On Classical PCF, Linear Logic and the MIX rule∗

Shahin Amini1 and Thomas Ehrhard2

1 PPS, UMR 7126, Univ Paris Diderot, Sorbonne Paris Cité
F-75205 Paris, France shahnie@gmail.com
2 CNRS, PPS, UMR 7126, Univ Paris Diderot, Sorbonne Paris Cité
F-75205 Paris, France thomas.ehrhard@pps.univ-paris-diderot.fr

Abstract

We study a classical version of PCF from a semantic point of view. We define a general notion of model based on categorical models of Linear Logic, in the spirit of earlier work by Girard, Regnier and Laurent. We give a concrete example based on the relational model of Linear Logic, that we present as a non-idempotent intersection type system, and we prove an Adequacy Theorem using ideas introduced by Krivine. Following Danos and Krivine, we also consider an extension of this language with a MIX construction introducing a form of must non-determinism; in this language, a program of type integer can have more than one value (or no value at all, raising an error). We propose a refinement of the relational model of classical PCF in which programs of type integer are single valued; this model rejects the MIX syntactical constructs (and the MIX rule of Linear Logic).

1998 ACM Subject Classification F.3.2 Semantics of Programming Languages, F.3.3 Studies of Program Constructs, F.4.1 Mathematical Logic

Keywords and phrases lambda-calculus, linear logic, classical logic, denotational semantics

Digital Object Identifier 10.4230/LIPIcs.xxx.yyy.p

Introduction

Since the fundamental discovery by Timothy Griffin [6] that the call/cc primitive of the programming language Scheme can be typed by a non intuitionistic classical tautology (the Law of Peirce), much work has been dedicated to the study of the computational content of classical proofs. A particular attention has been devoted to the denotational semantics of classically extended lambda-calculi. Among the notions of categorical model defined for this purpose, we can isolate two main concepts.

= Models extending the standard categorical setting of CCC’s for interpreting usual λ-calculi through a CPS translation. In these models of classical λ-calculi, terms are interpreted in a CCC of negated objects, that is objects of shape ΣP where P is an object of a cartesian and cocartesian category P where Σ is a distinguished baseable object, see [14] and its generalization [13] where the category of negated objects is axiomatized. In this latter paper, it is also shown that Parigot’s λµ-calculus [12] is the internal language of such categories and completeness of this notion of model for λµ theories is proven in [8], enforcing further its universality.

= Models based on Linear Logic (LL) and polarities. The basic idea of such models is to divide objects (formulas) in two categories, exchanged by linear negation: negative objects and positive ones: this is the basic idea of Girard’s LC logical system. The main
feature of these polarized objects is that each one carries its own structural morphisms (weakening and contraction). In [5], positive objects are correlation spaces (commutative $\otimes$-comonoids), and it is crucially used that all such comonoids are coalgebras for the $\;!$ functor, because, in the considered coherence space model, this exponential is the free commutative $\otimes$-comonoid functor. An obvious generalization, implicitly considered in [10], is to interpret directly positive formulas as $\;!$-coalgebras instead of $\otimes$-comonoids without making further assumptions on the $\;!$ comonad.

As recorded in Section 2.2, all models of the second class can be seen as models of the first one, but it is not clear that such a presentation is always particularly enlightening. We rather believe that, depending on the considered concrete model, one presentation might be more convenient than the other; for example, the classical PCF game model of [7] and the polarized HO game model of [9] are suitably described using the first notion. In the present paper, we focus on models for which the second presentation is more convenient.

To define the general interpretation of classical PCF, we assume therefore to be given a categorical model of classical LL $\mathcal{L}$ with a few additional features: an object $N$ of natural numbers that we simply assume to be the coproduct of $\omega$ copies of $1$ (the tensor unit) in the linear category $\mathcal{L}$ as well as a fix-point operator at each object, in the Kleisli category of $\;!$, which is a CCC.

Our classical version of PCF is based on the $\lambda\mu$-calculus described in [1] which features three kinds of expressions: terms, stacks (or continuation) and commands which are pairings of terms and stacks. The operational semantics is given as a rewriting system on commands. Stacks can be duplicated or erased during computations, hence types are interpreted as $\;!$-coalgebras and stacks as coalgebra morphisms. Notice that the interpretation of types corresponds to the linear negation of the usual PCF interpretation of types: roughly speaking, we interpret $\sigma \Rightarrow \tau$ as $\!(P^\perp) \otimes Q$ where $P$ and $Q$ are the interpretations of $\sigma$ and $\tau$. We retrieve the ordinary interpretation simply by taking the linear negation of this positive translation\(^1\).

In particular, the positive interpretation of the ground type of natural numbers has to be such a coalgebra. Therefore, the most tempting choice, which would be to take $[\iota] = N^\perp$, is not possible (it turns out that $N$ is canonically a $\;!$-coalgebra, but its dual is not). So we simply set $[\iota] = \!(N^\perp)$, the free $\;!$-coalgebra generated by $N^\perp$. We do not know if, depending on the concrete model under consideration, more “economical” choices of $\;!$-coalgebras would have been possible; this is certainly an interesting research direction. We describe the corresponding interpretation of expressions and state a general soundness theorem: this interpretation of commands is invariant under reduction.

Then we consider the simplest example of this situation, where we take for $\mathcal{L}$ the category of sets and relations, which is a well known model of LL. We provide a description of the interpretation of expressions in this particular model by means of an intersection typing system, in the spirit of [3, 15]. We then prove an Adequacy Theorem: if a command has a non-empty interpretation (that is, if it is typeable in this intersection typing system) then its reduction terminates. The proof is based on a standard reducibility method (see [2] for instance). This model accommodates a natural extension of classical PCF by a parallel

\(^1\) Due to the symmetries of a categorical model of LL, this is more an aesthetic choice of design than anything else. We could have preferred a negative interpretation, representing stacks as $?\!$-algebras and using $\!?$ instead of $\otimes$ for interpreting contexts. The two interpretations would have been the same, up to linear transposition. The positive interpretation is in some sense closer to usual $\lambda$-calculus intuitions because, when interpreting expressions, the context remains on the argument side of morphisms.
composition of commands corresponding to the MIX rule of LL as in [2].

In classical PCF with MIX, a normalizing command without free variables but with free names of ground type \( \nu \) can yield an arbitrary amount of unrelated natural numbers on each of its free names (outputs). Without MIX syntactical constructs such a command will produce exactly one natural number on exactly one of its outputs. This can be checked syntactically, but we also build a simple refinement of the relational model of LL which does not accommodate the MIX rule and gives a direct semantic account of this uniqueness of values for classical PCF without MIX. This means that this crucial property will remain true in any extension of classical PCF which can be interpreted in this model.

1 Classical PCF

Types are given by the following BNF syntax: \( \sigma ::= \nu | \sigma \Rightarrow \sigma | \sigma \times \sigma \).

The expressions of our language are those of the \( \overline{\lambda} \mu \)-calculus [1], extended with fix-points and primitives for dealing with integers. Let \( x, y \ldots \) be variables and \( \alpha, \beta \ldots \) be names. Terms \( t \), command \( c \) and stacks \( \pi \) are defined as follows (with \( n \in \mathbb{N} \)):

\[
\begin{align*}
   t &::= x | n | \lambda x^n t | \langle t, t \rangle | \mu \alpha^n c | \text{fix } x^n t \\
   \pi &::= \alpha | \text{arg}(t) \cdot \pi | \text{pr}_1 \cdot \pi | \text{pr}_2 \cdot \pi | \text{succ } \cdot \pi | \text{pred } \cdot \pi | \text{if}(t, t) \cdot \pi
\end{align*}
\]

The main reason for considering also product types is that, in the categorical interpretation of Section 2.1, these types will introduce !-coalgebras which are not isomorphic to coalgebras of shape \(!X\), that is, which are not free. We give now the typing rules, which correspond to a sequent calculus. \( \Gamma \)'s are typing variable contexts and \( \Delta \)'s are typing name contexts. We give rules for term typing judgments \( \Gamma \vdash t : \sigma \mid \Delta \), stack typing judgments \( \Gamma \vdash \pi : \sigma \mid \Delta \) and command typing judgments \( \Gamma \vdash c \mid \Delta \).

\[
\begin{array}{c}
   \Gamma, x : \sigma \vdash x : \sigma \mid \Delta \\
  \hline
   \Gamma \vdash t : \sigma \mid \Delta \\

   \hline
   \Gamma \vdash s : \sigma \mid \Delta \\
   \hline
   \hline
   \Gamma \vdash \langle s, t \rangle : \sigma \times \tau \mid \Delta \\
  \hline
   \hline
   \Gamma \vdash \lambda x^n t : \sigma \Rightarrow \tau \mid \Delta \\
  \\
  \hline
  \hline
   \Gamma \vdash \text{arg}(t) \cdot \pi : \sigma \Rightarrow \tau \mid \Delta \\
   \hline
  \hline
   \Gamma \vdash \text{pr}_1 \cdot \pi : \sigma \times \tau \mid \Delta \\
  \\
  \hline
  \hline
   \Gamma \vdash \text{pr}_2 \cdot \pi : \sigma \times \tau \mid \Delta \\
  \\
  \hline
  \hline
   \Gamma \vdash \mu \alpha^n c : \sigma \mid \Delta
\end{array}
\]

We define a deterministic reduction relation \( \rightarrow \) on processes.

\[
\begin{align*}
   \langle \lambda x^n s \rangle \ast \text{arg}(t) \cdot \pi &\rightarrow s[t/x] \ast \pi \\
   \langle s, t \rangle \ast \text{pr}_1 \cdot \pi &\rightarrow s \ast \pi \\
   \langle s, t \rangle \ast \text{pr}_2 \cdot \pi &\rightarrow t \ast \pi \\
   (\mu \alpha^n c) \ast \pi &\rightarrow c[\pi/\alpha] \\
   (\text{fix } x^n t) \ast \pi &\rightarrow t[\text{fix } x^n t/x] \ast \pi \\
   n \ast \text{succ } \cdot \pi &\rightarrow n + 1 \ast \pi \\
   \text{if}(t_1, t_2) \cdot \pi &\rightarrow t_1 \ast \pi \\
   n + 1 \ast \text{if}(t_1, t_2) \cdot \pi &\rightarrow t_2 \ast \pi
\end{align*}
\]

\( \rightarrow \) Proposition 1 (Subject Reduction). Assume that \( \Gamma \vdash c \mid \Delta \) and \( c \rightarrow c' \). Then \( \Gamma \vdash c' \mid \Delta \).
The proof is a straightforward case analysis involving a Substitution Lemma.

A typical example of classical PCF program is the call/cc operator $t = \lambda f^\uparrow (i \Rightarrow \sigma) \Rightarrow i \cdot \mu \alpha^\tau \cdot (f * \arg(\lambda x^\mu \beta^\tau (x * \alpha))) \cdot \alpha)$ which satisfies $\vdash t : (\langle i \Rightarrow \sigma \rangle \Rightarrow i) \Rightarrow i$ (its type is an instance of the well known Peirce classical tautology). When fed with an argument $s$ such that $\vdash s : (i \Rightarrow \sigma) \Rightarrow i$ (a functional), $t$ tests whether this functional returns directly a value (and then $t$ returns that value) or uses its argument (a function) by providing it with a natural number $n$, and in that case $t$ returns $n$. This choice between two options is implemented by a contraction (the two occurrences of $\alpha$).

**The MIX extension.** We consider also an extension of this classical version of PCF where we add two new constructs: a command $err$ and, given commands $c$ and $d$, a command $c \parallel d$. A similar extension of an untyped classical calculus has already been considered in [2]. It is more naturally introduced at the level of commands in the present $\lambda \mu$ setting. These constructions obey the following typing rules

$$
\Gamma \vdash err \mid \Delta \\
\Gamma \vdash c \mid \Delta \\
\Gamma \vdash d \mid \Delta
$$

We then extend the operational semantics of the calculus by adding the following reduction rules for commands.

$$
err \Rightarrow c \Rightarrow c \\
c \parallel err \Rightarrow c \\
c \Rightarrow c' \\
\parallel c \parallel d' \Rightarrow c \parallel d' \\
d \Rightarrow d' \\
\parallel c \Rightarrow d' \Rightarrow d'
$$

The resulting calculus on commands (with the other reduction rules given in Section 1) clearly satisfies the diamond property, the strongest form of confluence. These constructions can be extended as term constructions, available at all types. Simply set $err^\sigma = \mu \alpha^\tau \cdot err$ and $(s \parallel t) = \mu \alpha^\tau \cdot (s \parallel \alpha \parallel t \parallel \alpha)$. The term $s \parallel t$ is as a parallel composition of $s$ and $t$ enriching the language with a form of must non-determinism. It allows eg. to write $3 \parallel 7$, a closed term of type $\iota$, whose value is at the same time 3 and 7.

**Almost closed commands.** We come back to our initial version of classical PCF, without the MIX constructs. A name context $\Delta = (\alpha_1 : \tau_1, \ldots, \alpha_k : \tau_k)$ is ground if $\tau_j = \iota$ for each $j$. We say that a command $c$ is almost closed if $\vdash c \mid \Delta$ where $\Delta$ is ground.

An almost closed command is very similar to a closed term of type $\iota$ in ordinary PCF. The difference is twofold: first an almost closed command can have more than one output (one for each name in the name context), and second its outputs are named, simply to make them usable. Of course in ordinary PCF, naming the unique output of a term is useless.

**Proposition 2.** Assume that $c$ is almost closed and normal. Then $c = n \cdot \alpha$ for some $n \in \mathbb{N}$ and name $\alpha$.

The proof is a simple case analysis. Consider an almost closed command $c$ such that, say, $\vdash c \mid \alpha_1 : \iota, \ldots, \alpha_k : \iota$. Then either the $\Rightarrow$ reduction of $c$ does not terminate, or it ends with a normal almost closed command, which must be of shape $n \cdot \alpha_i$. In other words, the reduction of $c$ computes the final value $n$ and chooses the output on which this value will be issued.

The notion of almost closed command still makes sense in classical PCF with MIX. The difference is that normal forms are now MIX compositions of elementary command $n \cdot \alpha_i$. One can obtain for instance $(0 \cdot \alpha_1) \parallel ((3 \cdot \alpha_2) \parallel (7 \cdot \alpha_1))$ whose effect is to produce the value \( on output \( = 7\) on output $\alpha_1$ and nothing on the other outputs.
2  Linear logic based denotational semantics

The kind of denotational models we are interested in in this paper are those induced by a model of LL, in the spirit of Girard’s seminal work [5] further developed eg. in [10]. We first recall the general categorical definition of a model of LL implicit in [4], our main reference here is [11] to which we also refer for the rich bibliography on this general topic. A model of LL consists of:

- A category \( \mathcal{L} \).
- A symmetric monoidal structure \((\otimes, 1, \lambda, \rho, \alpha, \sigma)\) which is assumed to be closed: \( \otimes \) is a functor \( \mathcal{L}^2 \to \mathcal{L} \), an object of \( \mathcal{L} \), \( \lambda_X \in \mathcal{L}(1 \otimes X, X) \), \( \rho_X \in \mathcal{L}(X \otimes 1, X) \), \( \alpha_{X,Y,Z} \in \mathcal{L}((X \otimes Y) \otimes Z, X \otimes (Y \otimes Z)) \) and \( \sigma_{X,Y} \in \mathcal{L}(X \otimes Y, Y \otimes X) \) are natural isomorphisms satisfying coherence diagrams that we do not record here. We use \( X \to Y \) for the object of linear morphisms from \( X \) to \( Y \), ev for the evaluation morphism which belongs to \( \mathcal{L}((X \to Y) \otimes X, Y) \) and cur for the map \( \mathcal{L}(Z \otimes X, Y) \to \mathcal{L}(Z, X \to Y) \).
- An object \( \perp \) of \( \mathcal{L} \) such that the natural morphism \( \eta_X = \text{cur}((\text{ev} \ \sigma_{X \to \perp, X})) \in \mathcal{L}(X, (X \to \perp) \to \perp) \) be an iso for each object \( X \) (one says that \( \mathcal{L} \) is a *-autonomous category).

The category \( \mathcal{L} \) is assumed to be cartesian. We use \( \top \) for the terminal object, \& for the cartesian product and \( \pi_i \) for the projections. It follows by *-autonomy that \( \mathcal{L} \) has also all finite coproducts.

We are also given a comonad \( !_- : \mathcal{L} \to \mathcal{L} \) with counit \( \text{der}_X \in \mathcal{L}(!(X),X) \) (called dereliction) and comultiplication \( \text{dig}_X \in \mathcal{L}(!(X),!!X) \) (called digging).

And a strong symmetric monoidal structure for the functor \( !_- \), from the symmetric monoidal category \( (\mathcal{L}, \&), (\mathcal{L}, \otimes) \) to the symmetric monoidal category \( (\mathcal{L}, \&), (\mathcal{L}, \otimes) \). This means that we are given an iso \( m^{(0)} \in \mathcal{L}(1, !\top) \) and a natural isomorphism \( m^{(2)}_{X,Y} \in \mathcal{L}(!(X \otimes Y), !(X \& Y)) \) which satisfy a series of commutations that we do not record here (they are often called Stedly isomorphisms). We also require a coherence condition relating \( m^{(2)} \) and \( \text{dig} \).

It follows that we can define a lax symmetric monoidal structure for the functor \( !_- \) from the symmetric monoidal category \( (\mathcal{L}, \otimes) \) to itself, that is a natural morphism \( \mu^{(n)}_{X_1,\ldots,X_n} \in \mathcal{L}(!(X_1 \otimes \cdots \otimes X_n), !(X_1 \& \cdots \& X_n)) \) satisfying some coherence conditions.

We use \( ? \) for the “De Morgan dual” of \( !_- : ?X = (!(X^\perp))^\perp \) and similarly for morphisms. It is a monad on \( \mathcal{L} \) with unit \( \text{der}_X \) and multiplication \( \text{dig}_X \) defined straightforwardly, using \( \text{der}_Y \) and \( \text{dig}_Y \).

The Eilenberg-Moore category. It is then standard to define the category \( \mathcal{L}' \) of \( !_- \)-coalgebras. An object of this category is a pair \( P = (P, h_P) \) where \( P \in \text{Obj}(\mathcal{L}) \) and \( h_P \in \mathcal{L}(P, !_- P) \) is such that \( \text{der}_P \; h_P = \text{id} \) and \( \text{dig}_P \; h_P = !_P \; h_P \).

Given two such coalgebras \( P \) and \( Q \), an element of \( \mathcal{L}'(P, Q) \) is an \( f \in \mathcal{L}(P, Q) \) such that \( h_Q f = !_P \; h_P \). Identities and composition are defined in the obvious way. The functor \( !_- \) can then be seen as a functor from \( \mathcal{L} \) to \( \mathcal{L}' \): this functor maps \( X \) to the coalgebra \( (!(X), \text{dig}_X) \) and a morphism \( f \in \mathcal{L}(X, Y) \) to the coalgebra morphism \( !_P \; h_P = !_P \; h_P \). It is right adjoint to the forgetful functor \( U : \mathcal{L}' \to \mathcal{L} \) which maps a \( !_- \)-coalgebra \( P \) to \( P \) and a morphism \( f \) to itself.

The object \( 1 \) of \( \mathcal{L} \) induces an object of \( \mathcal{L}' \), still denoted as \( 1 \), namely \( (1, \mu^{(0)}) \).

Given two objects \( P \) and \( Q \) of \( \mathcal{L}' \), we can define an object \( P \otimes Q \) of \( \mathcal{L}' \) setting \( P \otimes Q = P \otimes Q \) and \( h_{P \otimes Q} = \mu^{(2)}_{P, Q}(h_P \otimes h_Q) \).

Any object \( P \) of \( \mathcal{L}' \) can be equipped with a canonical structure of commutative comonoid.

This means that we can define a morphism \( w_P \in \mathcal{L}'(P, 1) \) and a morphism \( c_P \in \mathcal{L}'(P, P \otimes P) \) which satisfy the following commutations.
One can check a stronger property, namely that 1 is the terminal object of \( \mathcal{L} \) and that \( P \otimes Q \) (equipped with projections defined in the obvious way using \( w_Q \) and \( w_P \)) is the cartesian product of \( P \) and \( Q \) in \( \mathcal{L} \); the proof consists of rather long computations, see [11].

It is also important to notice that, if the family \( (P_i)_{i \in I} \) of objects of \( \mathcal{L} \) is such that the family \( (P_i)_{i \in I} \) admits a coproduct \( (\bigoplus_{i \in I} P_i, (\in i)_{i \in I}) \) in \( \mathcal{L} \), then it admits a coproduct in \( \mathcal{L} \). This coproduct \( P = \bigoplus_{i \in I} P_i \) is defined as \( P = \bigoplus_{i \in I} P_i \), with a structure map \( h_P \) defined by the fact that, for each \( i \in I \), \( h_P \in I = \in i h_{P_i} \).

**Object of natural numbers and conditional.** We assume also that in \( \mathcal{L} \), the family of objects \( (X_n)_{n \in \mathbb{N}} \) such that \( X_n = 1 \) for each \( n \), has a coproduct \( \mathbb{N} \). For each \( n \in \mathbb{N} \), we use \( \pi \) for the \( n \)th injection \( \pi \in \mathcal{L}(1, \mathbb{N}) \). Using the obvious isomorphism between \( \mathbb{N} \) and \( 1 \oplus \mathbb{N} \), we define two morphisms \( \text{succ}, \text{pred} \in \mathcal{L}(\mathbb{N}, \mathbb{N}) \) such that \( \text{succ} \pi = n + 1 \), \( \text{pred} \pi = 0 \) and \( \text{pred} \pi = 1 = \pi \). Let \( \sigma \) be an object of \( \mathcal{L} \). Let \( \sigma \) be defined as the following composition in \( \mathcal{L} \):

\[
1 \oplus !X \oplus !X \xrightarrow{1 \otimes \text{der}_X \otimes w_X} 1 \otimes X \otimes 1 \xrightarrow{\varphi} X
\]

where \( \varphi \) is the obvious iso. Let \( \sigma \in \mathcal{L}(\mathbb{N} \otimes !X \otimes !X, \mathbb{X}) \) be defined as the following composition of morphisms in \( \mathcal{L} \):

\[
\mathbb{N} \otimes !X \otimes !X \xrightarrow{w_X \otimes w_X \otimes \text{der}_X} 1 \otimes 1 \otimes X \xrightarrow{\psi} X
\]

where \( \psi \) is the obvious iso. Observe that we use the fact that \( \mathbb{N} \) has a canonical structure of \( ! \)-coalgebra (as a sum of coalgebras) inducing the weakening morphism \( w_{\mathbb{N}} \). It is the only place where this property is used.

Using these two morphisms, the isomorphism between \( \mathbb{N} \) and \( 1 \oplus \mathbb{N} \) and the fact that \( \otimes \) commutes with sums (because it is a left adjoint), we define a morphism \( \sigma \in \mathcal{L}(\mathbb{N} \otimes !X \otimes !X, \mathbb{X}) \) such that the two following diagrams commute

\[
\begin{array}{ccc}
1 \otimes !X \otimes !X & \xrightarrow{1 \otimes \text{der}_X \otimes w_X} & 1 \otimes X \otimes 1 \\
\varphi & \downarrow & \psi \\
N \otimes !X \otimes !X & \xrightarrow{\pi} & X
\end{array}
\hspace{1cm}
\begin{array}{ccc}
1 \otimes !X \otimes !X & \xrightarrow{1 \otimes \text{der}_X \otimes w_X} & 1 \otimes 1 \otimes X \\
\pi + 1 & \otimes !X \otimes !X & \xrightarrow{\pi} & X
\end{array}
\]

**Fix-point operators.** For any object \( X \), we assume to be given a morphism \( \text{fix}_X \in \mathcal{L}(!(1X \to X), \mathbb{X}) \) such that the following diagram commutes

\[
\begin{array}{ccc}
!(1X \to X) & \xrightarrow{\text{fix}_X} & !(1X \to X) \otimes !(1X \to X) \\
\text{fix}_X & \xrightarrow{\text{der}_X \otimes \text{fix}_X} & !(1X \to X) \otimes !X
\end{array}
\]

\[
\begin{array}{ccc}
!(1X \to X) & \xrightarrow{\text{fix}_X} & !(1X \to X) \otimes !(1X \to X) \\
\text{fix}_X & \xrightarrow{\text{der}_X \otimes \text{fix}_X} & !(1X \to X) \otimes !X
\end{array}
\]

\[
\begin{array}{ccc}
!(1X \to X) & \xrightarrow{\text{fix}_X} & !(1X \to X) \otimes !(1X \to X) \\
\text{fix}_X & \xrightarrow{\text{der}_X \otimes \text{fix}_X} & !(1X \to X) \otimes !X
\end{array}
\]

\[
\begin{array}{ccc}
!(1X \to X) & \xrightarrow{\text{fix}_X} & !(1X \to X) \otimes !(1X \to X) \\
\text{fix}_X & \xrightarrow{\text{der}_X \otimes \text{fix}_X} & !(1X \to X) \otimes !X
\end{array}
\]

\[
\begin{array}{ccc}
!(1X \to X) & \xrightarrow{\text{fix}_X} & !(1X \to X) \otimes !(1X \to X) \\
\text{fix}_X & \xrightarrow{\text{der}_X \otimes \text{fix}_X} & !(1X \to X) \otimes !X
\end{array}
\]

\[
\begin{array}{ccc}
!(1X \to X) & \xrightarrow{\text{fix}_X} & !(1X \to X) \otimes !(1X \to X) \\
\text{fix}_X & \xrightarrow{\text{der}_X \otimes \text{fix}_X} & !(1X \to X) \otimes !X
\end{array}
\]
MIX in Linear Logic  The categorical setting introduced so far allows to interpret the MIX-free version of classical PCF. In order to interpret the MIX extension of Section 1, it suffices to assume that \( \perp \) is equipped with a structure of commutative \( \otimes \)-monoid (this is an additional structure of the model). If we think of \( \perp \) as an object of scalars, which is a natural intuition since \( \perp \) is the dualizing object, this means that these scalars have a multiplication, a natural intuition again if we have linear algebra in mind. We use \( \operatorname{mix}^0 \in \mathcal{L}(1, \perp) \) for the unit of this monoid and \( \operatorname{mix}^2 \in \mathcal{L}(\perp \otimes \perp, \perp) \) for its multiplication. When this structure is added, we say that \( \mathcal{L} \) is a model of LL with MIX.

2.1 Interpreting classical PCF

With any type \( \sigma \), we associate an object \([\sigma]_!\) of \( \mathcal{L} \). We set \( [t] = (!([N^+]), [\sigma \Rightarrow \tau] = !([\sigma]^+) \otimes [\tau] \) and \([\sigma \times \tau] = [\sigma] \otimes [\tau] \). Observe that \([t]^+ = \#N\), which will be the target object for the interpretation of terms of type \( t \). Let \( \Gamma = (x_1 : \sigma_1, \ldots, x_n : \sigma_n) \) be a variable context and \( \Delta = (\alpha_1 : \tau_1, \ldots, \alpha_k : \tau_k) \) be a name context, then we define two objects of \( \mathcal{L} \) by \([\Gamma] = ![\sigma^+] \otimes \cdots \otimes ![\sigma_n^+]\) and \([\Delta] = [\tau_1] \otimes \cdots \otimes [\tau_k] \). With any term \( t \) such that \( \Gamma \vdash t : \sigma \mid \Delta \), we associate \([t]_{\Gamma,\Delta} \in \mathcal{L}(\Gamma \otimes [\Delta], [\sigma]^+) \) with any command \( c \) such that \( \Gamma \vdash c \mid \Delta \) we associate \([c]_{\Gamma,\Delta} \in \mathcal{L}(\Gamma \otimes [\Delta], [\sigma]^+) \) and with any stack \( \pi \) such that \( \Gamma \mid \pi : \sigma \vdash \Delta \) we associate \([\pi]_{\Gamma,\Delta} \in \mathcal{L}(\Gamma \otimes [\Delta], [\sigma]) \). Do well notice that this latter is required to be a coalgebra morphism whereas the two others are not.

We give first the interpretation of terms. The interpretation of a variable \([x]_{\Gamma,\Delta} \) as the following composition of morphisms in \( \mathcal{L} \) (where \( \varphi \) is an obvious isomorphism):

\[
\begin{array}{c}
[\Gamma] \otimes ![\sigma^+] \otimes ![\Delta] \\
\xrightarrow{\varphi([\sigma]_!) \otimes \operatorname{der}_{[\sigma]_!} \otimes [\Delta]} \\
1 \otimes [\sigma]^+ \otimes 1 \xrightarrow{[\varphi]^{-1}} [\sigma]^+
\end{array}
\]

where \( \varphi \) is the obvious isomorphism. Let \( n \in \mathbb{N} \), remember that \( \pi \in \mathcal{L}(1, \mathbb{N}) \) so that \( ?n \in \mathcal{L}(1, \mathbb{N}) \). We define \([n]_{\Gamma,\Delta} \) as the following composition of morphisms in \( \mathcal{L} \) (where \( \varphi \) is an obvious isomorphism):

\[
\begin{array}{c}
[\Gamma] \otimes [\Delta] \\
\xrightarrow{\varphi([\sigma]_! \otimes [\Delta])} \\
1 \xrightarrow{d_1'} 1 \xrightarrow{?n} \mathbb{N}
\end{array}
\]

Assume next that \( \Gamma, x : \sigma : \tau \vdash t : \tau \mid \Delta \) so that we have \([t]_{\Gamma,\Delta} \in \mathcal{L}(\Gamma \otimes [\Delta], ![\sigma]^+, ![\tau]^+)\) where \( \varphi \) is an isomorphism induced by the symmetric monoidal structure of \( \mathcal{L} \). We set \([\lambda x^\sigma \tau]_{\Gamma,\Delta} = \operatorname{cur}([\mu]_{\Gamma,\Delta} \varphi) \in \mathcal{L}(\Gamma \otimes [\Delta], ![\sigma]^+, ![\tau]^+)\) and we have \( ![\sigma]^+ \to ![\tau]^+ = (!([\tau]_!) \otimes ![\tau])^+ = ![\sigma \Rightarrow \tau]^+ \) up to canonical iso associated with the *-autonomous structure.

Assume that \( \Gamma \vdash s : \sigma \mid \Delta \) and \( \Gamma \vdash t : \tau \mid \Delta \) so that we have \([s]_{\Gamma,\Delta} \in \mathcal{L}(\Gamma \otimes [\Delta], [\sigma]^+) \) and \([t]_{\Gamma,\Delta} \in \mathcal{L}(\Gamma \otimes [\Delta], [\tau]^+) \). So we set \([s, t]_{\Gamma,\Delta} = ([s]_{\Gamma,\Delta}, [t]_{\Gamma,\Delta}) \in \mathcal{L}(\Gamma \otimes [\Delta], [\sigma]^+ \otimes [\tau]^+) \) which has the prescribed codomain since \([\sigma]^+ \otimes [\tau]^+ = [\sigma \times \tau]^+ \).

Assume that \( \Gamma \vdash c : \alpha : \sigma : \Delta \) so that we have \([c]_{\Gamma,\Delta} \in \mathcal{L}(\Gamma \otimes [\Delta], [\sigma], \perp) \) where \( \varphi \) is an isomorphism induced by the symmetric monoidal structure of \( \mathcal{L} \). Then we set \([\mu \alpha^c]_{\Gamma,\Delta} = \operatorname{cur}([\mu \alpha^c]_{\Gamma,\Delta} \varphi) \in \mathcal{L}(\Gamma \otimes [\Delta], [\sigma]^+) \).

Assume that \( \Gamma, x : \sigma : \tau \vdash t : \sigma \mid \Delta \) so that we have \([t]_{\Gamma,\Delta} \in \mathcal{L}(\Gamma \otimes [\Delta], ![\sigma]^+, ![\sigma]^+)\) where \( \varphi \) is an isomorphism induced by the symmetric monoidal structure of \( \mathcal{L} \). We set \([\operatorname{fix} x^\sigma \tau]_{\Gamma,\Delta} = \operatorname{fix}([\sigma]_!) \varphi \in \mathcal{L}(\Gamma \otimes [\Delta], [\sigma]^+) \).

Concerning commands, assume that \( \Gamma \vdash t : \sigma \mid \Delta \) and that \( \Gamma \mid \pi : \sigma \vdash \Delta \) so that we have \([t]_{\Gamma,\Delta} \in \mathcal{L}((\Gamma \otimes [\Delta], ![\sigma]^+) \) and \([\pi]_{\Gamma,\Delta} \in \mathcal{L}((\Gamma \otimes [\Delta], ![\sigma]) \) and therefore \([\pi, t]_{\Gamma,\Delta} \in \mathcal{L}(\Gamma \otimes [\Delta], [\sigma]) \).

We define \([t \cdot \pi]_{\Gamma,\Delta} \) as the following composition of morphisms in \( \mathcal{L} \).
This paper establishes the correspondence with Selinger control categories \cite{13} which are equivalent to \(L\) of the MIX command constructions introduced in Section 1. If \(\Gamma\) and \(\Delta\) are morphisms in \(\Sigma\), assume now that

\[
[\Gamma] \otimes [\sigma] \otimes [\Delta] \xrightarrow{w_{\Gamma} \otimes [\sigma] \otimes w_{\Delta}} 1 \otimes [\sigma] \otimes 1 \xrightarrow{\varphi} [\sigma]
\]

where \(\varphi\) is the obvious iso.

Remember that we have defined \(\text{succ}, \text{pred} \in L(N, N)\), so that we have \(\text{succ} \otimes \text{pred} \in L'(L(N, N), L(N, N))\). Assume that \(\Gamma \mid \pi : t \vdash \Delta\) so that \([\pi]\Gamma,\Delta \in L'(L'(L\otimes [\Delta], [\pi]), [t])\), and we set \([\text{succ} \cdot \pi]\Gamma,\Delta = \text{succ} \otimes [\pi]\Gamma,\Delta\) and \([\text{pred} \cdot \pi]\Gamma,\Delta = \text{pred} \otimes [\pi]\Gamma,\Delta\); both morphisms belong to \(L'(L'(L\otimes [\Delta], [\pi]), [t])\).

Remember also that, for any object \(X\) of \(L\), we have defined \(\overline{\pi} \in L(X \otimes X \otimes X, X)\). Using \(\ast\)-autonomy and isomorphisms induced by the monoidal structure of \(L\), we can canonically turn this morphism into \(\overline{\pi} \in L(X \otimes X \otimes X, N)\). Assume that \(X = P \otimes X\) where \(P\) is an object of \(L\). Then we can set \(\overline{\pi} \in L'(L'(L\otimes [\Delta], [\pi]), [t]))\). Assume that \(\Gamma \mid \pi : t \vdash \Delta\) and \(\Gamma \vdash t_i : \sigma \mid \Delta\) for \(i = 1, 2\). Then we have \([\pi]\Gamma,\Delta \otimes [t_1]^{\Gamma,\Delta} \otimes [t_2]^{\Gamma,\Delta} \in L'(L'(L\otimes [\Delta], [\sigma]), [\pi])\) and we define \([[\pi](t_1, t_2, \pi)]\Gamma,\Delta\) as the following composition of morphisms in \(L\), using a ternary version of the contraction morphism

Assume that \(\Gamma \vdash \pi : t \mid \Delta\) and that \(\Gamma \vdash t : \sigma \mid \Delta\) so that \([\pi]\Gamma,\Delta \in L'(L'(L\otimes [\Delta], [\pi]), [t])\) and \([t]\Gamma,\Delta \in L'(L'(L\otimes [\Delta], [\pi]), [\sigma])\), we set \([\varphi]t \cdot \pi = \iota_{\sigma} \otimes [\pi]\Gamma,\Delta \in L'(L'(L\otimes [\Delta], [\sigma]), [\pi])\).

Assume that \(\Gamma \mid \tau : \sigma \vdash \Delta\) so that \([\tau]\Gamma,\Delta \in L'(L'(L\otimes [\Delta], [\sigma]), [\sigma])\), we can set \([\varphi]t \cdot \pi = \iota_{\sigma} \otimes [\pi]\Gamma,\Delta \in L'(L'(L\otimes [\Delta], [\sigma]), [\sigma])\).

Assume that \(\Gamma \vdash \pi : t \mid \Delta\) and \(\Delta \vdash t : \sigma\), we set \([\varphi]t \cdot \pi = \iota_{\sigma} \otimes [\pi]\Gamma,\Delta \in L'(L'(L\otimes [\Delta], [\sigma]), [\sigma])\).

By simple calculations using the fact that stacks are interpreted as \(!\)-coalgebra morphisms.

2.2 A continuation category

We record briefly the connection between this LL-based approach and the Lafont-Reus-Streicher (LRS) \cite{14} approach of continuation categories, see \cite{10} for more details\(^2\). Let \(\mathcal{P} = L\), we have seen that \(\mathcal{P}\) is a cocartesian and cartesian category, with \(\oplus\) as coproduct and \(\otimes\) as product. As object of responses, we take \(\Sigma = 1\). Let \(P\) and \(Q\) be objects of \(\mathcal{P}\). Then we have \(\mathcal{P}(P \otimes Q, \Sigma) = L(P \otimes Q, 1) \simeq L(1, L(P, Q))\) because \(!\) is right adjoint to \(U\). Hence \(\mathcal{P}(P \otimes Q, \Sigma) \simeq L(P, L(Q))\) by the same adjunction. So setting \(\Sigma^Q = \iota_{L(Q)}\) we have \(\mathcal{P}(P \otimes Q, \Sigma) = \mathcal{P}(P, \Sigma^Q)\). Hence \(\Sigma\) is a baseable object of \(\mathcal{P}\).

\(^2\) This paper establishes the correspondence with Selinger control categories \cite{13} which are equivalent to continuation categories. They use therefore a negative translation whereas we use a positive one.
The category $\Sigma^P$ of negated objects has the same objects as $\mathcal{P}$, and $\Sigma^P(P,Q) = \mathcal{P}(\Sigma^P, \Sigma^Q)$. It is a cartesian closed category with product $P \times Q = P \oplus Q$ and object of morphisms $P \Rightarrow Q = \Sigma^P \otimes Q$ as easily checked, using the fact that $\Sigma$ is baseable. In the LRS setting, interpretation of types is done in $\mathcal{P}$, setting $[\sigma \Rightarrow \tau] = \Sigma^{[\sigma]} \otimes [\tau]$ and given contexts $\Gamma = (x_1 : \sigma_1, \ldots, x_n : \sigma_n)$ and $\Delta = (\alpha_1 : \tau_1, \ldots, \alpha_k : \tau_k)$, a term $t$ such that $\Gamma \vdash t : \sigma \mid \Delta$ is interpreted as $[t]_{\Gamma,\Delta} \in \mathcal{P}(\Sigma^{[\sigma]} \times \cdots \times \Sigma^{[\sigma_n]} \times [\tau_1] \times \cdots \times [\tau_k], \Sigma^{[\tau]})$, a command $c$ such that $\Gamma \vdash c \mid \Delta$ is interpreted as $[c]_{\Gamma,\Delta} \in \mathcal{P}(\Sigma^{[\sigma]} \times \cdots \times \Sigma^{[\sigma_n]} \times [\tau_1] \times \cdots \times [\tau_k], \Sigma)$ and a stack $\pi$ such that $\Gamma \vdash \pi : \tau \mid \Delta$ is interpreted as $[\pi]_{\Gamma,\Delta} \in \mathcal{P}(\Sigma^{[\sigma]} \times \cdots \times \Sigma^{[\sigma_n]} \times [\tau_1] \times \cdots \times [\tau_k], [\tau])$ and it is easily checked again that this interpretation is exactly the same as the one described above, up to the identification of $\mathcal{P}(\mathcal{P},!X)$ with $\mathcal{L}(\mathcal{P},X)$.

3 Relational semantics

In this most simple and canonical interpretation of LL, $\mathcal{L}$ is the category Rel whose objects are sets\(^3\) and where $\text{Rel}(X,Y) = \mathcal{P}(X \times Y)$, composition being defined as the usual composition of relations. We recall that the tensor unit is $1 = \{\star\}$ (arbitrary one-point set), that $X \otimes Y = X \times Y$ with tensor product of morphisms defined accordingly, that $X \rightarrow Y = X \times Y$ (and evaluation defined in the obvious way), that $\bot = 1$ so that $X^\bot = X$ up to canonical iso. This category is countably cartesian, with cartesian product $\bigotimes_{i \in I} X_i = \bigcup_{i \in I} (\{i\} \times X_i)$ (disjoint union) and projections defined in the obvious way ($\text{pr}^i = \{(i,a) \mid a \in X_i\}$). It is cocartesian with coproducts defined exactly as products and injections given by $\text{in}^i = \{(a, (i,a)) \mid a \in X_i\}$. It has an exponential functor defined on objects by $IX = M_{\text{fin}}(X)$ (set of all finite multisets of elements of $X$) and on morphisms by $I\mathcal{F} = \{[(b_1, \ldots, b_n) | (a_i, b_i) \in f \text{ for each } i]\}$. Dereliction (comultiplication) is given by $\text{der}_X = \{([a], a) \mid a \in X\}$ and digging (coequipment) is given by $\text{dig}_X = \{\{a_1 + \cdots + a_k, [m_1, \ldots, m_k] \mid m_1, \ldots, m_k \in !X\}$. The symmetric monoidality isomorphisms are given by $m^{(0)} = \{([*], \star)\}$ and

$$m^{(2)}_{X,Y} = \{([(a_1, \ldots, a_n), [b_1, \ldots, b_k]), ([1, a_1], \ldots, (1, a_n), (2, b_1), \ldots, (2, b_k)) \mid a_1, \ldots, a_n \in X \text{ and } b_1, \ldots, b_k \in Y\}$$

Let $P = (\mathcal{P}, h_P)$ be an object of $\text{Rel}^f$. Given $f \in \text{Rel}(\mathcal{P},X)$, the generalized promotion $f^f \in \text{Rel}(\mathcal{P},!X)$ is given by $f^f = \{(b, [a_1, \ldots, a_n]) \mid \exists b_1, \ldots, b_n \in P \{b, [b_1, \ldots, b_n]\} \in h_P \text{ and } (b_i, a_i) \in f \text{ for each } i\}$. The $n$-ary contraction $c^{(n)}_P \in \text{Rel}^f(P,P^\otimes n)$ is given by $c^{(n)}_P = \{(a, [a_1, \ldots, a_n]) \mid (a, [a_1, \ldots, a_n]) \in h_P\}$. In particular (0-ary case) we have $w_P = \{(a, *) \mid (a, \star) \in h_P\}$.

Let us describe the coalgebra structures of products and sums of coalgebras.

**Lemma 4.** Let $P_1$ and $P_2$ be objects of $\text{Rel}^f$. One has $(a, b, \{[a_1, b_1], \ldots, [a_n, b_n]\}) \in h_{P_1 \otimes P_2} \iff \{[a, [a_1, b_1], \ldots, [a_n, b_n]] \in h_{P_1} \text{ and } (b, [b_1, \ldots, b_n]) \in h_{P_2}\}$. And, given $l \in \{1,2\}$, one has $(l, a, \{[b_1, b_1], \ldots, [b_k, b_k]\}) \in h_{P_1 \oplus P_2} \iff \text{ for each } i = 1,\ldots,n, \text{ one has } b_i = (l, a_i) \text{, and moreover } (a, [a_1, \ldots, a_n]) \in h_{P_2}$.

The proof is a straightforward verification.

For each set $X$, we can define a fix-point operator as a least fix-point wrt morphism inclusion as follows: $\overline{\text{fix}}_X = \{m_1 + \cdots + m_k + \{(a_1, \ldots, a_k), a\} \mid \forall i \ (m_i, a_i) \in \overline{\text{fix}}_X\}$.

---

\(^3\) All sets can be assumed to be at most countable, this is a very reasonable assumption which is preserved by all the constructions that we introduce.
The object of natural numbers is the set \( \mathbb{N} \), the morphisms \( \text{succ} \) and \( \text{pred} \) are given obviously by \( \text{succ} = \{ (n, n + 1) \mid n \in \mathbb{N} \} \), \( \text{pred} = \{ (0, 0) \cup \{(n + 1, n) \mid n \in \mathbb{N} \} \). When \( X = P^+ \) where \( P \) is an object of \( \text{Rel}^1 \), the corresponding coalgebra morphism \( \Pi_X \in \text{Rel}^1 (P \otimes !P^\perp \otimes !P^\perp, \mathbb{N}^\perp) \) is given by \( \Pi_X = \{ (a_1, \ldots, a_k), [n_k, \ldots, n_l] \mid n_1 = \cdots = n_k = 0 \text{ and } n_{k+1}, \ldots, n_l \neq 0 \text{ and } (a_1, \ldots, a_l) \in h_P \} \).

This model of LL is also a model of MIX. It suffices to take \( \text{mix}^0 = \{ (*, *) \} \) and \( \text{mix}^2 = \{ ((*, *), *) \} \) and these morphisms define clearly a structure of commutative \( \otimes \)-monoid on \( \perp \).

### 3.1 Interpretation as a type deduction system

We introduce a typing system extending the one of [15] for representing the relational denotational semantics described above (the other difference of course is that, here, we are in a typed setting). A semantic variable context is a sequence \( \Phi = (x_1 : m_1 : \sigma_1, \ldots, x_n : m_n : \sigma_n) \) where \( m_i \in [\sigma_i]^+ \) for each \( i \) and variables are pairwise distinct. A semantic name context is a sequence \( \Psi = (\alpha_1 : \tau_1, \ldots, \alpha_k : \tau_k) \) where \( \alpha_i \in [\tau_i] \) for each \( i \) and the names are pairwise distinct. We also define the underlying typing contexts \( u(\Phi) = (x_1 : \sigma_1, \ldots, x_n : \sigma_n) \) and \( u(\Psi) = (\alpha_1 : \tau_1, \ldots, \alpha_k : \tau_k) \) as well as the underlying tuples \( \langle \Phi \rangle = (m_1, \ldots, m_n) \) and \( \langle \Psi \rangle = (a_1, \ldots, a_k) \). We extend multiset addition to tuples of multisets componentwise.

Observe that \( \langle \Phi \rangle \in [u(\Phi)] \) and similarly for \( \Psi \). Given a variable context \( \Gamma = (x_1 : \sigma_1, \ldots, x_n : \sigma_n) \) one defines the corresponding zero semantic context \( 0_\Gamma = (x : [] : \sigma_1, \ldots, x_n : [] : \sigma_n) \).

\[
0_\Gamma, x : [a] : \sigma \vdash x : a : \sigma \mid \Psi \\
0_\Gamma \mid \alpha : a : \sigma \vdash \alpha : a : \sigma, \Psi
\]
as soon as \( ([\Psi]_{0_u(\Psi)} \in h_{[u(\Psi)]} \).

\[
\Phi_1 \vdash t : a : \sigma \mid \Psi_1 \\
\Phi_2 \vdash \pi : a : \sigma + \Psi_2
\]
as soon as \( u(\Phi_1) = \Gamma, u(\Psi_1) = \Delta \) for \( i = 1, 2, \langle \Phi \rangle = \langle \Phi_1 \rangle + \langle \Phi_2 \rangle \) and \( ([\Psi]_{0_u(\Psi)} \in h_{[\Delta]} \).

\[
\Phi, x : m : \sigma + t : b : \tau \mid \Psi
\]

\[
\Phi | \lambda x^2 : t : (m, b) : \sigma \Rightarrow \tau \mid \Psi
\]
as soon as \( u(\Phi_1) = \Gamma, u(\Psi_1) = \Delta \) for each \( i = 0, \ldots, k, \langle \Phi \rangle = \langle \Phi_0 \rangle + \cdots + \langle \Phi_k \rangle \) and \( ([\Psi]_{0_u(\Psi)} \in h_{[\Delta]} \).

\[
\Phi | \alpha : a : \sigma \vdash \Psi
\]

\[
\Phi | \beta : b : \tau \vdash \Psi
\]
as soon as \( u(\Phi_1) = \Gamma, u(\Psi_1) = \Delta \) for each \( i = 0, \ldots, k, \langle \Phi \rangle = \langle \Phi_0 \rangle + \cdots + \langle \Phi_k \rangle \) and \( ([\Psi]_{0_u(\Psi)} \in h_{[\Delta]} \).

\[
\Phi | \mu \alpha : a : \sigma \mid \Psi
\]

\[
\Phi | \pi : a : \mu \alpha \mid \Psi
\]

\[
\Phi | \mu \alpha : a : \sigma \mid \Psi
\]
as soon as \( u(\Phi_1) = \Gamma, u(\Psi_1) = \Delta \) for each \( i = 0, \ldots, k, \langle \Phi \rangle = \langle \Phi_0 \rangle + \cdots + \langle \Phi_k \rangle \) and \( ([\Psi]_{0_u(\Psi)} \in h_{[\Delta]} \).

\[
\Phi | \beta : b : \tau \vdash \Psi
\]

\[
\Phi | \beta : b : \tau \vdash \Psi
\]
Theorem 6. Let \( \Phi = (x_1 : m_1 : \sigma_1, \ldots, x_n : m_n : \sigma_n) \) and \( \Psi = (a_1 : a_1 : \tau_1, \ldots, a_k : a_k : \tau_k) \) be semantic contexts. Assume that

\[ \Phi \vdash t : a : \sigma | \Psi \]

Proof. The first statement results directly from the definition, and the second one is proved by a simple induction on types. \( \square \)
12 Classical PCF

We check that it is closed. Given two weighted sets

\[ \pi_1 \in \|a_1\|^{\gamma^1}, \ldots, \pi_k \in \|a_k\|^{\gamma^k}, \] one has

\[ t' = t \left( t_1 / x_1, \ldots, t_n / x_n \right) \left( \pi_1 / a_1, \ldots, \pi_k / a_k \right) \in \|a\|^\sigma. \]

resp. \( \Phi \vdash \pi : \sigma \vdash \Psi \).

Then, for all \( t_1 \in \|m_1\|^\sigma, \ldots, t_n \in \|m_n\|^\sigma \) and all \( \pi_1 \in \|a_1\|^\gamma^1, \ldots, \pi_k \in \|a_k\|^\gamma^k \), one has

\[ t' = t \left( t_1 / x_1, \ldots, t_n / x_n \right) \left( \pi_1 / a_1, \ldots, \pi_k / a_k \right) \in \|a\|^\sigma. \]

resp. \( \pi' = \pi \left( t_1 / x_1, \ldots, t_n / x_n \right) \left( \pi_1 / a_1, \ldots, \pi_k / a_k \right) \in \|a\|^\sigma. \)

The proof is given in the appendix, Section 6.

In particular, if an almost closed command \( c \) has a non-empty interpretation, it normalizes for the \( \rightarrow \)-reduction to a uniquely defined normal almost closed command which can easily be retrieved from the semantics of \( c \). For instance, if \( \vdash c \mid a_1 : [0, 7] \mid t, a_2 : [3] \mid t, a_3 : [] \mid t \), then \( c \) normalizes to \((\{0 \cdot a_1\}|(\{3 \cdot a_2\}|(\{a_3 \cdot a_1\} (or a permutation of this MIX composition).

If \( c \) does not contain MIX constructs, we know that it will reduce to a normal command of shape \( \gamma \cdot a \), but the model “does not know” this property that we proved by purely syntactic means. We introduce now a light refinement of the relational model which takes this uniqueness of values property into account, and therefore rejects the MIX constructs.

## 5 A semantic account of uniqueness of values

This model originates from an observation made independently by several authors\(^4\) at an early stage of the development of LL that, in a multiplicative proof-net, there is a simple relation between the number of \( \otimes \) and of \( \forall \).

A weighted set is a pair \( X = (|X|, \gamma_X) \) where \( |X| \) is a set and \( \gamma_X : |X| \rightarrow \mathbb{Z} \) is a function. If we think of \( a \) as a proof tree in (constant-free) multiplicative LL (MLL) with only one conclusion (the root of the tree), then \( \gamma_X(a) = p - t \) where \( p \) is the number of \( \forall \) and \( t \) is the number of \( \otimes \) binary connectives occurring in \( a \). It is well known that if such a multiplicative proof tree can be sequentialized into a sequent calculus proof in MLL, then we have \( p - t = 1 \) (the converse is of course not true). This intuition explains the next definitions. One sets

\[ C(X) = \{ x \subseteq |X| \mid \forall a \in x \quad \gamma_X(a) = 1 \}. \]

Let \( \text{Rel}^W \) be the category of weighted sets and such that \( \text{Rel}^W(X, Y) = \{ t \subseteq |X| \times |Y| \mid \forall (a, b) \in t \quad \gamma_X(a) = \gamma_Y(b) \}. \) It is clear that \( !d_X = \{(a, a) \mid a \in |X| \} \in \text{Rel}^W(X, X) \) and that the relational composition of two morphisms is a morphism so that \( \text{Rel}^W \) is indeed a category.

One defines the weighted set \( 1 \) by \( |1| = \{ * \} \) (a singleton) and \( \gamma_1(*) = 1 \). Given two weighted sets \( X_1 \) and \( X_2 \), one defines \( X_1 \otimes X_2 \) by \( |X_1 \otimes X_2| = |X_1| \times |X_2| \) and \( \gamma_{X_1 \otimes X_2}(a_1, a_2) = \gamma_{X_1}(a_1) + \gamma_{X_2}(a_2) - 1 \). Given \( t_i \in \text{Rel}^W(X_i, Y_i) \) for \( i = 1, 2 \), one defines \( t_1 \otimes t_2 \) as in \( \text{Rel} \), then it is clear that \( t_1 \otimes t_2 \in \text{Rel}^W(X_1 \otimes X_2, Y_1 \otimes Y_2) \) and that this operation is a functor. Moreover, the usual bijections \( |1 \otimes X| \rightarrow |X|, |X \otimes 1| \rightarrow |X| \) and \( |X_1 \otimes X_2 \otimes X_3| \rightarrow |X_1 \otimes (X_2 \otimes X_3)| \) are isomorphisms in \( \text{Rel}^W \). Indeed we have

\[ \gamma_{t_1 \otimes t_2}(*, a) = 1 + \gamma_{X_1}(a) - 1 = \gamma_{X_1}(a) \]

\[ \gamma_{X_1 \otimes X_2 \otimes X_3}((a_1, a_2), a_3) = \gamma_{X_1}(a_1) + \gamma_{X_2}(a_2) + \gamma_{X_3}(a_3) = 2. \]

In that way, we have equipped \( \text{Rel}^W \) with a structure of symmetric monoidal category.

We check that it is closed. Given two weighted sets \( X \) and \( Y \), let \( X \rightarrow Y = (|X| \times |Y|, \gamma_X \rightarrow Y) \) and \( \gamma_X \rightarrow_Y (a, b) = \gamma_Y(b) - \gamma_X(a) + 1 \). Observe that \( C(X \rightarrow Y) = \text{Rel}^W(X, Y) \).

Then \( \text{ev} = \{(a, b) \mid a \in |X| \text{ and } b \in |Y| \} \) belongs to \( \text{Rel}^W([X \rightarrow Y] \otimes X, Y) \).

Indeed we have \( \gamma_{X \rightarrow Y \otimes X}(a, b) = \gamma_Y(b) - \gamma_X(a) + 1 + \gamma_X(a) - 1 = \gamma_Y(b) \).

\(^4\) At least: Girard, Danos and Regnier, Métayer, Rétoré and Fleury, Guerrini...
Let $Z$ be another weighted set and let $((c, a), b) \in [(Z \otimes X) \to Y]$. Then we have $\gamma_{(Z \otimes X) \to Y}((c, a), b) = \gamma_Y(b) - (\gamma_{Z}(c) + \gamma_X(a)) + 1 = \gamma_Y(b) - \gamma_Z(c) - \gamma_X(a) + 2$. On the other hand we have $\gamma_{Z \to (X \to Y)}((c, a), b) = (\gamma_Y(b) - \gamma_X(a) + 1 - \gamma_Z(c) + 1 = \gamma_Y(b) - \gamma_Z(c) - \gamma_X(a) + 2$ and therefore, given $t \in \text{RelW}(Z \otimes X, Y)$, we have $\text{cur}(t) = \{(c, (a, b)) \mid (c, a), b) \in t\} \in \text{RelW}(Z, X \to Y)$. This shows that $\text{RelW}$ is closed.

Let $\bot = \{(*), \gamma_\bot\}$ with $\gamma_\bot(*) = -1$. Then we have $\gamma_{X \to \bot}(a, *) = -1 - \gamma_X(a) + 1 = -\gamma_X(a)$. It follows that the canonical morphism $\eta_X \in \text{RelW}(X, (X \to \bot) \to \bot)$ given by $\eta_X = \text{cur}(\text{ev}_{\sigma, X, \to \bot})$ (where $\sigma$ is the symmetry natural isomorphism associated with the symmetric monoidal closed structure of $\text{RelW}$) is an isomorphism in $\text{RelW}$. This shows that, equipped with $\bot$ as dualizing object, the symmetric monoidal closed category $\text{RelW}$ is *-autonomous.

The co-tensor product, called $\text{par}$, is the operation defined by $X \bowtie Y = (X^\bot \otimes Y^\bot)^\bot$ and is characterized by $|X \bowtie Y| = |X| \times |Y|$ and $\gamma_{\bowtie Y}(a, b) = \gamma_Y(a) + \gamma_X(b) + 1$.

Let $X^\bot = \{|X|, -\gamma_X\}$. Then $X^\bot$ is naturally isomorphic to $X \to \bot$ and defines a strictly involutive functor $\text{RelW} \to \text{RelW}^{op}$. Its action on morphisms is contraposition: $i^\bot = \{(a, b) \mid (a, b) \in t\} \in \text{RelW}(Y^\bot, X^\bot)$ for any $t \in \text{RelW}(X, Y)$.

The category $\text{RelW}$ is cartesian and cocartesian. Given a family $(X_i)_{i \in I}$ of objects, let $X = \sqcup_{i \in I} X_i$ be defined by $|X| = \bigcup_{i \in I} \{X_i\}$ and $\gamma_X(i, a) = \gamma_{X_i}(a)$. Let $\pi_i = \{(i, a, a) \mid a \in |X_i|\} \in \text{RelW}(X, X_i)$. Then $(X, (\pi_i))$ is a cartesian product of the family $(X_i)_{i \in I}$.

The coproduct is defined in a completely similar way. Observe that the product of the empty family (the terminal object) is $\top = (\emptyset, \emptyset)$, which is also the initial object of $\text{RelW}$.

Let $X = (M_{fin}(\{X\}), \gamma_X)$ where

$$\gamma_X([a_1, \ldots, a_n]) = -n + 1 + \sum_{i=1}^{n} \gamma_X(a_i) = 1 + \sum_{i=1}^{n} (\gamma_X(a_i) - 1).$$

Given $t \in \text{RelW}(X, Y)$, it is clear that $!t \in \text{RelW}(!X, !Y)$ where $!t$ is defined as in $\text{Rel}$.

So $!\_\_$ is a functor $\text{RelW} \to \text{RelW}$. We equip this functor with a structure of comonad. For each object $X$, let $\text{der}_X = \{([a], a) \mid a \in |X|\}$. Since $\gamma_X([a]) = \gamma_X(a)$, we have $\text{der}_X \in \text{RelW}(!X, X)$. The naturality of $\text{der}_X$ is obvious (it already holds in $\text{Rel}$).

One defines also $\text{dig}_X = \{(m_1 + \cdots + m_k, [m_1, \ldots, m_k]) \mid k \in \mathbb{N} \text{ and } \forall i \in M_{fin}(|X|)\}$. Let $m_1, \ldots, m_k \in M_{fin}(|X|)$, and let us write $m_i = [a_{i1}, \ldots, a_{ik}]$. We have

$$\gamma_{!X}([m_1, \ldots, m_k]) = 1 + \sum_{i=1}^{k} (\gamma_X(m_i) - 1) = 1 + \sum_{i=1}^{k} (1 + \sum_{j=1}^{k} \gamma_X(a_{ij}) - 1) - 1$$

and therefore $\text{dig}_X \in \text{RelW}(!X, !X)$. One proves easily that $(!X, \text{der}_X, \text{dig}_X)$ defines a comonad (the definition of this structure is the same as in $\text{Rel}$).

To conclude that $\text{RelW}$ is a model of LL, we check that the standard Seely isomorphisms of $\text{Rel}$ are isomorphisms in $\text{RelW}$. The 0-ary isomorphism is $m^{(0)} = \{(*, [\_])\}$ and belongs to $\text{RelW}(1, !\top)$ since $\gamma_{\top}([\_]) = 1$. The binary version is $m^{(2)}_{X,Y} = \{([a_1, \ldots, a_n], [b_1, \ldots, b_p]) \mid \forall i \in |X| \text{ and } \forall j \in |Y|\}$. We prove that $m^{(2)}_{X,Y} \in \text{RelW}((X \otimes !Y!), (!X \& !Y))$: indeed, with the notations of this definition, we have

$$\gamma_{X \otimes !Y}([a_1, \ldots, a_n], [b_1, \ldots, b_p]) = 1 + \sum_{i=1}^{n} (\gamma_X(a_i) - 1) + \sum_{j=1}^{p} (\gamma_Y(b_j) - 1) - 1$$

and therefore

$$\gamma_{X \otimes !Y}([a_1, \ldots, a_n], [b_1, \ldots, b_p]) = 1 + \sum_{i=1}^{n} (\gamma_X(a_i) - 1)$$

and

$$\gamma_{X \otimes !Y}([a_1, \ldots, a_n], [b_1, \ldots, b_p]) = 1 + \sum_{j=1}^{p} (\gamma_Y(b_j) - 1) - 1 = \gamma_{!(X \& Y)}([(1, a_1), \ldots, (1, a_n), (2, b_1), \ldots, (2, b_p)])$$

where $\gamma_{X \otimes !Y}$ is the canonical morphism $\gamma_{X \otimes !Y} \in \text{RelW}((X \otimes !Y!), (!X \& !Y))$.
We have developed a semantic investigation of classical PCF, presented in Herbelin’s very pleasant $\lambda\mu$ format. We have recorded the general LL semantic framework for this calculus, based on Girard’s categorical semantics of LC, and its connection with Lafont-Reus-Streicher continuation categories. We have given a simple adequacy proof for the relational model and proposed a model which enforces uniqueness of values, rejecting the extension of classical PCF by a parallel composition construct based on the MIX rule of LL. In a longer version of this paper, we shall show that the Eilenberg-Moore category of the Scott semantics of LL admits a very simple description. All these results suggest that LL-based semantic investigations of

Conclusion

We have developed a semantic investigation of classical PCF, presented in Herbelin’s very pleasant $\lambda\mu$ format. We have recorded the general LL semantic framework for this calculus, based on Girard’s categorical semantics of LC, and its connection with Lafont-Reus-Streicher continuation categories. We have given a simple adequacy proof for the relational model and proposed a model which enforces uniqueness of values, rejecting the extension of classical PCF by a parallel composition construct based on the MIX rule of LL. In a longer version of this paper, we shall show that the Eilenberg-Moore category of the Scott semantics of LL admits a very simple description. All these results suggest that LL-based semantic investigations of
classical PCF are worthwhile even if we know that these models can be reduced to models of a subsystem of the typed lambda-calculus through a kind of CPS translation.

References


6 Appendix: proof of the Adequacy Theorem

Proof. By induction on the semantic typing derivations for t, c or π. If e is an expression (term, command or stack), we use e′ for the expression e [t_1/x_1, ..., t_n/x_n] [π_1/α_1, ..., π_k/α_k].
If \( t = x_i \), we must have \( \sigma_i = \sigma \), \( m_i = [a] \) and since \( t' = t_i \), we conclude straightforwardly that \( t' \in [a]^\sigma \).

Assume that \( t = \lambda x^\tau s \), \( \sigma = (\tau \Rightarrow \varphi) \) and \( a = (m, b) \) and the premise of the last rule is \( \Phi, x : m, \tau \vdash s : b : \Psi \). We must prove that \( t' \ast \pi \in \mathcal{N} \) for all \( \pi \in \langle \([m, b]\rangle \rangle^\pi \ast \tau \). But such a \( \pi \) is of shape \( \pi = \text{arg}(a) \ast \rho \) with \( u \in \langle m \rangle^\sigma \) and \( \rho \in \langle b \rangle^\tau \). So we have \( t' \ast \pi \rightarrow \varphi'[u/x] \ast \rho \). By inductive hypothesis, we have \( \varphi'[u/x] \in \langle b \rangle^\tau \) and hence \( \varphi'[u/x] \ast \rho \in \mathcal{N} \) from which it follows that \( t' \ast \pi \in \mathcal{N} \) as required.

Assume that \( \sigma = \sigma_1 \times \sigma_2, t = (s_1, s_2) \) and \( a = (1, b_1) \) with \( \Phi \vdash s_1 : \sigma_1 : \Psi \). We must prove that \( t' = (s'_1, s'_2) \in \langle (1, b_1) \rangle^\sigma_1 \times \sigma_2 \). So let \( \pi \in \langle (1, b_1) \rangle^\sigma_1 \times \sigma_2 \), this means that \( \pi = \text{pr}_i \ast \rho \) for some \( \rho \in \langle b_1 \rangle^\sigma_1 \). We have \( t' \ast \pi \rightarrow s'_1 \ast \rho \in \mathcal{N} \) by inductive hypothesis. The case where \( a = (2, b_2) \) is similar.

Assume that \( t = \mu \alpha^\sigma c \) so that the premise of the last rule is \( \Phi \vdash c \vdash \alpha : \sigma : \Psi \). We must show that \( t' \ast \pi \in \mathcal{N} \) for all \( \pi \in \langle [a] \rangle^\sigma \). But \( t' \ast \pi \rightarrow c'[\pi / \alpha] \in \mathcal{N} \) by inductive hypothesis and the expected conclusion follows.

Assume that \( t = \text{fix} x^\tau s \) so that there are \( a_1, \ldots, a_l \) such that the premises of the last rule are \( \Phi_0, x : [a_1, \ldots, a_l] : \sigma \vdash s : \sigma \vdash \Psi_0 \) and \( \Phi_i \vdash t : a_i : \sigma : \Psi_i \) for \( i = 1, \ldots, l \). Moreover, we have \( \langle \Phi \rangle = \sum_{i=0}^l \langle \Phi_i \rangle \) and \( \langle \langle \Psi \rangle_i \rangle \rangle \in \mathcal{N}^{\langle \Psi \rangle} \). Let \( \pi \in \langle [a] \rangle^\sigma \), we must prove that \( t' \ast \pi \in \mathcal{N} \). Let us write \( \Phi_i = (x_1 : m_1 : \sigma_1, \ldots, x_n : m_n : \sigma_n) \) and \( \Psi_i : (a_1 : a_1' : \tau_1, \ldots, a_k : a_k' : \tau_k) \) so that, with the notations of the statement of the Theorem, we have \( m_i = \sum_{j=1}^n m_j[i] \) for \( i = 1, \ldots, n \) and we have \( \langle a_i, a_i' \rangle \in \langle \Psi \rangle_{\tau_i} \) for \( i = 1, \ldots, k \). By Lemma 5, we can therefore apply the inductive hypothesis and we get \( s'[t'/x] \in \langle [a] \rangle^\sigma \). Let \( \pi \in \langle [a] \rangle^\sigma \) we have \( s'[t'/x] \ast \pi \in \mathcal{N} \) and it follows that \( t' \ast \pi \in \mathcal{N} \) since \( t' \ast \pi \rightarrow s'[t'/x] \ast \pi \).

Assume that \( c = c \ast \pi \) so that the premises of the last rule are \( \Phi_1 \vdash c_1 : \Psi_1 \) and \( \Phi_2 \vdash t : a : \sigma : \Psi_2 \) with \( \langle \Phi \rangle = \langle \Phi_1 \rangle + \langle \Phi_2 \rangle \) and \( \langle \langle \Psi \rangle, \langle \Psi_1 \rangle, \langle \Psi_2 \rangle \rangle \in \mathcal{N}^{\langle \Psi \rangle} \). As before, using Lemma 5 we can apply the inductive hypothesis which yields \( t' \in [a] \rangle^\sigma \) and \( \pi' \in \langle [a] \rangle^\sigma \). Therefore \( \pi' \ast \pi \in \mathcal{N} \) as required.

Assume that \( c = 1 \ast \pi \) so that the premises of the last rule are \( \Phi_1 \vdash c_1 : \Psi_1 \) for \( i = 1, 2 \) with \( \langle \Phi \rangle = \langle \Phi_1 \rangle + \langle \Phi_2 \rangle \) and \( \langle \langle \Psi \rangle, \langle \Psi_1 \rangle, \langle \Psi_2 \rangle \rangle \in \mathcal{N}^{\langle \Psi \rangle} \). As before, using Lemma 5 we can apply the inductive hypothesis which yields \( c_i' \ast \pi \in \mathcal{N} \) for \( i = 1, 2 \) and hence \( c_{\pi} = c_1 [c_2] \in \mathcal{N} \).

If \( c = \text{err} \), there is nothing to prove since \( c_{\pi} = c \in \mathcal{N} \).

Assume that \( \pi = \text{arg}(t) \ast \rho \) with premises of the last rule \( \Phi_0 \vdash b : \varphi : \Psi_0 \) and \( \Phi_i \vdash t : a_i : \varphi : \Psi_i 
\) for \( i = 1, \ldots, l \) with \( \Phi = \sum_{i=0}^l \langle \Phi_i \rangle \) and \( \langle \langle \Psi \rangle_i \rangle \rangle \in \mathcal{N}^{\langle \Psi \rangle} \). As before, using Lemma 5 we can apply the inductive hypothesis which yields \( \rho' \in \langle [b] \rangle^\tau \) and \( t' \in \langle [a_1, \ldots, a_l] \rangle^\tau \). It follows that \( \pi' = \text{arg}(t') \ast \rho' \in \langle [a_1, \ldots, a_l, b] \rangle^\tau \) as expected.

Assume that \( \pi = \text{pr}_1 \ast \rho \) with premise of the last rule \( \Phi \vdash a : \sigma : \Psi \) and conclusion \( \Phi \vdash \lambda (1, a) : \sigma \ast \tau : \Psi \). By inductive hypothesis we have \( \rho' \in \langle [a] \rangle^\tau \) and hence \( \pi \in \langle [1, a] \rangle^\tau \ast \pi \) by definition of that set. The case \( \pi = \text{pr}_2 \ast \rho \) is of course similar.

Assume that \( \pi = \text{succ} \ast \rho \) with premises of the last rule \( \Phi \vdash \rho : [p_1 + 1, \ldots, p_i + 1] : \iota : \Psi \). By inductive hypothesis we have \( \rho' \in \langle [p_1 + 1, \ldots, p_i + 1] \rangle^\iota \), that is \( p_i + 1 \ast \rho' \in \mathcal{N} \) for \( i = 1, \ldots, l \). It follows that \( p_i \ast \text{succ} \ast \rho' \in \mathcal{N} \) for \( i = 1, \ldots, l \) as expected. The case where \( \pi = \text{pred} \ast \rho \) is similar.

Assume last that \( \pi = (t_1, t_2) \ast \rho \) and that the premises of the last rule are \( \Phi_0 \vdash b : a : \sigma : \Psi_0, \Phi_i \vdash t_1 : a_i : \sigma : \Psi_i \) for \( i = 1, \ldots, l \) and \( \Phi_i \vdash t_2 : a_i : \sigma : \Psi_i \) for \( i = l + 1, \ldots, r \), with \( \langle a_i, a_i' \rangle \in \mathcal{N} \). The conclusion of that rule is \( \Phi \vdash \rho : [p_1, \ldots, p_r] : t : \Psi \) where \( p_1, \ldots, p_r \) are natural numbers such that \( p_1 = \cdots = p_1 = 0 \) and \( p_{i+1} + \ldots + p_r \neq 0 \). \( \Phi = \sum_{i=0}^r \langle \Phi_i \rangle \) and \( \langle \langle \Psi \rangle_i \rangle \rangle \in \mathcal{N}^{\langle \Psi \rangle} \). As before, using Lemma 5 we can apply the inductive hypothesis which yields \( \forall i \in \{1, \ldots, l\} \) \( t_i' \in [a_i]^\pi \), \( \forall i \in \{l + 1, \ldots, r\} \) \( t_i' \in [a_i]^\pi \).
and \( \rho' \in \|a\|'' \). We must prove that if \((t'_1, t'_2) \cdot \rho' \in \|[p_1, \ldots, p_r]\|'' \). This results from the fact that \( p_i \ast \text{if}\,(t'_1, t'_2) \cdot \rho' \rightarrow t'_1 \ast \rho' \) for \( i \leq l \) and \( p_i \ast \text{if}\,(t'_1, t'_2) \cdot \rho' \rightarrow t'_2 \ast \rho' \) for \( i > l \), and from the fact that \( \rho' \in \|a_i\|'' \) for each \( i \), by inductive hypothesis and by Lemma 5. \( \square \)