Fully Complete Minimal PER Models for the Simply Typed λ-calculus*

Samson Abramsky[1] and Marina Lenisa[2]

[1] Oxford University Computing Laboratory
Wolfson Building, Parks Road, OX1 3QD, England.
e-mail: `Samson.Abramsky@comlab.ox.ac.uk`
[2] Dipartimento di Matematica e Informatica, Università di Udine,
Via delle Scienze 206, 33100 Udine, ITALY.
e-mail: `lenisa@dimi.uniud.it`.

Abstract. We show how to build a *fully complete* model for the *maximal theory* of the simply typed λ-calculus with k ground constants, λ_k. This is obtained by *linear realizability* over an *affine combinatory algebra* of *partial involutions* from natural numbers into natural numbers. For simplicity, we give the details of the construction of a fully complete model for λ_k extended with ground *permutations*. The fully complete minimal model for λ_k can be obtained by carrying out the previous construction over a suitable subalgebra of partial involutions. The full completeness result is then put to use in order to prove some simple results on the maximal theory.

Introduction

A *categorical* model of a type theory (or logic) is said to be *fully-complete* ([AJ94a]) if, for all *types* (*formulae*) A, B, all morphisms $f : [\![A]\!] \to [\![B]\!]$, from the interpretation of A into the interpretation of B, are denotations of a *proof-term* of the entailment $A \vdash B$, i.e. if the interpretation function from the category of syntactical objects to the category of denotations is *full*. The notion of full-completeness is a counterpart to the notion of *full abstraction* for programming languages. A fully complete model indicates that there is a very tight connection between syntax and semantics. Equivalently, one can say that the term model has been made into a mathematically respectable structure.

Over the past decade, Game Semantics has been used successfully by various people to define fully-complete models for various fragments of Linear Logic, and to give fully-abstract models for many programming languages, including PCF, and other functional and non-functional languages. Recently, a new technique, called *linear realizability* (see [AL99,AL00]), has been proposed as a valid and less complex alternative to Game Semantics in providing fully complete and fully abstract models. In particular, this technique has been used in [AL99,AL00] to define a model fully complete w.r.t. the fragment of system F consisting of

* Work partially supported by TMR Linear FMRX-CT98-0170.

ML polymorphic types, and in [AL99a] to provide a fully complete model for PCF. The linear (linear affine) realizability technique amounts to constructing a category of *Partial Equivalence Relations (PERs)* over a *Linear Combinatory Algebra, LCA*, (Affine Combinatory Algebra, ACA). This category turns out to be *linear* (affine), and to form an *adjoint model* with its co-Kleisli category. The notion of Linear (Affine) Combinatory Algebra introduced by the first author ([Abr97]) refines the standard notion of Combinatory Algebra, in the same way in which intuitionistic linear (affine) logic refines intuitionistic logic. The construction of PER models from LCA's (ACA's) of [AL99,AL00] is quite simple and clear, and it yields models with *extensionality* properties. Many examples of linear combinatory algebras arise in the context of Abramsky's categorical version of Girard's *Geometry of Interaction* ([AJ94,Abr97,Abr96,AHS98]).

In this paper, we define a *fully complete PER model* for the *maximal theory* $\approx$ on λ_k. λ_k is the simply typed λ-calculus with finitely many ground constants in the ground type o. The theory $\approx$ equates two closed λ-terms M, N of type $T_1 \rightarrow \ldots \rightarrow T_n \rightarrow o$ if and only if, for all $P_1 \approx Q_1$ of type T_1, ..., $P_n \approx Q_n$ of type T_n, $MP_1 \ldots P_n =_\beta NQ_1 \ldots Q_n$. To our knowledge, our model is the first model of $\approx$ different from the term model.

For simplicity, we show first how to build a fully complete minimal PER model $\mathcal{M}_{\mathcal{O}_k}$ for the simply typed λ-calculus with k ground constants extended with *permutations* of ground type. The fully complete minimal model for λ_k can then be obtained by cutting down the combinatory algebra. The model $\mathcal{M}_{\mathcal{O}_k}$ for the extended language arises from the special affine combinatory algebra of *partial involutions* used in [AL99,AL00] for modeling System F. It consists, essentially, of the hierarchy of simple PERs over a PER having exactly k distinct equivalence classes (for any $k \geq 2$). The proof of full completeness carried out in this paper is based on the *linear affine* analysis of the *intuitionistic* arrow, which is possible in our PER category. Our proof uses a *Decomposition Theorem*, which is by now a standard tool in discussing full completeness. In the present case, given a partial involution which inhabits a PER interpreting a simple type, the Decomposition Theorem allows to recover the *top-level structure* (up-to permutations) of the (possibly infinite) *typed Böhm tree* corresponding to the given partial involution. Once we have the Decomposition Theorem, in order to prove λ-definability, we still need to rule out possibly infinite typed Böhm trees from the model. In order to do this, we prove an *Approximation Theorem*, and we study an intermediate PER model $\mathcal{M}_{\mathcal{O}_k^\perp}$ for λ_k extended with a new ground constant $\perp$, intended to denote the undefined constant. A variant of this model, $\mathcal{M}_{\mathcal{O}}$, for the special case of two gound constants, $\top$ and $\perp$, has been used implicitly as an intermediate construction in the work of [AL99] on system F.

In order to get a fully complete minimal model for λ_k, we only need to cut down the algebra of involutions by putting an extra constraint, which allows us to rule out permutations from the model $\mathcal{M}_{\mathcal{O}_k}$.

The full completeness result is then put to use to prove (or re-prove) some simple facts on the maximal λ-theory. In particular, the *Context Lemma* follows immediately from the full completeness of the model. Moreover, we are able to prove that this holds also for the simply typed (possibly infinite) Böhm trees, in the special case of exactly two ground constants. We give also some decidability results, such as whether a *finite* partial involution belongs to a given type. These give a *semantical* procedure to decide, in some special cases, whether two terms are equivalent in the maximal theory, alternative to the *syntactical* procedures by Padovani [Pad95] and Loader [Loa97].

The paper is organized as follows. In Section 1, we recall some definitions and notations concerning the simply typed λ-calculus with finitely many base constants, the construction of a PER category over an ACA, and we present the special ACA of partial involutions. In Section 2 we study the PER model $\mathcal{M}_{\mathcal{O}_k}$ for λ_k and we prove that it is fully complete and minimal w.r.t. the extended calculus. In Section 3, we present the construction of the fully complete and minimal model for the simply typed λ-calculus. In Section 4, we illustrate some uses of the full completeness result. Conclusions and directions for future work appear in Section 5.

Notation. Vectors are written in bold. For $f, f' : \mathrm{Nat} \rightharpoonup \mathrm{Nat}$ partial functions from Nat to Nat, and $n, n' \in \mathrm{Nat}$, we denote, respectively, by $f(n)\downarrow$ and by $f(n)\uparrow$ the fact that f is defined on n and the fact that f is not defined (diverges) on n. We denote by $f(n) \updownarrow f'(n')$ the equiconvergence predicate to be read as: $f(n)\downarrow \Leftrightarrow\ f'(n')\downarrow$. Let X, Y be sets, then (l, x) and (r, y), where $x \in X$ and $y \in Y$, denote elements of $X + Y$. Let $X_1, \ldots, X_n$ be sets, often we will omit the parentheses in denoting $((X_1 + \ldots) + X_{n-1}) + X_n$. We will use the abbreviated notation (i, x), for denoting the element of $X_1 + \ldots + X_n$ coming from an element $x \in X_i$.

1 Preliminaries

This section consists of two parts. In the first part, we recall some definitions and notations concerning the simply typed λ-calculus λ_k with k base constants, the maximal theory, and its models. In the second part, we recall the notion of *affine combinatory algebra* (ACA) [Abr97], the construction of a PER category over an ACA [AL99,AL00], and we present the special ACA of *partial involutions*.

1.1 Simply Typed λ-calculus, Maximal Theories, Models

Definition 1 **(λ_k).** *The class* SimType *of simple types over a ground type o is defined by:*

$$(\mathit{SimType} \ni)\ T \ ::=\ o \ \mid\ T \to T\ .$$

Raw Terms are defined as follows:

$$\Lambda \ni\ M\ ::= c_i\ \mid\ x\ \mid\ \lambda x : T.M\ \mid\ MM\ ,$$

where $c_i \in Const_k = \{c_1, \ldots, c_k\}$, is the set of ground constants, $x \in Var$. We denote by Λ^0 the set of closed λ-terms.

Well-typed terms. *We introduce a* proof system *for deriving* typing judgements *of the form* $\Delta \vdash M : T$*, where* Δ *is a* type assignment*, i.e. a finite list* $x_1 : T_1, \ldots, x_n : T_n$*. The rules of the proof system are the following:*

$$\frac{}{\Delta \vdash c_i : o} \qquad\qquad \frac{}{\Delta, x : T, \Delta' \vdash x : T}$$

$$\frac{\Delta, x : T \vdash M : S}{\Delta \vdash \lambda x : T.M : T \to S} \qquad\qquad \frac{\Delta \vdash M : T \to S \quad \Delta \vdash N : T}{\Delta \vdash MN : S}$$

β-conversion. *β-conversion between well-typed terms is the least relation generated by the following rule and the rules for congruence closure (which we omit):* $\Delta \vdash (\lambda x : T.M)N = M[N/x] \ : \ S$*, where* $\Delta, x : T \vdash M : S$*, and* $\Delta \vdash N : T$*.*

It is well-known that the *maximal theory* over λ_k can be characterized as follows:

Definition 2 (Maximal Theory on λ_k). *Let* $M, N \in \Lambda$*. We define the equivalence* $\approx \subseteq \Lambda^0 \times \Lambda^0$ *by induction on types as follows:*
$M \approx_{T_1 \to \ldots T_n \to o} N$ *iff* $\vdash M : T_1 \to \ldots T_n \to o$, $\vdash N : T_1 \to \ldots T_n \to o$, *and* $\forall P_1 \approx_{T_1} Q_1 \ldots P_n \approx_{T_n} Q_n.\ MP_1 \ldots P_n =_\beta NQ_1 \ldots Q_n$.

In this paper, we will focus on categorical models for the simply typed λ-calculus λ_k. As usual, categorical models of λ_k are cartesian closed categorie, in which types are interpreted by objects and terms in contexts are interpreted by morphisms.

Definition 3 (Fully Complete Model). *A categorical model of* λ_k *is* fully complete *if, for all simple types* T *and for all* $h : 1 \to [\![T]\!]$*, there exists* $M \in \Lambda^0$ *such that* $h = [\![\vdash M : T]\!]$*, where* $[\![\ ]\!]$ *is the interpretation function in the model.*

1.2 PERs over Affine Combinatory Algebras

In this section, we briefly recall the construction of a PER category from an affine combinatory algebra (ACA) (see [AL99,AL00] for more details on this construction). This category turns out to be *affine*, while its co-Kleisli category turns out to be cartesian closed. In particular, we focus on a combinatory algebra of partial involutions $\mathcal{A}_{\mathsf{PInv}}$, and we consider the model of λ_k induced by the cartesian closed subcategory of PERs over $\mathcal{A}_{\mathsf{PInv}}$ generated by the special PER $\mathcal{O}_k$. This latter PER is intended to denote the ground type o.

We start by giving the notion of *affine combinatory algebra* (ACA):

Definition 4 (Affine Combinatory Algebra, [Abr97]). *An* affine combinatory algebra $\mathcal{A} = (A, \bullet, !)$ *is an applicative structure* $(A, \bullet)$ *with a unary (injective) operation* $!$*, and distinguished elements (combinators)* $\boldsymbol{B}, \boldsymbol{C}, \boldsymbol{I}, \boldsymbol{K}, \boldsymbol{W}, \boldsymbol{D}, \delta, \boldsymbol{F}$ *satisfying the following equations:*

$$\boldsymbol{I}x = x \qquad \boldsymbol{B}xyz = x(yz) \qquad \boldsymbol{C}xyz = (xz)y \qquad \boldsymbol{K}xy = x$$
$$\boldsymbol{W}x!y = x!y!y \qquad \boldsymbol{D}!x = x \qquad \delta!x = !!x \qquad \boldsymbol{F}!x!y = !(xy).$$

Proposition 1 ([AL99,AL00]). *Let $\mathcal{A} = (A, \bullet)$ be an ACA. We define the category $PER_{\mathcal{A}}$ as follows.*
Objects*: PERs $\mathcal{R} \subseteq A \times A$, i.e. symmetric and transitive relations.*
Morphisms*: a morphism f from $\mathcal{R}$ to $\mathcal{S}$ is an equivalence class of the PER $\mathcal{R} \multimap \mathcal{S}$, where $\mathcal{R} \multimap \mathcal{S}$ is defined by: $\alpha(\mathcal{R} \multimap \mathcal{S})\beta$ iff $\forall \gamma \,\mathcal{R}\, \gamma'.\ \alpha \bullet \gamma \,\mathcal{S}\, \beta \bullet \gamma'$.*
Let P be the pairing combinator, *i.e. (using λ-notation) $P = \lambda xyz.zxy$. Then, for all PERs $\mathcal{R}, \mathcal{S}$, let $\mathcal{R} \otimes \mathcal{S}$ be the PER defined by:*

$$\mathcal{R} \otimes \mathcal{S} = \{(P\alpha\beta, P\alpha'\beta') \mid \alpha \,\mathcal{R}\, \alpha' \ \wedge \ \beta \,\mathcal{S}\, \beta'\} \ .$$

$\otimes$ gives rise to a tensor product on $PER_{\mathcal{A}}$. We define the PER $\mathcal{I} = \{(\boldsymbol{I}, \boldsymbol{I})\}$ to be the tensor identity.
For all $\mathcal{R}$, we define $!\,\mathcal{R} = \{(!\alpha, !\beta) \mid \alpha \ \mathcal{R}\ \beta\}$. $!$ gives rise to a symmetric monoidal comonad $(!, der, \delta)$ on $PER_{\mathcal{A}}$, where $der = [\boldsymbol{D}]$, and δ denotes, by abuse of notation, the equivalence class of the combinator δ.
Summarizing, we have:
- *The category $PER_{\mathcal{A}}$ is affine.*
- *The co-Kleisli category $(PER_{\mathcal{A}})_!$, induced by the comonad $!$ on the category $PER_{\mathcal{A}}$, is cartesian closed.*
- *The categories $PER_{\mathcal{A}}$ and $(PER_{\mathcal{A}})_!$ form an adjoint model.*

The first author introduced in [Abr97] a basic important example of an ACA on the space [Nat $\rightharpoonup$ Nat] of partial functions from natural numbers into natural numbers. Here we briefly recall the definition of this ACA (see [AL99,AL00] for more details). The ACA of partial involutions, which we will consider in the next section, arises as subalgebra of this.

Let us consider the space [Nat $\rightharpoonup$ Nat] of partial functions from natural numbers to natural numbers. For any $\alpha \in$ [Nat $\rightharpoonup$ Nat] injective, we denote by α^{-1} the inverse of α. We start by fixing two injective *coding* functions t and p:

$$t : \mathrm{Nat} + \mathrm{Nat} \rightharpoonup \mathrm{Nat} \quad , \quad p : \mathrm{Nat} \times \mathrm{Nat} \rightharpoonup \mathrm{Nat} \ .$$

The first is used in order to define application, and it allows to transform a one-input/one-output function into a two-input/two-output function. The latter is used for creating *infinitely* many copies of a one-input/one-output function α, i.e. for defining $!\alpha$. Application can be explained *geometrically*, using the language of "boxes and wires" which arises in the general setting of traced symmetric monoidal categories (see [JSV96] for an abstract treatment). Let us represent a one-input/one-output function $\alpha \in$ [Nat $\rightharpoonup$ Nat] by the one-input-port/one-output-port box in Fig. 1(i) below. In order to define the application $\alpha \bullet \beta$, for $\alpha, \beta \in$ [Nat $\rightharpoonup$ Nat], we regard α as a two-input/two-output function via the coding t. In particular, $t; \alpha; t^{-1}$: Nat + Nat $\rightharpoonup$ Nat + Nat can be described as a matrix of 4 one-input/one-output functions, where each entry α_{ij} : Nat $\rightharpoonup$ Nat, $\alpha_{ij} = in_i; t; \alpha; t^{-1}; in_j^{-1}$ accounts for the contribution from the i-th input wire into the j-th output wire (see Fig. 1(ii)).

The result of the application $\alpha \bullet \beta$ is the following one-input/one-output function (see Fig. 1(iii)): $\alpha \bullet \beta = \alpha_{22} \cup \alpha_{21}; (\beta; \alpha_{11})^\star; \beta; \alpha_{12}$,
where $\cup$ denotes union of graph relations, and $(\beta; \alpha_{11})^\star$ denotes $\bigcup_{n \geq 0} (\beta; \alpha_{11})^n$.

The above formula for computing the application is essentially the *Execution Formula* from Girard's Geometry of Interaction ([Gir89]).

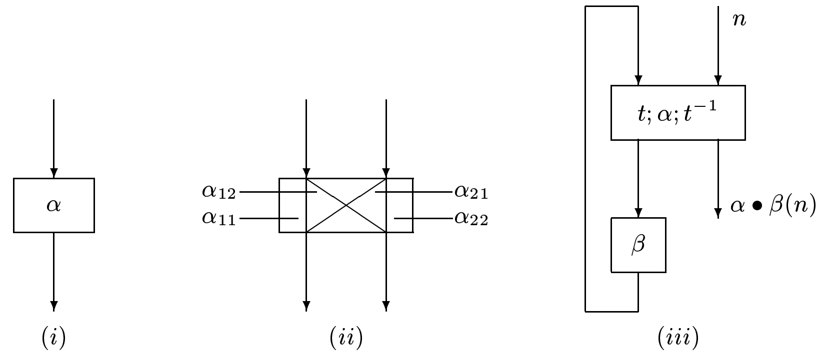


Fig. 1. Geometrical description of linear application.

The definition of the !-operation on our applicative structure is quite simple. The operation ! is intended to produce, from a single copy of α, *infinitely* many copies of α. These are obtained by simply tagging each of these copies with a natural number, i.e. we define: $!\alpha = p^{-1}; (id_{\mathrm{Nat}} \times \alpha); p$.

For the definition of the affine combinators on $([\mathrm{Nat} \rightharpoonup \mathrm{Nat}], \bullet, !)$ see [AL00].

There are many possible conditions that can be imposed on partial functions in order to cut down the space $[\mathrm{Nat} \rightharpoonup \mathrm{Nat}]$, still maintaining closure under the application, !, and all the affine combinators. The subalgebra which we are interested in is obtained by considering *partial involutions*:

Proposition 2. *Let $f : Nat \rightharpoonup Nat$. f is a* partial involution *iff its graph is a symmetric relation. Let us denote by $[Nat \rightharpoonup_{\mathsf{Inv}} Nat]$ the space of partial involutions from Nat to Nat. Then $\mathcal{A}_{\mathsf{PInv}} = ([Nat \rightharpoonup_{\mathsf{Inv}} Nat], \bullet, !)$ is an affine combinatory algebra.*

$\mathcal{A}_{\mathsf{PInv}}$ is a highly constrained algebra. Partial involutions are reminiscent of the *copy-cat* strategies of game categories. Notice that the only computational effect that the combinators have is that of *copying* information from input to output wires. Partial involutions f on a set S correspond biuniquely to pair-wise disjoint families of subsets $\{x, y\}$ of S, where $\{x, y\}$ is in the family if and only if $f(x) = y$ (and hence also $f(y) = x$). We can think of these as abstract families of "axiom links" as in the proof-nets of Linear Logic.

2 A Fully Complete Minimal Model for an Extended Calculus

In this section, we define a model for λ_k in a suitable subcategory of the co-Kleisli category of $\mathrm{PER}_{\mathcal{A}_{\mathsf{PInv}}}$. We prove that this model is fully complete w.r.t. λ_k extended with constants for all *transpositions* [1] of type $o \to o$. The proof of full completeness will use an *Approximation Theorem* and an intermediate model, which allows for partial elements.

[1] I.e. permutations which exchange exactly two elements.

We start by introducing the PER $\mathcal{O}_k$ on $\mathcal{A}_{\mathsf{PInv}}$, with k distinct equivalence classes. Our model is defined in the co-Klesli category of the affine category freely generated by $\mathcal{O}_k$.

Definition 5 (The PER $\mathcal{O}_k$). *Fix distinct natural numbers ("moves") $*_1, \dots *_k$, $a_1, \dots, a_k$. Let $\mathcal{O}_k$ be the PER on the combinatory algebra $\mathcal{A}_{\mathsf{PInv}}$ consisting of k equivalence classes defined by:*
– $c_1 = \{f : Nat \rightharpoonup_{\mathsf{Inv}} Nat \mid f(*_1) = a_1 \ \wedge \ \forall m \neq 1.\ *_m, a_m \notin dom(f)\}$
– ...
– $c_k = \{f : Nat \rightharpoonup_{\mathsf{Inv}} Nat \mid f(*_k) = a_k \ \wedge \ \forall m \neq k.\ *_m, a_m \notin dom(f)\}$.

Definition 6 ($\mathcal{M}_{\mathcal{O}_k}$). *Let $\mathcal{M}_{\mathcal{O}_k} = (CCPER_{\mathcal{O}_k}, \bullet_{\mathcal{O}_k}, [\![\ ]\!]^{\mathcal{O}_k})$ be the model of λ_k, where $CCPER_{\mathcal{O}_k}$ is the co-Klesli category of the (linear) affine category freely generated by $\mathcal{O}_k$.*

First of all notice that, by the fact that the PER $\mathcal{O}_k$ has only a finite number of equivalence classes and by extensionality of the PER model, each PER in CCPER$_{\mathcal{O}_k}$ has only *finitely many* equivalence classes. Moreover, it is easy to check that the equivalence classes of the PER $\mathcal{O}_k \multimap \mathcal{O}_k$ correspond to the permutations from $\mathcal{O}_k$ to $\mathcal{O}_k$. I.e. an involution f belongs to dom($\mathcal{O}_k \multimap \mathcal{O}_k$) if and only if $\forall c_i \exists c_j.\ f \bullet c_i = c_j$. Different permutations are in different equivalence classes of $\mathcal{O}_k \multimap \mathcal{O}_k$. It is standard that all permutations can be obtained by suitably composing elementary permutations, i.e. *transpositions*. Permutations (transpositions) of ground type are sufficient to λ-define all the elements of $\mathcal{M}_{\mathcal{O}_k}$, i.e. $\mathcal{M}_{\mathcal{O}_k}$ is fully complete. The proof of this fact is based on a *Decomposition Theorem* for the partial involutions in the domains of the PERs interpreting a simple type:

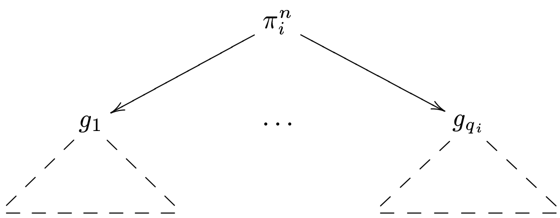
Theorem 1 (Decomposition). *Let $T = T_1 \to \dots \to T_n \to o \in SimType$, $n \geq 0$, where, for all $i = 1, \dots, n$, $T_i = U_{i1} \to \dots \to U_{iq_i} \to o$. If $f \in dom(\otimes_{i=1}^n ![\![T_i]\!]^{\mathcal{O}_k} \multimap \mathcal{O}_k)$, then*

- *either $f \in dom([\![\boldsymbol{x} : \boldsymbol{T} \vdash c_i : o]\!]^{\mathcal{O}_k})$, for some c_i*
- *or $\exists i \in \{1, \dots, n\}$, $\exists p_{\mathcal{O}_k} : \mathcal{O}_k \multimap \mathcal{O}_k$ permutation, and $\exists g_1, \dots, g_{q_i}$, where $\forall j \in \{1, \dots q_i\}$. $g_j \in dom(\otimes_{i=1}^n ![\![T_i]\!]^{\mathcal{O}_k} \multimap [\![U_{ij}]\!]^{\mathcal{O}_k})$, such that*

$$f \ (\overset{n}{\underset{i=1}{\times}} ![\![T_i]\!]^{\mathcal{O}_k} \multimap \mathcal{O}_k) \ (con_{\otimes_{i=1}^n ![\![T_i]\!]^{\mathcal{O}_k}}; (\pi_i^n \otimes \langle g_1, \dots, g_{q_i} \rangle^\dagger); Ap); p_{\mathcal{O}_k} \ ,$$

where, by abuse of notation, we denote representatives of equivalence classes of some canonical morphisms in the affine category freely generated by $\mathcal{O}_k$ by the canonical morphisms themselves.

Since the g's appearing in the Decomposition Thereom still live (up-to uncurrying) in a PER interpreting a simple type, we could keep on iterating the decomposition, expanding in turn these g's, thus getting (up-to permutations) a possible infinite tree from f:



Partial involutions which generate, under Decomposition a *finite* tree are easily proved to be λ-*definable*. Therefore, if the Decomposition Theorem holds, in order to get the full completeness result, we are only left to deal with partial involutions generating trees whose height is infinite, which would correspond to *infinite* typed Böhm trees. The proof of Theorem 1 follows a standard pattern, and it is carried out in detail in Section 2.1. The proof of λ-definability uses the Decomposition Theorem, and two furher ingredients: an *Approximation Theorem*, and an intermediate model. These are discussed in Section 2.2. From the λ-definability result, we have immediately:

Theorem 2 (Full Completeness for an Extended Language). *The model $\mathcal{M}_{\mathcal{O}_k}$ is fully complete and minimal w.r.t. λ_k, enriched with constants for all transpositions* [2] *of type $o \to o$.*

Notice that we use the full completeness result in order to show that $\mathcal{M}_{\mathcal{O}_k}$ is minimal, ie. it realizes the maximal theory on the extended calculus.

2.1 Proof of the Decomposition Theorem

Let T $\in$SimType, and $\mathcal{R}= [\![T]\!]^{\mathcal{O}_k}$, where $\mathcal{R}= \otimes_{i=1}^{n}!\ \mathcal{R}_i \multimap \mathcal{O}_k$, and, for all $i = 1, \ldots, n$, $\mathcal{R}_i= \otimes_{j=1}^{q_i}!S_{ij} \multimap \mathcal{O}_k$. Let $f \in \mathrm{dom}(\otimes_{i=1}^{n}!\ \mathcal{R}_i \multimap \mathcal{O}_k)$. We analyze the behaviour of f as operator in an application to arguments $!g_1 \in !\ \mathcal{R}_1, \ldots, !g_n \in$ $!\ \mathcal{R}_n$. I.e., let us apply the coding functions t, p of Section 1.2, in order to get

$$\overline{f} : ((\mathrm{Nat} \times \mathrm{Nat}) + \ldots + (\mathrm{Nat} \times \mathrm{Nat})) + \mathrm{Nat} \rightharpoonup ((\mathrm{Nat} \times \mathrm{Nat}) + \ldots + (\mathrm{Nat} \times \mathrm{Nat})) + \mathrm{Nat}\ ,$$

where in the domain (codomain) of $\overline{f}$ there are n occurrences of Nat $\times$ Nat, each one corresponding to one of the n arguments to which f applies. In the interaction with the i-th argument $!g_i$ only the i-th occurrence of Nat $\times$ Nat in the domain (codomain) of $\overline{f}$ is involved. In particular, the lefthand occurrence of Nat in Nat$\times$Nat refers to the copy of the argument $!g_i$ used, while the righthand occurence of Nat carries the values from (to) g_i.

We have two possibilities, according to the behaviour of $\overline{f}$ on the inputs $*_1, \ldots, *_k$ (other input values are not relevant, by definition of the PER $\mathcal{O}_k$):

Lemma 1. *Let $f \in dom(\boldsymbol{\mathcal{R}} \multimap \mathcal{O}_k)$, where $\mathcal{R}= \otimes_{i=1}^{n}!\ \mathcal{R}_i \multimap \mathcal{O}_k$, and, for all $i = 1, \ldots, n$, $\mathcal{R}_i= \otimes_{j=1}^{q_i}!S_{ij} \multimap \mathcal{O}_k$. Then*

[2] I.e. permutations which exchange exactly two elements.

1. *either* $\exists *_j \,.\, \overline{f}(r, *_j) = (r, a_j) \ \wedge \ \forall *_k \neq q_j.\ ((r, *_k), (r, a_k)) \notin dom(\overline{f})$.
   I.e. $f \in [\![\boldsymbol{x} : \boldsymbol{T} \vdash c_j : o]\!]^{\mathcal{O}_k}$.
2. *or* $\exists i \in \{1, \ldots, n\}$ *such that*
   (a) $\forall *_j \ \exists i_0.\ \overline{f}(r, *_j) = (l, (i, (i_0, m))) \ \wedge \ m = (r, y_k)$ *where* $y_k \in \{*_k, a_k\}$ *and*
   (b) $\forall i, j\ (\overline{f}(r, *_j) = (l, (i, (i_0, (r, *_k)))) \ \Rightarrow \ \overline{f}(l, (i, (i_0, (r, a_k)))) = (r, a_j) \ \wedge$
   $\overline{f}(r, *_j) = (l, (i, (i_0, (r, a_k)))) \ \Rightarrow \ \overline{f}(l, (i, (i_0, (r, *_k)))) = (r, a_j)$.

Proof. Item 1 is easy to prove. We focus on the proof of item 2. As far as item 2a, one can easily show that, for any given $*_j$, $\exists i, i_0$ s.t. $\overline{f}(r, *_j) = (l, (i, (i_0, m)))$ and $m = (r, y_k)$. The proof of the fact that, for all $*_j$, the argument i interrogated by f is the same is based on a "counting argument". If we "split" the responses to initial questions $*_j$ among different arguments, then we lose totality, because of the constraints of being a partial involution. Namely, assume by contradiction that it is not the case that for all initial questions $*_j$ the responses are in the same argument. Then, for all $i = 1, \ldots, n$ $\exists p_i$ s.t.

$$\forall *_j \ \forall i_0 \in \mathrm{Nat}.\ \overline{f}(r, *_j) \neq (l, (i, (i_0, (r, x_{p_i})))) \ , \text{for } x \in \{*, a\}.$$

Then consider constants $(\mathbf{K}c_{p_1}), \ldots, (\mathbf{K}c_{p_n})$ in $\mathcal{R}_1, \ldots, \mathcal{R}_n$. Then we have $\forall *_j \,.\, f \bullet (\mathbf{K}c_{p_1}) \ldots (\mathbf{K}c_{p_n})(*_j) \uparrow$, i.e. $f \notin \mathrm{dom}(\boldsymbol{\mathcal{R}} \multimap \mathcal{O}_k)$. Contradiction.

Finally, in order to prove item 2b, one can proceed by contradiction, and by case analysis. This concludes the proof of Lemma 1. □

Using Lemma 1 above, one can easily prove the following two lemmata: *Linearization of Head Occurrence* and *Linear Function Extensionality.* The factorization of the proof of the Decomposition Theorem in these two lemmata is standard, but notice the special form of *Linear Function Extensionality*, where permutations come into play.

Lemma 2 (Linearization of Head Occurrence). *Let* $\mathcal{R} = \otimes_{i=1}^{n} !\, \mathcal{R}_i \multimap \mathcal{O}_k$, *where, for all* $i = 1, \ldots, n$, $\mathcal{R}_i = \otimes_{j=1}^{q_i} !S_{ij} \multimap \mathcal{O}_k$. *Then, for all* $f \in dom(\boldsymbol{\mathcal{R}} \multimap \mathcal{O}_k)$, *where* $\boldsymbol{\mathcal{R}}$ *is an abbreviation for* $\otimes_{i=1}^{n} !\, \mathcal{R}_i$, *there exist* $i \in \{1, \ldots, n\}$ *and* $f' \in dom(R_i \multimap \boldsymbol{\mathcal{R}} \multimap \mathcal{O}_k)$ strict, *i.e.* $t + t; t; f'; t^{-1}; t^{-1} + t^{-1}(r, (r, *)) = (l, (r, *))$, *such that* $\quad f \ \mathcal{R} \ (con_{\boldsymbol{\mathcal{R}}}; \pi_i^n \otimes id_{\boldsymbol{\mathcal{R}}}; \Lambda^{-1}(f'))$. □

Now we examine the structure of f'. One can show that:

Lemma 3 (Linear Function Extensionality). *Let* $\mathcal{S}, \mathcal{R}$ *be PERs. Then, for all* $f \in dom((\mathcal{S} \multimap \mathcal{O}_k) \multimap (\mathcal{R} \multimap \mathcal{O}_k))$ *strict, there exists* $f' \in dom(\mathcal{R} \multimap \mathcal{S})$ *such that* $\quad f \ \ ((\mathcal{S} \multimap \mathcal{O}_k) \multimap (\mathcal{R} \multimap \mathcal{O}_k)) \ \ (\Lambda(((id_{\mathcal{S}} \multimap p_{\mathcal{O}_k}) \otimes f'); Ap))$,
where $p_{\mathcal{O}_k}$ *is a permutation in the PER* $\mathcal{O}_k \multimap \mathcal{O}_k$.

The last technical lemma that we need in order to prove the Decomposition Theorem amounts to a general fact which follows by construction of the category of PERs over an ACA (see Section 1.2):

Lemma 4 (Uniformity of Threads). *The following isomorphism holds, for all PERs* $\mathcal{R}, \mathcal{S}$,

$$(id_{!\mathcal{R}} \multimap der_{\mathcal{S}}) \ : \ \ (!\, \mathcal{R} \multimap !\, \mathcal{S} \simeq \ !\, \mathcal{R} \multimap \mathcal{S}) \ \ : \ (\lambda[f] \in !\, \mathcal{R} \multimap \mathcal{S} \,.\, ([f])^{\dagger}_{\mathcal{R},\mathcal{S}}) \ ,$$

where $(\ )^{\dagger}_{\mathcal{R},\mathcal{S}}$ *is the canonical morphism induced by the comonad* $!$.

Finally, we have:

Proof of the Decomposition Theorem 1. If case 1 of Lemma 1 applies, then we are done. If case 2 applies, then, by Lemma 2, there exist $i \in \{1, \dots, n\}$ and $f' \in \mathrm{dom}(\mathcal{R}_i \multimap (\boldsymbol{\mathcal{R}} \multimap \mathcal{O}_k))$ s.t. $f\ (\boldsymbol{\mathcal{R}} \multimap \mathcal{O}_k)\ (\mathrm{con}_{\boldsymbol{\mathcal{R}}}; \pi_i^n \otimes \mathrm{id}_{\boldsymbol{\mathcal{R}}}; \Lambda^{-1}(f'))$. By Lemma 3, $\exists g, p_{\mathcal{O}_k}$ s.t. $f'\ ((\boldsymbol{\mathcal{S}_i} \multimap \mathcal{O}_k) \multimap (\boldsymbol{\mathcal{R}} \multimap \mathcal{O}_k))\ \Lambda(((\mathrm{id}_{\boldsymbol{\mathcal{S}_i}} \multimap p_{\mathcal{O}_k}) \otimes g); \mathrm{Ap})$. Then $f\ (\boldsymbol{\mathcal{R}} \multimap \mathcal{O}_k)\ \mathrm{con}_{\boldsymbol{\mathcal{R}}}; (\mathrm{id}_{\boldsymbol{\mathcal{S}_i}} \multimap p_{\mathcal{O}_k}); \pi_i^n \otimes g; \mathrm{Ap}$. Finally, by Lemma 4, by definition of the product of PERs and by the universality property of the product, we obtain $g\ (\boldsymbol{\mathcal{R}} \multimap \boldsymbol{\mathcal{S}_i})\ \langle g_1, \dots, g_{q_i} \rangle^{\dagger}$, for some $g_1 \in \mathcal{S}_{i1}, \dots, g_{q_i} \in \mathcal{S}_{iq_i}$. □

2.2 Proof of $\boldsymbol{\lambda}$-definability

The proof of the λ-definability property of $\mathcal{M}_{\mathcal{O}_k}$ is quite involved and it uses an *Approximation Theorem* and an intermediate PER model $\mathcal{M}_{\mathcal{O}_k^{\perp}}$ for the simply typed λ-calculus λ_k with k ground constants plus an extra *undefined* ground constant $\perp$. This model allows for *partial* elements (*approximants*).

The Approximation Theorem. We start by introducing the notion of *approximant* of a partial involution. By repeatedly applying the Decomposition Theorem to a partial involution f in the model $\mathcal{M}_{\mathcal{O}_k}$, we obtain a (possibly) infinite typed Böhm tree (up to permutations). The j-th approximant of f is a partial involution obtained by truncating at level j this tree, and by substituting the *empty partial involution* for each possibly erased subtree. Notice that approximants are not in the model, in general. Formally:

Definition 7 (Approximants).
Let $f \in dom(\mathcal{R})$, where $\mathcal{R} = [\![T]\!]^{\mathcal{O}_k}$, for some simple type T.

- *We define the* j-th tree, *$t_j(f)$, of height at most j, generated from f after iterated applications of the Decomposition Theorem by induction on j as follows:*
  - *$t_0(f)$ is the tree of height* 0 *with only a root labeled by f;*
  - *given the tree $t_j(f)$ of height at most j, the tree $t_{j+1}(f)$ is obtained from the tree $t_j(f)$ by expanding the possible leaves at level j via the Decomposition Theorem.*
- *We define the* j-th approximant *of f, $p_j(f)$, as the partial involution obtained from the tree $t_j(f)$ by substituting any partial involution at level j by the* empty *partial involution.*

By monotonicity of $\bullet$ we have immediately:

Lemma 5. *Let $T_1 \to \dots \to T_n \to o \in SimType$, let $f \in dom(\bigotimes_{i=1}^{n} ![\![T_i]\!]^{\mathcal{O}_k} \multimap \mathcal{O}_k)$. Then $\forall j \geq 0.\ p_j(f) \subseteq p_{j+1}(f)$.*

Theorem 3 (Approximation). *Let $T_1 \to \dots \to T_n \to o$ be a simple type, and let $f \in dom(\bigotimes_{i=1}^{n} ![\![T_i]\!]^{\mathcal{O}_k} \multimap \mathcal{O}_k)$. Then*

$$f\ (\bigotimes_{i=1}^{n} ![\![T_i]\!]^{\mathcal{O}_k} \multimap \mathcal{O}_k)\ (\bigcup_{j \in \omega} p_j(f))\ .$$

Proof. We have to show that, $\forall \boldsymbol{g} \in \bigotimes_{i=1}^{n} ![\![T_i]\!]^{\mathcal{O}_k}$, $f \bullet \boldsymbol{g}\ \mathcal{O}_k\ (\bigcup_{j\in\omega} p_j(f)) \bullet \boldsymbol{g}$. I.e.:
i) $\forall *_i\ .\ f \bullet \boldsymbol{g}(*_i) \uparrow \iff (\bigcup_{j\in\omega} p_j(f)) \bullet \boldsymbol{g}(*_i) \uparrow$ and
ii) $\forall *_i, a_i.\ f \bullet \boldsymbol{g}(*_i) = a_i \iff (\bigcup_{j\in\omega} p_j(f)) \bullet \boldsymbol{g}(*_i) = a_i$.
The implications ($\Rightarrow$) in i) and ($\Leftarrow$) in ii) follow from monotonicity of $\bullet$ and from the fact that, for all j the graph of $p_j(f)$ is contained in the graph of f. In order to show i)($\Leftarrow$) and ii)($\Rightarrow$), one can check that, if $f \bullet \boldsymbol{g}(*_i) = a_i$ and the result is obtained with a thread of length at most $2j$, then $p_j(f) \bullet \boldsymbol{g}(*_i) = a_i$. $\square$

The PER model $\mathcal{M}_{\mathcal{O}_k^\perp}$.

Definition 8. *Let $\mathcal{M}_{\mathcal{O}_k^\perp}$ be the PER model induced by the ground PER $\mathcal{O}_k^\perp$ defined as follows. Fix distinct natural numbers $*_1, \ldots *_k$, $a_1, \ldots, a_k$. Let $\mathcal{O}_k^\perp$ be the PER on the ACA $\mathcal{A}_{\mathsf{PInv}}$ consisting of $k+1$ equivalence classes defined by:*
- $\perp = \{f : Nat \rightharpoonup_{\mathsf{Inv}} Nat \mid \forall i \in \{1, \ldots, k\}.\ *_i, a_i \notin dom(f)\}$
- $c_1 = \{f : Nat \rightharpoonup_{\mathsf{Inv}} Nat \mid f(*_1) = a_1\ \wedge\ \forall m \neq 1.\ *_m, a_m \notin dom(f)\}$
- $\ldots$
- $c_k = \{f : Nat \rightharpoonup_{\mathsf{Inv}} Nat \mid f(*_k) = a_k\ \wedge\ \forall m \neq k.\ *_m, a_m \notin dom(f)\}$.

$\mathcal{M}_{\mathcal{O}_k^\perp}$ is a model of $\lambda_k^\perp$, i.e. the simply typed λ-calculus with k ground constants plus the extra ground constant $\perp$. In particular, approximants are in $\mathcal{M}_{\mathcal{O}_k^\perp}$. The relationship with the model $\mathcal{M}_{\mathcal{O}_k}$ is given by the following lemma:

Lemma 6. *Let $f \in dom([\![\boldsymbol{T} \multimap o]\!]^{\mathcal{O}_k})$. Then*
i) $f \in dom([\![\boldsymbol{T} \multimap o]\!]^{\mathcal{O}_k^\perp})$.
ii) $\exists J \geq 0.\ f\ [\![\boldsymbol{T} \multimap o]\!]^{\mathcal{O}_k^\perp}\ p_J(f)$.

Proof. i) Assume that $f \in \mathrm{dom}([\![\boldsymbol{T} \multimap o]\!]^{\mathcal{O}_k})$. By the Approximation Theorem 3, $f \sim \bigcup_{j\in\omega} p_j(f)$ in $\mathcal{M}_{\mathcal{O}_k}$. Moreover, for all j, $p_j(f) \in \mathrm{dom}([\![\boldsymbol{T} \multimap o]\!]^{\mathcal{O}_k^\perp})$, and hence also $\bigcup_{j\in\omega} p_j(f) \in \mathrm{dom}([\![\boldsymbol{T} \multimap o]\!]^{\mathcal{O}_k^\perp})$. Then, using an argument similar to the one used in the proof of the Approximation Theorem, one can check that, $\forall \boldsymbol{g} \in [\![\boldsymbol{T}]\!]^{\mathcal{O}_k^\perp}.\ \forall *_i\ .\ f \bullet \boldsymbol{g}(*_i) \simeq \bigcup_{j\in\omega} p_j(f) \bullet \boldsymbol{g}(*_i)$. Hence, in particular, $f \in \mathrm{dom}([\![\boldsymbol{T} \multimap o]\!]^{\mathcal{O}_k^\perp})$.
ii) By the proof of item i) of this lemma, $f \sim \bigcup_{j\in\omega} p_j(f)$ in $\mathcal{M}_{\mathcal{O}_k^\perp}$, and $\forall j.\ p_j(f) \in \mathrm{dom}([\![\boldsymbol{T} \multimap o]\!]^{\mathcal{O}_k^\perp})$. Moreover, since $[\![\boldsymbol{T} \multimap o]\!]^{\mathcal{O}_k^\perp}$ has only finitely many equivalence classes, by Lemma 5, there exists $J \geq 0$ such that, for all $m_1, m_2 \geq J$, $p_{m_1}(f)[\![\boldsymbol{T} \multimap o]\!]^{\mathcal{O}_k^\perp} p_{m_2}(f)$. Hence $f \sim p_J(f)$ in $\mathcal{M}_{\mathcal{O}_k^\perp}$. $\square$

λ-definability of $\mathcal{M}_{\mathcal{O}_k}$. Finally, we are in the position of proving that all partial involutions $f \in \mathrm{dom}([\![\boldsymbol{T} \multimap o]\!]^{\mathcal{O}_k})$ are λ-definable. We proceed by induction on types. The base case is easy. Let us consider the induction step.

By the Approximation Theorem 3, $f[\![\boldsymbol{T} \multimap o]\!]^{\mathcal{O}_k}(\bigcup_{j\in\omega} p_j(f))$. By Lemma 6ii), there exists J such that $f \sim p_J(f)$ in the model $\mathcal{M}_{\mathcal{O}_k^\perp}$. Hence, using Lemma 6i),

we have, in particular, that: $\forall \boldsymbol{g} \in [\![T]\!]^{\mathcal{O}_k}.\ \forall *_i .\ f \bullet \boldsymbol{g}(*_i) \simeq p_J(f) \bullet \boldsymbol{g}(*_i)$.
Therefore, $p_J(f) \sim f$ in $\mathcal{M}_{\mathcal{O}_k}$. Let us call P_J the λ-term whose interpretation in $\mathcal{M}_{\mathcal{O}_k^\perp}$ is $p_J(f)$. Two cases can arise:
1) $\exists \boldsymbol{M} : \boldsymbol{T}$, $\boldsymbol{M}$ $\perp$-free, such that $P_J \boldsymbol{M} =_\beta \perp$.
2) $\forall \boldsymbol{M} : \boldsymbol{T}$, $\boldsymbol{M}$ $\perp$-free, such that $P_J \boldsymbol{M} \neq_\beta \perp$.
If case 1 applies, then we have a contradiction, since $\perp \notin \mathcal{O}_k$. If case 2 applies, then one can check that P_J is equivalent in the maximal theory to any λ-term P' obtained from P_J by substituting any constant $c \neq \perp$ for each possible occurrence of $\perp$ in P_J. But then, since, by induction hypothesis, $\forall \boldsymbol{g} \in [\![T]\!]^{\mathcal{O}_k}$, $\boldsymbol{g}$ is λ-definable, $[\![P']\!]^{\mathcal{O}_k} \bullet \boldsymbol{g} \sim [\![P_J]\!]^{\mathcal{O}_k} \bullet \boldsymbol{g}$ in $\mathcal{M}_{\mathcal{O}_k}$, and hence $[\![P']\!]^{\mathcal{O}_k} \sim p_J(f) \sim f$, i.e. f is λ-definable.

This concludes the proof of Theorem 2.

3 A Fully Complete Minimal Model for the Simply Typed $\boldsymbol{\lambda}$-calculus

One can build a fully complete model for the simply typed λ-calculus λ_k, by getting rid of permutations in the models $\mathcal{M}_{\mathcal{O}_k}$. There are two ways of doing this. The first is "low-level", and it amounts to cutting down the affine combinatory algebra of partial involutions, by placing additional constraints on the partial involutions, similarly to what one does in [AL99a] for getting a fully abstract model for PCF. Alternatively, one can define a suitable *logical relation* and use it to cut down the PER model. We briefly sketch the first technique.

For the sake of simplicitly, let us consider, in place of the set of natural numbers, the following set of inductively defined moves:

Definition 9. *Let $k \in Nat$. We define*

$$(\boldsymbol{M}_k \ni)\ m\ ::=\ *_i\ \mid\ a_i\ \mid\ (l,m)\ \mid\ (r,m)\ \mid\ \langle j,m\rangle\ ,$$

where $i = 1, \ldots, k$ and $j \in Nat$.
We regard $\boldsymbol{M}_k$ as equipped with the intrinsic coding functions $[l,r] : \boldsymbol{M}_k + \boldsymbol{M}_k \to \boldsymbol{M}_k$, and $\langle\ ,\ \rangle : Nat \times \boldsymbol{M}_k \to \boldsymbol{M}_k$.

One could equivalently take the moves to be natural numbers (under suitable assumptions on coding functions), but the set $\mathbf{M}_k$ simplifies the argument. We can immediately define a function v on moves, which, for any move m, provides the index i of the basic move $*_i$ or a_i which the move m is made up. I.e.:

Definition 10. *Let $v : \boldsymbol{M}_k \to \boldsymbol{M}_k$ be defined as follows. For all $i \in Nat$, for all $m \in \boldsymbol{M}_k$,*
$v(*_i) = v(a_i) = i$
$v((l,m)) = v((r,m)) = v(\langle i,m\rangle) = v(m)$.

Partial involutions which preserves the function v still form an affine combinatory algebra.

Proposition 3 (Full Completeness and Minimality). *Let $\mathcal{A}_{\mathtt{vPInv}}$ be the affine combinatory algebra whose carrier is the set of partial involutions $f : \boldsymbol{M}_k \to \boldsymbol{M}_k$ such that, for all $m \in dom(f)$, $v(f(m)) = v(m)$. Then the model induced by the PER $\mathcal{O}_k$ over $\mathcal{A}_{\mathtt{vPInv}}$ is fully complete and minimal w.r.t. λ_k.*

4 Some Applications of the Full Completeness Results

Our fully complete model $\mathcal{M}_{\mathcal{O}_k}$ provides immediately *semantical* proofs of some interesting facts concerning the maximal theory $\approx$ over the simply typed λ-calculus with or without permutations.

Context Lemma.

Definition 11 (Applicative Equivalence). *Let $\approx^{\mathsf{app}} \subseteq \Lambda^0 \times \Lambda^0$ be defined by*

$$M \approx^{\mathsf{app}} N \;\Leftrightarrow\; \forall P_1, \ldots, P_n \in \Lambda^0.\; MP_1 \ldots P_n =_\beta NP_1 \ldots P_n \;.$$

Lemma 7 (Context Lemma). *The theory $\approx$ admits an* applicative *characterization, i.e.* $M \approx N \iff M \approx^{\mathsf{app}} N$.

The Theory on Infinitary Böhm Trees is Conservative. Let us call $\mathcal{M}_{\mathcal{O}}$ the the variant of the model $\mathcal{M}_{\mathcal{O}_k^\perp}$ obtained by considering two ground constants, $\perp$ and $\top$, and by defining the ground Sierpinski PER as follows: fix $* \in$ Nat, $\perp = \{f : \mathrm{Nat} \rightharpoonup_{\mathsf{Inv}} \mathrm{Nat} \mid * \notin \mathrm{dom}(f)\}$, $\top = \{f : \mathrm{Nat} \rightharpoonup_{\mathsf{Inv}} \mathrm{Nat} \mid f(*) = *\}$.

Then one can check that $\mathcal{M}_{\mathcal{O}}$ is fully complete and minimal directly for the simply typed λ-calculus with two ground constants. Actually $\mathcal{M}_{\mathcal{O}}$ is a fully complete model for the maximal theory on the *infinitary calculus*, i.e. the simply typed, possibly infinite, Böhm trees:

Definition 12 (Infinitary Typed Böhm Trees). *We define the* infinitary *typed Böhm trees as the trees obtained as supremums of typed Böhm trees corresponding to approximants.*

Using the model $\mathcal{M}_{\mathcal{O}}$, we can show that, in the case of two ground constants, the theory over infinitary typed Böhm trees $\approx^\infty$ is a conservative extension of $\approx$ w.r.t. terms in $\lambda_1^\perp$, i.e.:

Proposition 4. $\approx^\infty_{|\mathtt{TBT}} = \approx_{|\mathtt{TBT}}$,
where $\approx^\infty_{|\mathtt{TBT}}$, $\approx_{|\mathtt{TBT}}$, denote the theory $\approx^\infty$ and the theory $\approx$ restricted to the finite typed Böhm trees, respectively, in the case of two ground constants.

Decidability Results. The following decidability result can be proved in $\mathcal{M}_{\mathcal{O}_k}$:

Theorem 4. *Let $\mathcal{R}$ be a simple PER, i.e. $\mathcal{R}{=}\mathcal{R}_1{\to} \ldots {\to}\mathcal{R}_n{\to}\mathcal{O}_k$. For all $f : Nat \to Nat$ whose graph is finite, it is decidable whether $f \in dom(\mathcal{R})$.*

Proof. In order to decide whether $f \in \mathrm{dom}(\mathcal{R})$, it is sufficient to check the behaviour of f when applied to the "relevant" $\boldsymbol{g} \in \boldsymbol{\mathcal{R}}$, i.e. to the g's whose domains (and codomains), roughly, are contained in a suitable subset of the

domains of h, h'. More precisely, let
$\overline{h}, \overline{h}' : ((\text{Nat} \times \text{Nat}) + \ldots + (\text{Nat} \times \text{Nat})) + \text{Nat} \rightharpoonup$
$((\text{Nat} \times \text{Nat}) + \ldots + (\text{Nat} \times \text{Nat})) + \text{Nat}$
be the partial involutions obtained from h, h' using the coding functions t, p. Then g_i is "relevant" to h, h' if $\text{dom}(g_i) \subseteq \{n \mid \exists k.\ (l,(i,(k,n))) \in \text{dom}(\overline{h}) \cup \text{dom}(\overline{h}')\}$. These $\boldsymbol{g}$'s are the only "relevant" ones for h, h' in the sense that, for any other "non-relevant" $\boldsymbol{g}$, there exists $\boldsymbol{g}'$ "relevant" such that, for all $*_i$, $h \bullet \boldsymbol{g}(*_i) \simeq h \bullet \boldsymbol{g}'(*_i)$ and $h' \bullet \boldsymbol{g}(*_i) \simeq h' \bullet \boldsymbol{g}'(*_i)$. Since h, h' have finite graphs, then there are only finitely many $g_1, \ldots, g_n$ whose graphs are finite. We can easily generate all these "relevant" partial involutions $g_1, \ldots, g_n$. At this point, we have to eliminate the relevant $\boldsymbol{g}$'s which are not in $\text{dom}(\boldsymbol{\mathcal{R}})$. Moreover, we need to know, for all relevant g_i, g_i', whether g_i is equivalent to g_i'. In order to decide this, we compute, in turn, the partial involutions which are relevant for g_i and g_i'. And, recursively, we have to compute the relevant partial involutions of the relevant partial involutions, until we reach the ground PER $\mathcal{R}$. Once we have eliminated those $\boldsymbol{g}$ which are not in $\text{dom}(\boldsymbol{\mathcal{R}})$, and we have divided the set of relevant $\boldsymbol{g}$'s in equivalence classes, we can check, finally, the applicative behaviour of f. The computations $h \bullet \boldsymbol{g}(*_i)$ and $h' \bullet \boldsymbol{g}(*_i)$ always terminate, since, by definition of partial involution, and by the fact that the graphs of $h, h', g_1, \ldots, g_n$ are all finite, there cannot be an infinite (possibly cyclic) computation. Namely, the computation $h \bullet \boldsymbol{g}(*_i)$ either converges to a_i or diverges because h or $\boldsymbol{g}$ are not defined on some element. This concludes the proof.

$\square$

Using a similar argument, we can prove:

Proposition 5. *Let $\vdash M : T$, $\vdash N : T$ be such that $[\![\vdash M : T]\!]^{\mathcal{O}_k}$, $[\![\vdash N : T]\!]^{\mathcal{O}_k}$ have representatives with finite graphs. Then it is decidable whether $M \approx N$.*

However, there are very few λ-terms whose interpretation is finite in the sense of Proposition 5 above. E.g. the identity of type $((o \to o) \to o) \to ((o \to o) \to o)$ has no representatives with finite graph. Intuitively, this depends on the fact that it can ask for *any* number of copies of its argument.

5 Conclusions and Future Work

In this paper, we have studied *fully complete* PER models for the *maximal theory* of the simply typed λ-calculus with finitely many base constants, and we have seen some applications of this construction. Here we summarize a list of remarks and interesting issues which remain to be addressed:
– Following [Abr00], we could abstract axioms for full completeness from the lemmata in our proof. However, these would not imply faithfulness w.r.t. the maximal theory, i.e. that the theory of models would be maximal.
– One could define fully complete Game Models for λ_k by considering strategies in the style of [AJM00]. But, by the intensionality of the Game Semantics, these models would not realize the maximal theory, but rather the $\beta\eta$-theory.

– In [Lai00], a fully abstract translation of "Finitary μ-PCF" into the simply typed lambda calculus with constants is given. An interesting consequence of this result is that our model is also fully abstract for this finitary μ-PCF.
– We feel that the fully complete models based on partial involutions defined in this paper should provide a semantical proof of the decidability of the maximal theory, alternative to those of Padovani and Loader. We should capitalize on the possibility of checking equivalence of involutions by evaluating them on *finite* sets of inputs (moves).
– We conjecture that the model $\mathcal{M}_{\mathcal{O}_k^\perp}$, for $k = 1$, is fully complete for the decidable fragment of PCF called "Unary PCF".
– We end this paper with avery speculative comment. Partial involutions provide models for *reversible* computations ([Abr01]). Is this related to the decidability of the various theories which can be modeled by partial involutions?

References

[Abr96] S.Abramsky. Retracing some paths in Process Algebra, *Concur'96*, U. Montanari and V. Sassone eds., 1996, 1–17.

[Abr97] S.Abramsky. Interaction, Combinators, and Complexity, Notes, Siena (Italy), 1997.

[Abr00] S.Abramsky. Axioms for Definability and Full Completeness, in *Proof, Language and Interaction: Essays in Honour of Robin Milner*, G. Plotkin, C. Stirling and M. Tofte, eds., MIT Press, 2000, 55–75.

[Abr01] S.Abramsky. A Structural Approach to Reversible Computation, May 2001.

[AHS98] S.Abramsky, E.Haghverdi, P.Scott. Geometry of Interaction and Models of Combinatory Logic, 1998, to appear.

[AJ94] S.Abramsky, R.Jagadeesan. New foundations for the Geometry of Interaction, *Inf. and Comp.* **111**(1), 1994, 53–119.

[AJ94a] S.Abramsky, R.Jagadeesan. Games and Full Completeness for Multiplicative Linear Logic, *J. of Symbolic Logic* **59**(2), 1994, 543–574.

[AJM00] S.Abramsky, R.Jagadeesan, P.Malacaria. Full Abstraction for PCF, *Inf. and Comp.* **163**, 2000, 409–470.

[AL99] S.Abramsky, M.Lenisa. Fully Complete Models for ML Polymorphic Types, Technical Report **ECS-LFCS-99-414**, LFCS, 1999 (available at http://www.dimi.uniud.it/~lenisa/Papers/Soft-copy-ps/lfcs99.ps.gz).

[AL00] S.Abramsky, M.Lenisa. A Fully-complete PER Model for ML Polymorphic Types, *CSL'00*, LNCS **1862**, 2000, 140–155.

[AL99a] S.Abramsky, J.Longley. Realizability models based on history-free strategies, 1999.

[Gir89] J.Y.Girard. Towards a Geometry of Interaction, *Contemporary Mathematics* **92**, 1989, 69–108.

[KNO99] A.Ker, H.Nickau, L.Ong. More Universal Game Models of Untyped λ-Calculus: The Böhm Tree Strikes Back, *CSL'99*, LNCS, 1999.

[JSV96] A.Joyal, R.Street, D.Verity. Traced monoidal categories, *Math. Proc. Comb. Phil. Soc.* **119**, 1996, 447–468.

[Lai00] J.Laird. Games, control and full abstraction, February 2000.

[Loa97] An Algorithm for the Minimal Model, Note, 1997.

[Pad95] V.Padovani. Decidability of all Minimal Models, *TYPES'95*, LNCS **1185**, 201–215.