm.

Counterexample 4.17 $\mathrm{norm}(\mathrm{Test}\,\bot) \neq (\mathrm{norm}\,\mathrm{Test})(\mathrm{norm}\,\bot)$.

Proof. By Lemma 4.4(iii) and Counterexample 4.7, we have that

$$\mathrm{Test}\,\bot \equiv^c \lambda y.\,\mathrm{If}\; y\,(\mathrm{Test}\,\bot\, y)\,(\mathrm{Test}\,\bot\, y),$$

so that $(\mathrm{norm}(\mathrm{Test}\,\bot))\,\bot\,\mathrm{por} = \bot$. Since $\mathrm{norm}\,\bot = \bot$, it is thus sufficient to show that $h\,\bot\,\bot\,\mathrm{por} \neq \bot$, where $h = \mathrm{norm}\,\mathrm{Test}$. But

$$h\,0\,\bot\,\mathrm{LOr} = 0, \qquad h\,\bot\,0\,\mathrm{ROr} = 0, \qquad h\,0\,0\,\mathrm{DOr} = 0,$$

since $h \equiv^c \mathrm{Test}$, and thus $h\,\bot\,\bot\,\mathrm{por} \neq \bot$ by Lemma 4.16. □

Since $\mathrm{norm}((\mathrm{norm}\,\mathrm{Test})(\mathrm{norm}\,\bot)) = \mathrm{norm}(\mathrm{Test}\,\bot)$, it follows from the preceding counterexample that the image of norm is not closed under application.

Counterexample 4.18 *There is no* $\mathrm{norm}' \in R(E)_{(\iota\to\iota^2\to\iota)^1}$ *such that* $\mathrm{norm}'\, x = \mathrm{norm}\, x$ *for all* $x \in R(E)_{\iota\to\iota^2\to\iota}$.

Proof. Suppose, toward a contradiction, that such a norm' does exist. Then, for all $y \in R(E)_\iota$,

$$\mathrm{Test}\, y \equiv^c \mathrm{norm}(\mathrm{Test}\, y) = \mathrm{norm}'(\mathrm{Test}\, y) = (\lambda y.\, \mathrm{norm}'(\mathrm{Test}\, y))\, y,$$

so that $\text{Test} \equiv^c \lambda y.\, \text{norm}'(\text{Test}\, y)$. Then,

$$\begin{array}{rcl}(\text{norm}\,\text{Test})(\text{norm}\,\bot) & = & (\text{norm}(\lambda y.\, \text{norm}'(\text{Test}\, y)))\,\bot \\ & \sqsubseteq\equiv^c & (\lambda y.\, \text{norm}'(\text{Test}\, y))\,\bot \\ & = & \text{norm}'(\text{Test}\,\bot) \\ & = & \text{norm}(\text{Test}\,\bot).\end{array}$$

But then $(\text{norm}\,\text{Test})(\text{norm}\,\bot) = \text{norm}(\text{Test}\,\bot)$, contradicting Counterexample 4.17. $\square$

The following counterexample shows that denotable elements can be contextually equivalent to nondenotable ones.

Counterexample 4.19 $h \equiv^c \text{Test}$ *does not imply that* $h \in R(E)$.

Proof. Let $h \in E_{\iota\to\iota\to\iota^2\to\iota}$ be $\lambda xy.\, \text{If}\,(\text{pcon}\, x\, y)\,(\text{Test}\, x\, y)\,\Omega$, where the parallel convergence operation $\text{pcon} \in E_{\iota^2}$ is defined by: $\text{pcon}\, x\, y = 1$, if $x \neq \bot$ or $y \neq \bot$, and $\text{pcon}\, x\, y = \bot$, otherwise. Then $h \notin R(E)$, by Lemma 4.16. It remains to show that $h \equiv^c \text{Test}$.

In the remainder of the proof, we work in the result of adding to PCF a constant PCon of sort ι^2 whose interpretation is pcon. All of the results preceding Lemma 4.16 hold for the extended language, with the exception of Lemma 3.9. This lemma can be repaired, however, by adding PCon to the list of constants in its hypothesis. The logical relation defined in the proof of Lemma 3.10 is also satisfied by PCon and thus this lemma is true for the extended language. (The original proof that parallel or is not definable from parallel convergence can be found in [Abr90]. [Plo77] did not show that the continuous function model of the extended language was not inequationally fully abstract.)

It is sufficient to show $h \equiv^c \text{Test}$, and, since $h \in R(E)$, this will be a consequence of showing that $h\, x\, y \equiv^c \text{Test}\, x\, y$ for all $x, y \in R(E)_\iota$. If $x \neq \bot$ or $y \neq \bot$, then $h\, x\, y = \text{Test}\, x\, y$. But $\text{Test}\,\bot\,\bot \equiv^c \bot$ was shown in Counterexample 4.7. $\square$

Although we were able to solve negatively the question of whether norm preserves application, the following problem is still open.

Open Problem 4.20 *Is* $\text{norm}\,\sigma = \sigma$ *for all constants* $\sigma \in \Sigma$? *In particular, is* $(\text{norm}\,\text{K})\, x\, y$ *ever strictly less than* x?

Now, we are able to show how the unique inequationally fully abstract, order-extensional model lives inside the continuous function model.

Definition 4.21 We define the ordered algebra $N(\mathcal{E})$ as follows. For all $s \in S$, $N(E)_s$ consists of $\text{norm}\, R(E)_s$, ordered by the restriction of $\sqsubseteq_{E_s}$ to $\text{norm}\, R(E)_s$. For all $x \in N(E)_{s_1\to s_2}$ and $y \in N(E)_{s_1}$, $x \cdot_{N(\mathcal{E})} y = \text{norm}(x \cdot_{\mathcal{E}} y)$. For all constants σ, $\sigma_{N(\mathcal{E})} = \text{norm}\,\sigma_{\mathcal{E}}$.

$N(E)$ is a subcpo of $R(E)$ and $N(\mathcal{E})$ is well-defined, since norm is strict and continuous.

Theorem 4.22 *$N(\mathcal{E})$ is an order-extensional model and* norm *is a surjective morphism from $R(\mathcal{E})$ to $N(\mathcal{E})$.*

Proof. $N(\mathcal{E})$ is a complete ordered algebra by the preceding remark and the continuity of norm. Condition (i) of the definition of model holds by Lemma 4.14, and the remaining conditions can be shown using Lemma 4.12(ii) and (v) and (for condition (iv)) the continuity of norm. For the order-extensionality of $N(\mathcal{E})$, suppose that $x_1, x_2 \in N(E)_{s_1 \to s_2}$ are such that $x_1 \cdot_{N(\mathcal{E})} y \sqsubseteq x_2 \cdot_{N(\mathcal{E})} y$ for all $y \in N(E)_{s_1}$. Then, for all $y \in R(E)_{s_1}$,

$$\mathrm{norm}(x_1 \cdot_{\mathcal{E}} y) = x_1 \cdot_{N(\mathcal{E})} \mathrm{norm}\, y \sqsubseteq x_2 \cdot_{N(\mathcal{E})} \mathrm{norm}\, y = \mathrm{norm}(x_2 \cdot_{\mathcal{E}} y),$$

and thus $x_1 \cdot_{\mathcal{E}} y \leq^c x_2 \cdot_{\mathcal{E}} y$. But then $x_1 \leq^c x_2$ by Lemma 4.4(ii), so that $x_1 = \mathrm{norm}\, x_1 \sqsubseteq \mathrm{norm}\, x_2 = x_2$. Finally, norm is a surjective morphism from $R(\mathcal{E})$ to $N(\mathcal{E})$ because of the way $N(\mathcal{E})$ was defined. □

By Lemma 3.5, we know that $N(E)$ is a consistently complete, ω-algebraic cpo.

Lemma 4.23 *For all terms M, $[\![M]\!]_{N(\mathcal{E})} = \mathrm{norm}[\![M]\!]_{\mathcal{E}}$.*

Proof. A consequence of norm being a morphism from $R(\mathcal{E})$ to $N(\mathcal{E})$. □

Theorem 4.24 *$N(\mathcal{E})$ is inequationally fully abstract.*

Proof. Follows from Lemmas 4.2(iii) and 4.23. □

5 Full Abstraction and Lambda Definability

There appears to be no clear definition of what the "full abstraction problem" for PCF really is. By Milner's construction [Mil77] we know that there is a unique inequationally fully abstract, order-extensional model $\mathcal{F}$ (which we refer to below as *the fully abstract model*) that is made up out of Scott-domains of continuous (set-theoretic) functions. Why are we not satisfied? The answer to this question, as one often reads, is that Milner's model is "syntactic in nature". The same words are used against Mulmuley's description [Mul87] of the fully abstract model. What people vaguely imagine is that there ought to be a description of $\mathcal{F}$ using cpo's enriched with some additional structure (order-theoretic, topological, etc.) which allows the domains of the fully abstract model to be constructed without recourse to the syntax of PCF. Of course, nobody can specify what this additional structure will be or should be. Stated this way, there is no chance to falsify this research programme, in the sense that there is no way one can prove a result saying that there is no "semantic" presentation of $\mathcal{F}$.

We would therefore like to give a weak but precise minimal condition that a semantic solution of the full abstraction problem should satisfy. Namely, it should allow us to *effectively* construct the finite domains F_s of the fully abstract model $\mathcal{F}$ of *Finitary PCF*, i.e., the variant of PCF in which the sort ι is interpreted as the booleans ($\{\bot, 0, 1\}$) rather than the natural numbers. (The results of this paper can be trivially adapted to Finitary PCF.) Clearly, neither Milner's nor Mulmuley's constructions achieve this. On the other hand, even if we can find such an algorithm for presenting $\mathcal{F}$, we may still be unsatisfied with it as a semantic description.

The results of this paper give one of the simplest descriptions of the fully abstract model to date. In order to satisfy the above condition, all one needs to find is an algorithm that decides whether an element of E is denotable, since then one will be able to effectively present $R(E)$ and thus $N(E)$.

The problem of deciding which elements of a model are definable in the case of the typed lambda calculus (without constants) and the full set-theoretic type hierarchy based on a finite set is known as "Plotkin's conjecture". (It seems that the term was coined by Statman in his 1982 paper [Sta82]. We do not know whether Plotkin ever considered the question nor whether he ever conjectured anything.) The "conjecture" is that the problem is decidable. We prefer to call it the "lambda definability problem" (cf. [JT93]). This problem can be studied in all kinds of contexts, and it certainly makes sense to ask whether it is decidable which elements of E are denotable. We refer to this as the lambda definability problem for Finitary PCF.

Since a positive solution to the lambda definability problem for Finitary PCF will mean that $N(\mathcal{E})$ and thus $\mathcal{F}$ are effectively presentable, it is natural to ask whether the converse is also true. We conjecture that it is.

Acknowledgments

Part of this work was done while the second author was a Guest Researcher at the Technische Hochschule Darmstadt in September, 1991.

References

[Abr90] S. Abramsky. The lazy lambda calculus. In D. L. Turner, editor, *Research Topics in Functional Programming*, pages 65–116. Addison-Wesley, 1990.

[BCL85] G. Berry, P.-L. Curien and J.-J. Lévy. Full abstraction for sequential languages: the state of the art. In M. Nivat and J. C. Reynolds, editors, *Algebraic Methods in Semantics*, pages 89–132. Cambridge University Press, 1985.

[Cur86] P.-L. Curien. *Categorical Combinators, Sequential Algorithms and Functional Programming*. Research Notes in Theoretical Computer Science. Pitman/Wiley, 1986.

[JT93] A. Jung and J. Tiuryn. A new characterization of lambda definability. In *International Conference on Typed Lambda Calculi and Applications*, Lecture Notes in Computer Science. Springer-Verlag, 1993.

[Mil77] R. Milner. Fully abstract models of typed λ-calculi. *Theoretical Computer Science*, 4:1–22, 1977.

[Mul87] K. Mulmuley. *Full Abstraction and Semantic Equivalence*. MIT Press, 1987.

[Plo77] G. D. Plotkin. LCF considered as a programming language. *Theoretical Computer Science*, 5:223–256, 1977.

[Plo80] G. D. Plotkin. Lambda-definability in the full type hierarchy. In J. Seldin and J. Hindley, editors, *To H. B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism*, pages 363–374. Academic Press, 1980.

[Sie92] K. Sieber. Reasoning about sequential functions via logical relations. In M. P. Fourman, P. T. Johnstone and A. M. Pitts, editors, *Applications of Categories in Computer Science*, volume 177 of *LMS Lecture Note Series*, pages 258–269. Cambridge University Press, 1992.

[Sta82] R. Statman. Completeness, invariance and λ-definability. *Journal of Symbolic Logic*, 47:17–26, 1982.

[Sto88] A. Stoughton. *Fully Abstract Models of Programming Languages*. Research Notes in Theoretical Computer Science. Pitman/Wiley, 1988.

[Sto90] A. Stoughton. Equationally fully abstract models of PCF. In M. Main, A. Melton, M. Mislove and D. Schmidt, editors, *Fifth International Conference on the Mathematical Foundations of Programming Semantics*, volume 442 of *Lecture Notes in Computer Science*, pages 271–283. Springer-Verlag, 1990.