Categorical Semantics for Mixed Linear/Non-linear Recursive Types

Bert Lindenhovius\textsuperscript{1}, Michael Mislove\textsuperscript{1}, and Vladimir Zamzhiev\textsuperscript{2}

\textsuperscript{1}Tulane University
\textsuperscript{2}Université de Lorraine, CNRS, Inria, LORIA, F 54000 Nancy, France

We describe a type system with mixed linear/non-linear recursive types called LNL-FPC (the linear/non-linear fixpoint calculus). LNL-FPC supports type-level recursion which in turn induces term-level recursion. We also provide sound and computationally adequate categorical models for LNL-FPC which describe the categorical structure of the substructural operations of Intuitionistic Linear Logic at all non-linear types, including the recursive ones. In order to do so, we describe a new technique for solving recursive domain equations within the category CPO by constructing the solutions over pre-embeddings. The type system enjoys implicit weakening and contraction rules which we are able to model by identifying the canonical comonoid structure of all non-linear types. We also show that the requirements of our abstract model are reasonable by constructing a large class of concrete models that have found applications not only in classical functional programming, but also in emerging programming paradigms that incorporate linear types, such as quantum programming and circuit description programming languages.

Paper Information. This is an extended abstract for [LMZ19].

Introduction. The Fixpoint Calculus (FPC) is a type system that has been extensively studied as a foundation for functional programming languages with recursive types [Fio94, FP94]. Girard’s introduction of linear logic [Gir87] initiated a parallel line of research into the logics underpinning functional pro-
gramming languages [Abr93]. Linear types also appear in concurrent set-
tings [CPT16], in quantum programming [RS17] and in circuit description pro-
gramming languages [LMZ18], among others.

In this paper we present a foundational treatment of mixed linear/non-
linear recursive types. "Mixed" should be understood in the sense the type
system has both linear and non-linear types, terms and contexts. The primary
focus of the paper is to demonstrate that recursive types behave very well syn-
tactically, operationally and categorically in such a mixed linear/non-linear
setting.

Overview and summary of results. To illustrate our results, we present a type system
called \textit{LNL-FPC} (the linear/non-linear fixpoint calculus). The syntax of our
language is an extension with recursive types of the circuit-free fragment of
Proto-Quipper-M [RS17] which is referred to as the CLNL calculus in [LMZ18].
Notably, the type system has \textit{implicit} weakening and contraction rules – non-
linear variables are automatically copied and discarded whenever necessary
by the language and not by the users. We also equip LNL-FPC with a call-
by-value big-step operational semantics and show that type-level recursion
induces term-level recursion, thus recreating a well-known result from FPC.

The main difficulty in designing LNL-FPC is on the categorical side. Our
categorical model is given by a \textit{CPO}-enriched symmetric monoidal adjunction
\[
\begin{array}{ccc}
\text{CPO} & \xrightarrow{\perp} & \text{L} \\
\downarrow F & & \downarrow G \\
\end{array}
\]
which has suitable \(\omega\)-colimits and order properties. We
show the category \textit{L} is \textit{CPO}-algebraically compact in the sense that every
\textit{CPO}-endofunctor has an initial algebra whose inverse is a final coalgebra.

However, since we work in a mixed linear/non-linear setting, we also have
to explain how to construct (parameterised) initial algebras of functors de-
dpending on mixed-variance functors within the category \textit{CPO} which is a more
challenging problem. We do so by \textit{reflecting} the solutions from \textit{L}_e (the sub-
category of embeddings) via the left adjoint \(F\) into the subcategory of \textit{pre-}
embeddings \textit{CPO}_{pe}. We show the subcategory of pre-embeddings \textit{CPO}_{pe} has
sufficient structure for constructing (parameterised) initial algebras, which
moreover satisfy important coherence properties with respect to the (parame-
terised) initial algebras constructed in \textit{L}_e.

Then, the (standard) interpretation of an (arbitrary) type \(\Theta \vdash A\) is a covari-
ant functor \([\Theta \vdash A] : \text{L}_e(\Theta) \to \text{L}_e\). A non-linear type \(\Theta \vdash P\) admits an \textit{additional}
non-linear interpretation as a covariant functor $(\Theta \vdash P) : \text{CPO}_{pe}^{\Theta} \to \text{CPO}_{pe}$ which is strongly related to its standard interpretation via a natural isomorphism $\alpha_{\Theta \vdash P} : (\Theta \vdash P) \circ F_{pe}^{\Theta} \Rightarrow F_{pe} \circ (\Theta \vdash P) : \text{CPO}_{pe}^{\Theta} \to L_e$, where $F_{pe}$ is the restriction of $F$ to $\text{CPO}_{pe}$. By exploiting this natural isomorphism and the strong coherence properties that it enjoys, we provide a coherent interpretation of the substructural operations of ILL at all non-linear types, including the recursive ones. This then allows us to characterise the canonical comonoid structure of non-linear (recursive) types and we prove our semantics sound.

We show the requirements of our abstract model are reasonable by constructing a large class of concrete models which have been used in different programming paradigms ranging from classical to quantum and we cement our results by presenting a computational adequacy result.

References


