Compositional Characterisations of λ-terms using Intersection Types ⋆

M. Dezani-Ciancaglini [a] F. Honsell [b] Y. Motohama [b]

[a] *Dipartimento di Informatica, Università di Torino, corso Svizzera 185, 10149 Torino, Italy*

[b] *Dipartimento di Matematica ed Informatica, Università di Udine, via delle Scienze 208, 33100 Udine, Italy*

Abstract

We show how to characterise *compositionally* a number of *evaluation* properties of λ-terms using *Intersection Type* assignment systems. In particular, we focus on termination properties, such as strong normalisation, normalisation, head normalisation, and weak head normalisation. We consider also the persistent versions of such notions. By way of example, we consider also another evaluation property, unrelated to termination, namely reducibility to a closed term.

Many of these characterisation results are new, to our knowledge, or else they streamline, strengthen, or generalise earlier results in the literature.

The completeness parts of the characterisations are proved uniformly for all the properties, using a *set-theoretical* semantics of intersection types over suitable kinds of *stable* sets. This technique generalises Krivine's and Mitchell's methods for strong normalisation to other evaluation properties.

Key words: λ-calculus; Intersection Types; Normalisation Properties; Set-theoretical Semantics of Types.

⋆ Partially supported by EU within the FET - Global Computing initiative, project DART ST-2001-33477 and by MURST Cofin'01 project COMETA, MURST Cofin'02 project McTati. The funding bodies are not responsible for any use that might be made of the results presented here.

Email addresses: mdezani@di.unito.it (M. Dezani-Ciancaglini), honsell@dimi.uniud.it (F. Honsell), motohama@dimi.uniud.it (Y. Motohama).

Introduction

The *intersection-types discipline* was introduced in [10] as a means of overcoming the limitations of Curry's type assignment system. Subsequently it was used in [5] as a tool for proving Scott's conjecture concerning the *completeness* of the set-theoretic semantics for *simple types.*

Very early on, however, it was realised that intersection type theories are a very expressive tool for giving *compositional* characterisations (i.e. a characterisations based on properties of proper subterms) of *evaluation* properties of λ-terms. There are two seminal results in this respect.

The first result is that the Ω-free fragment of intersection-types allows one to type *all* and *only* the *strongly normalising terms.* This is largely a folklore result; the first published proof appears in [25].

The second result is the *filter model* construction based on the intersection type theory $\mathcal{BCD}$, carried out in [5]. This result was the first to show that there is a very tight connection between intersection types and *compact elements* in ω-algebraic denotational models of λ-calculus. This connection later received a categorically principled explanation by Abramsky in the broader perspective of "domain theory in logical form" [1].

Since then, the number of intersection type theories, used for studying the fine structure of the denotational semantics of untyped λ-calculus, has increased considerably (e.g. [12,11,19,16,2,24,18]). In all these cases the corresponding intersection type assignment systems are used to provide *finite logical presentations* of particular domain models, which can thereby be viewed also as *filter models.* And hence, intersection type theories provide characterisations of particular semantical properties.

In this paper we address the problem of investigating *uniformly* the use of intersection type theories, and corresponding type assignment systems, for giving a *compositional* characterisation of *evaluation* properties of λ-terms.

In particular we discuss *termination* properties such as *strong normalisation, normalisation, head normalisation, weak head normalisation.* We consider also the *persistent* versions of such notions (see Definition 2.2). By way of example we consider also another evaluation property, unrelated to termination, namely reducibility to a *closed term.*

Many of the characterisation results that we give are indeed inspired by earlier semantical work on filter models of the untyped λ-calculus, but they are rather novel in spirit. We focus, in fact, on *proof-theoretic* properties of intersection type assignment systems *per se.* Most of our characterisations are therefore new, to our knowledge, or else they streamline, strengthen, or generalise earlier results in the literature.

The *completeness* part of the characterisations is proved uniformly for all the properties. We use a very elementary presentation of the technique of *logical relations* phrased in terms of a *set-theoretical* semantics of intersection types over suitable kinds of *stable* sets. This technique generalises Krivine's [20] and Mitchell's [22] proof methods for *strong normalisation*, to other evaluation properties.

The paper is organised as follows. In Section 1 we introduce the intersection type language, intersection type theories and type assignment systems. We prove also some general results about such systems. In Section 2 we introduce the various properties of λ-terms on which we shall focus. In Section 3 we give the compositional characterisations of such properties and we prove the *soundness* of the characterisations. Completeness is proved in Section 4. Final remarks and open problems appear in Section 5. The auxiliary notion of *polarised normal form*, which is instrumental to the study of persistent normal forms, is discussed in the Appendix.

An extended abstract of the present paper is [15].

1 Intersection type theories and type assignment systems

Intersection types are syntactical objects which are built inductively by closing a given set $\mathbf{C}$ of *type atoms* (constants) under the *function type* constructor $\rightarrow$ and the *intersection* type constructor $\cap$.

Definition 1.1 (Intersection type languages) *The* intersection type language *over* $\mathbf{C}$*, denoted by* $\mathbf{T} = \mathbf{T}(\mathbf{C})$*, is defined by the following abstract syntax:*

$$\mathbf{T} = \mathbf{C} \mid \mathbf{T} {\rightarrow} \mathbf{T} \mid \mathbf{T} \cap \mathbf{T}. \blacksquare$$

Notation 1.2 *Upper case Roman letters i.e.* $A, B, \ldots$*, will denote arbitrary types. In writing intersection-types we shall use the following convention: the constructor* $\cap$ *takes precedence over the constructor* $\rightarrow$ *and both associate to the right. Moreover* $A^n \rightarrow B$ *will be short for* $\underbrace{A \rightarrow \cdots \rightarrow A}_{n} \rightarrow B$*.*

Much of the expressive power of intersection type disciplines comes from the fact that types can be endowed with a *preorder relation* $\leq$, which induces the structure of a meet semi-lattice with respect to $\cap$.

Definition 1.3 (Intersection type preorder) *Let* $\mathbf{T} = \mathbf{T}(\mathbf{C})$ *be an intersection type language. An* intersection type preorder *over* $\mathbf{T}$ *is a binary relation* $\leq$ *on* $\mathbf{T}$ *satisfying the following set* $\bigtriangledown^0$ *("nabla-zero") of axioms and rules:*

$$
\begin{array}{llll}
(\mathit{refl}) & A \leq A & (\mathit{idem}) & A \leq A \cap A \\[2ex]
(\mathit{incl}_L) & A \cap B \leq A & (\mathit{incl}_R) & A \cap B \leq B \\[1ex]
(\mathit{mon}) & \dfrac{A \leq A' \quad B \leq B'}{A \cap B \leq A' \cap B'} & (\mathit{trans}) & \dfrac{A \leq B \quad B \leq C}{A \leq C}
\end{array} \blacksquare
$$

Notation 1.4 *We will write* $A \sim B$ *for* $A \leq B$ *and* $B \leq A$.

Notice that associativity and commutativity of $\cap$ (modulo $\sim$) follow easily from the above axioms and rules.

Notation 1.5 *Since* $\cap$ *is commutative and associative, we will write* $\bigcap_{i \leq n} A_i$ *for* $A_1 \cap \ldots \cap A_n$. *Similarly we will write* $\bigcap_{i \in I} A_i$ *where we assume that* I *denotes always a finite non-empty set.*

Possibly effective, syntactical presentations of intersection type preorders can be given using the notion of *intersection type theory.* An intersection type theory includes always the basic set $\bigtriangledown^0$ for $\leq$ and possibly other special purpose axioms and rules.

Definition 1.6 (Intersection type theories) *Let* $\mathbf{T} = \mathbf{T}(\mathbf{C})$ *be an intersection type language, and let* $\bigtriangledown$ *be a collection of axioms and rules for deriving judgements of the shape* $A \leq B$, *with* $A, B \in \mathbf{T}$. *The* intersection type theory $\Sigma(\mathbf{C}, \bigtriangledown)$ *is the set of all judgements* $A \leq B$ *derivable from the axioms and rules in* $\bigtriangledown^0 \cup \bigtriangledown$. $\blacksquare$

Notation 1.7 *When we consider the intersection type theory* $\Sigma(\mathbf{C}, \bigtriangledown)$, *we will write*

$$
\begin{array}{ll}
\mathbf{C}^{\bigtriangledown} & \textit{for } \mathbf{C}, \\[1ex]
\mathbf{T}^{\bigtriangledown} & \textit{for } \mathbf{T}(\mathbf{C}), \\[1ex]
\Sigma^{\bigtriangledown} & \textit{for } \Sigma(\mathbf{C}, \bigtriangledown).
\end{array}
$$

Moreover $A \leq_{\bigtriangledown} B$ *will be short for* $(A \leq B) \in \Sigma^{\bigtriangledown}$. *Finally we will write* $A \sim_{\bigtriangledown} B$ *for* $A \leq_{\bigtriangledown} B \leq_{\bigtriangledown} A$.

In Figure 1 appears a list of special purpose axioms and rules which have been considered in the literature. We give just a few lines of motivation for each.

Axiom (Ω) states that the resulting type preorder has a maximal element. Axiom (Ω) is particularly meaningful when used in combination with the Ω-type assignment system, which essentially treats Ω as the universal type of all

(Ω)	$A \leq \Omega$
$(\Omega\text{-}\eta)$	$\Omega \leq \Omega {\to} \Omega$
$(\Omega\text{-}lazy)$	$A {\to} B \leq \Omega {\to} \Omega$
$({\to}\text{-}\cap)$	$(A {\to} B) \cap (A {\to} C) \leq A {\to} B \cap C$
(η)	$\dfrac{A' \leq A \qquad B \leq B'}{A {\to} B \leq A' {\to} B'}$
$(\omega\text{-}Scott)$	$\Omega {\to} \omega \sim \omega$
$(\omega\text{-}Park)$	$\omega {\to} \omega \sim \omega$
$(\omega\varphi)$	$\omega \leq \varphi$
$(\varphi {\to} \omega)$	$\varphi {\to} \omega \sim \omega$
$(\omega {\to} \varphi)$	$\omega {\to} \varphi \sim \varphi$

Fig. 1. Some special purpose Axioms and Rules concerning $\leq$.

λ-terms (see Definition 1.11).

The meaning of the axioms $(\Omega\text{-}\eta)$, $(\Omega\text{-}lazy)$, $({\to}\text{-}\cap)$ and of the rule (η) can be grasped easily if we consider the *set theoretic semantics* of intersection types. According to this semantics types are interpreted as subsets of the domain of discourse, $\cap$ is interpreted as set-theoretic intersection, $\leq$ is interpreted as set inclusion, $A {\to} B$ as the set of functions which map each element of A into an element of B.

For instance, in combination with Axiom (Ω), Axiom $(\Omega\text{-}\eta)$ expresses the fact that all the objects in our domain of discourse are total functions, i.e. that Ω is equal to $\Omega {\to} \Omega$ [5].

However, if we want to capture only those terms which truly represent functions, as is necessary, for instance, in discussing the lazy λ-calculus [2], we cannot assume axiom $(\Omega\text{-}\eta)$ in order to ensure that all functions are total. To this end we can postulate instead the weaker property $(\Omega\text{-}lazy)$. According to the set theoretic semantics, this axiom states, in effect, simply that an element which is a function, (since it maps A into B) maps also the whole universe into itself.

The set-theoretic meaning of Axiom $({\to}\text{-}\cap)$ is immediate: if a function maps

$\mathbf{C}^{\mathcal{B}a}$	$= \mathbf{C}_\infty$	$\mathcal{B}a$	$= \{(\to\text{-}\cap), (\eta)\}$	[4]
$\mathbf{C}^{\mathcal{AO}}$	$= \{\Omega\}$	$\mathcal{AO}$	$= \mathcal{B}a \cup \{(\Omega), (\Omega\text{-}lazy)\}$	[2]
$\mathbf{C}^{\mathcal{BCD}}$	$= \{\Omega\} \cup \mathbf{C}_\infty$	$\mathcal{BCD}$	$= \mathcal{B}a \cup \{(\Omega), (\Omega\text{-}\eta)\}$	[5]
$\mathbf{C}^{\mathcal{S}c}$	$= \{\Omega, \omega\}$	$\mathcal{S}c$	$= \mathcal{BCD} \cup \{(\omega\text{-}Scott)\}$	[26]
$\mathbf{C}^{\mathcal{P}a}$	$= \{\Omega, \omega\}$	$\mathcal{P}a$	$= \mathcal{BCD} \cup \{(\omega\text{-}Park)\}$	[23]
$\mathbf{C}^{\mathcal{CDZ}}$	$= \{\Omega, \varphi, \omega\}$	$\mathcal{CDZ}$	$= \mathcal{BCD} \cup \{(\omega\varphi), (\varphi{\to}\omega), (\omega{\to}\varphi)\}$	[11]
$\mathbf{C}^{\mathcal{DHM}}$	$= \{\Omega, \varphi, \omega\}$	$\mathcal{DHM}$	$= \mathcal{BCD} \cup \{(\omega\varphi), (\omega\text{-}Scott), (\omega{\to}\varphi)\}$	[15]

Fig. 2. Type Theories: atoms, axioms and rules.

A into B, and also A into C, then, actually, it maps the whole A into the intersection of B and C [5].

Rule (η) is also very natural set-theoretically: it asserts that the arrow constructor is contra-variant in the first argument and covariant in the second one. Namely, if a function maps A into B, and we take a subset A' of A and a superset B' of B, then this function will map also A' into B' [5].

The remaining axioms express peculiar properties of D_∞-like inverse limit models [12,11,19].

The element Ω plays a very special role in the development of the theory. Therefore we stipulate the following blanket assumption:

$$\text{if } \Omega \in \mathbf{C}^\triangledown \text{ then } (\Omega) \in \triangledown.$$

We introduce in Figure 2 a list of significant intersection type theories which have been extensively considered in the literature. We shall denote such theories as $\Sigma^\triangledown$ with various different names $\triangledown$, corresponding to the initial of the authors which have first considered the λ-model induced by such a theory [4,2,5,26,23,11,15]. For each such $\triangledown$ we specify in Figure 2 the type theory $\Sigma^\triangledown = \Sigma(\mathbf{C}, \triangledown)$ by giving the set of constants $\mathbf{C}^\triangledown$ and the set $\triangledown$ of extra axioms and rules taken from Figure 1. Here $\mathbf{C}_\infty$ is an infinite set of fresh atoms, i.e. different from Ω, φ, ω. The last column contains the reference to the paper where the λ-model induced by such a theory was defined.

Now that we have introduced intersection type theories we have to explain how to capitalise effectively on their expressive power. This is achieved via the crucial notion of *intersection type assignment system*. This is a natural extension of Curry's type assignment type to intersection types. First we need some preliminary definitions and notations.

Definition 1.8 *i) A* $\triangledown$-basis *is a set of statements of the shape* $x{:}B$*, where* $B \in \mathbf{T}^{\triangledown}$*, all whose variables are distinct.*

ii) We will write $x \in \Gamma$ *as short for* $\exists A\ x{:}A \in \Gamma$*, i.e.* x *occurs as the subject of an assertion in* Γ*.*

iii) If Γ, Γ' *are* $\triangledown$*-basis then* $\Gamma \uplus \Gamma'$ *is the* $\triangledown$*-basis defined by:*

$$\begin{aligned}\Gamma \uplus \Gamma' = &\{x{:}\, A \cap B \mid x{:}\, A \in \Gamma \text{ and } x{:}\, B \in \Gamma'\}\\ &\cup\{x{:}\, A \mid x{:}\, A \in \Gamma \text{ and } x \notin \Gamma'\}\\ &\cup\{x{:}\, B \mid x{:}\, B \in \Gamma' \text{ and } x \notin \Gamma\}.\end{aligned}$$

iv) An intersection type assignment system $\lambda\cap^{\triangledown}$ *relative to* $\Sigma^{\triangledown}$ *is a formal system for deriving judgements of the form* $\Gamma \vdash^{\triangledown} M : A$*, where the* subject M *is an untyped* λ*-term, the* predicate A *is in* $\mathbf{T}^{\triangledown}$*, and* Γ *is a* $\triangledown$*-basis.*

v) We say that a term M *is* typable *in* $\lambda\cap^{\triangledown}$*, for a given* $\triangledown$*-basis* Γ*, if there is a type* $A \in \mathbf{T}^{\triangledown}$ *such that the judgement* $\Gamma \vdash^{\triangledown} M : A$ *is derivable.* ■

As usual λ-terms are considered modulo α-conversion. We denote by V the set of term variables and by $FV(M)$ the set of free variables of the term M.

Definition 1.9 (Basic Type Assignment System) *Let* $\Sigma^{\triangledown}$ *be a type theory. The* basic type assignment system $\lambda\cap^{\triangledown}_{\mathcal{B}}$ *is a formal system for deriving judgements of the shape* $\Gamma \vdash^{\triangledown}_{\mathcal{B}} M : A$*. Its rules are the following:*

$$(\mathrm{Ax})\ \frac{x{:}A \in \Gamma}{\Gamma \vdash^{\triangledown}_{\mathcal{B}} x{:}A} \qquad (\to\mathrm{I})\ \frac{\Gamma, x{:}A \vdash^{\triangledown}_{\mathcal{B}} M : B}{\Gamma \vdash^{\triangledown}_{\mathcal{B}} \lambda x.M : A\to B}$$

$$(\to\mathrm{E})\ \frac{\Gamma \vdash^{\triangledown}_{\mathcal{B}} M : A \to B \quad \Gamma \vdash^{\triangledown}_{\mathcal{B}} N : A}{\Gamma \vdash^{\triangledown}_{\mathcal{B}} MN : B} \qquad (\cap\mathrm{I})\ \frac{\Gamma \vdash^{\triangledown}_{\mathcal{B}} M : A \quad \Gamma \vdash^{\triangledown}_{\mathcal{B}} M : B}{\Gamma \vdash^{\triangledown}_{\mathcal{B}} M : A \cap B}$$

$$(\leq_{\triangledown})\ \frac{\Gamma \vdash^{\triangledown}_{\mathcal{B}} M : A \quad A \leq_{\triangledown} B}{\Gamma \vdash^{\triangledown}_{\mathcal{B}} M : B}\ ■$$

Example 1.10 *Self-application can be easily typed in* $\lambda\cap^{\triangledown}_{\mathcal{B}}$*, as follows.*

$$\dfrac{\dfrac{\dfrac{x{:}(A\to B)\cap A \vdash^{\triangledown}_{\mathcal{B}} x{:}(A\to B)\cap A}{x{:}(A\to B)\cap A \vdash^{\triangledown}_{\mathcal{B}} x{:}A\to B}(\leq_{\triangledown}) \quad \dfrac{x{:}(A\to B)\cap A \vdash^{\triangledown}_{\mathcal{B}} x{:}(A\to B)\cap A}{x{:}(A\to B)\cap A \vdash^{\triangledown}_{\mathcal{B}} x{:}A}(\leq_{\triangledown})}{x{:}(A\to B)\cap A \vdash^{\triangledown}_{\mathcal{B}} xx : B}(\to\mathrm{E})}{\vdash^{\triangledown}_{\mathcal{B}} \lambda x.xx : (A\to B)\cap A\to B}(\to\mathrm{I})$$

If $\Omega \in \mathbf{C}^\triangledown$, in line with the intended set-theoretic interpretation of Ω as the universe, we extend the Basic Type Assignment System with a suitable axiom for Ω:

Definition 1.11 (Ω-type Assignment System) *Let $\Sigma^\triangledown$ be a type theory with $\Omega \in \mathbf{C}^\triangledown$. The axioms and rules of the* Ω-type assignment system $\lambda\cap^\triangledown_\Omega$ *are those of the Basic type Assignment System, together with the further axiom:*

$$(\text{Ax-}\Omega) \quad \Gamma \vdash^\triangledown_\Omega M : \Omega. \ \blacksquare$$

Example 1.12 *Also non-strongly normalising terms can be typed in $\lambda\cap^\triangledown_\Omega$ even with a type $A \not\leq_\triangledown \Omega$. Note the usage of the axiom* (Ax-Ω)*. Let $\Delta \equiv \lambda x.xx$.*

$$\dfrac{\dfrac{\dfrac{y{:}\Omega, x{:}A \vdash^\triangledown_\Omega x{:}A}{y{:}\Omega \vdash^\triangledown_\Omega \lambda x.x : A\to A}(\to\text{I})}{\vdash^\triangledown_\Omega \lambda yx.x : \Omega\to A\to A}(\to\text{I}) \qquad \vdash^\triangledown_\Omega \Delta\Delta : \Omega}{\vdash^\triangledown_\Omega (\lambda yx.x)(\Delta\Delta) : A\to A}(\to\text{E})$$

An interesting example is that the Fixed-point Combinator $\mathbf{Y} \equiv \lambda f.(\lambda x.f(xx))(\lambda x.f(xx))$ can be typed in $\lambda\cap^\triangledown_\Omega$ as follows.

$$\dfrac{\dfrac{\dfrac{\dfrac{f{:}\Omega \to A, x{:}\Omega \vdash^\triangledown_\Omega f{:}\Omega \to A \quad f{:}\Omega \to A, x{:}\Omega \vdash^\triangledown_\Omega xx : \Omega}{f{:}\Omega \to A, x{:}\Omega \vdash^\triangledown_\Omega f(xx) : A}(\to\text{E})}{f{:}\Omega \to A \vdash^\triangledown_\Omega \lambda x.f(xx) : \Omega \to A}(\to\text{I}) \qquad f{:}\Omega \to A \vdash^\triangledown_\Omega \lambda x.f(xx) : \Omega}{f{:}\Omega \to A \vdash^\triangledown_\Omega (\lambda x.f(xx))(\lambda x.f(xx)) : A}(\to\text{E})}{\vdash^\triangledown_\Omega \lambda f.(\lambda x.f(xx))(\lambda x.f(xx)) : (\Omega \to A) \to A}(\to\text{I})$$

For ease of notation, we assume that the symbol Ω is reserved for the type constant used in the system $\lambda\cap^\triangledown_\Omega$, and hence we forbid $\Omega \in \mathbf{C}^\triangledown$ when we deal with $\lambda\cap^\triangledown_\mathcal{B}$.

Notation 1.13 *In the following $\lambda\cap^\triangledown$ will range over $\lambda\cap^\triangledown_\mathcal{B}$ and $\lambda\cap^\triangledown_\Omega$. More precisely we convene that $\lambda\cap^\triangledown$ stands for $\lambda\cap^\triangledown_\Omega$ whenever $\Omega \in \mathbf{C}^\triangledown$, and for $\lambda\cap^\triangledown_\mathcal{B}$ otherwise. Similarly for $\vdash^\triangledown$.*

We refer to [7] for a detailed account on the interest and differences of the two intersection type assignment systems introduced above.

Notice that the structural rules of (weakening) and (strengthening) are admissible in all $\lambda\cap^\triangledown$s:

$$(\text{weakening}) \ \dfrac{\Gamma \vdash^\triangledown M : A}{\Gamma, x{:}B \vdash^\triangledown M : A} \quad (\text{strengthening}) \ \dfrac{\Gamma \vdash^\triangledown M : A}{\Gamma \lceil M \vdash^\triangledown M : B}$$

where $\Gamma\lceil M = \{x{:}B \mid x \in FV(M)\}$.

Another admissible rule allowing us to strengthen the premises is the following:

$$(\leq_{\nabla}\mathrm{L})\ \frac{\Gamma, x{:}B \vdash^{\nabla} M : A \qquad C \leq_{\nabla} B}{\Gamma, x{:}C \vdash^{\nabla} M : A}$$

Lastly notice also that the intersection elimination rules

$$(\cap\mathrm{E})\ \frac{\Gamma \vdash^{\nabla} M : A \cap B}{\Gamma \vdash^{\nabla} M : A} \qquad \frac{\Gamma \vdash^{\nabla} M : A \cap B}{\Gamma \vdash^{\nabla} M : B}$$

can immediately be proved to be derivable in all $\lambda\cap^{\nabla}$'s using $(\leq_{\nabla})$.

We prove now a crucial technical result concerning intersection-type theories. It is a form of generation (or inversion) lemma, which provides conditions for "reversing" some of the rules of the type assignment systems $\lambda\cap^{\nabla}$.

Notation 1.14 *When we write "...assume $A \not\sim_{\nabla} \Omega$..." we mean that this condition is always true when we deal with $\vdash^{\nabla}_{\mathcal{B}}$, while it must be checked for $\vdash^{\nabla}_{\Omega}$.*

Theorem 1.15 (Generation Lemma) *Let Σ^{∇} be a type theory.*

i) Assume $A \not\sim_{\nabla} \Omega$. Then $\Gamma \vdash^{\nabla} x{:}A$ if and only if $x{:}B \in \Gamma$ and $B \leq_{\nabla} A$ for some $B \in \mathbf{T}^{\nabla}$.

ii) Assume $A \not\sim_{\nabla} \Omega$. Then $\Gamma \vdash^{\nabla} MN : A$ if and only if $\Gamma \vdash^{\nabla} M : B_i \to C_i$, $\Gamma \vdash^{\nabla} N : B_i$, and $\bigcap_{i \in I} C_i \leq_{\nabla} A$ for some non-empty set I and $B_i, C_i \in \mathbf{T}^{\nabla}$.

iii) $\Gamma \vdash^{\nabla} \lambda x.M : A$ if and only if $\Gamma, x{:}B_i \vdash^{\nabla} M : C_i$, and $\bigcap_{i \in I}(B_i \to C_i) \leq_{\nabla} A$ for some non-empty set I and $B_i, C_i \in \mathbf{T}^{\nabla}$.

Proof. The proof of each $(\Leftarrow)$ is easy. So we only treat $(\Rightarrow)$.

i) Easy by induction on derivations, since only the axioms (Ax), (Ax-Ω), and the rules $(\cap\mathrm{I})$, $(\leq_{\nabla})$ can be applied. Notice that the condition $A \not\sim_{\nabla} \Omega$ implies that $\Gamma \vdash^{\nabla} x : A$ cannot be obtained just using axiom (Ax-Ω).

ii) By induction on derivations. The only interesting case is when $A \equiv A_1 \cap A_2$ and the last rule applied is $(\cap\mathrm{I})$:

$$(\cap\mathrm{I})\ \frac{\Gamma \vdash^{\nabla} MN : A_1 \quad \Gamma \vdash^{\nabla} MN : A_2}{\Gamma \vdash^{\nabla} MN : A_1 \cap A_2}.$$

The condition $A \not\sim_{\nabla} \Omega$ implies that we cannot have $A_1 \sim_{\nabla} A_2 \sim_{\nabla} \Omega$. We do the proof for $A_1 \not\sim_{\nabla} \Omega$ and $A_2 \not\sim_{\nabla} \Omega$, the other cases can be treated similarly. By induction there are I, B_i, C_i, J, D_j, E_j such that

$$\forall i \in I.\ \Gamma \vdash^{\nabla} M : B_i \to C_i, \Gamma \vdash^{\nabla} N : B_i,$$
$$\forall j \in J.\ \Gamma \vdash^{\nabla} M : D_j \to E_j, \Gamma \vdash^{\nabla} N : D_j,$$

and moreover $\bigcap_{i\in I} C_i \leq_{\nabla} A_1$, $\bigcap_{j\in J} E_j \leq_{\nabla} A_2$. So we are done since $(\bigcap_{i\in I} C_i) \cap (\bigcap_{j\in J} E_j) \leq_{\nabla} A$.

iii) If $A \sim_{\nabla} \Omega$ we can choose $B \equiv C \equiv \Omega$. Otherwise the proof is by induction on derivations. Notice that $\Gamma \vdash^{\nabla} \lambda x.M : A$ cannot be obtained just using axiom (Ax-Ω). The only interesting case is again when $A \equiv A_1 \cap A_2$ and the last rule applied is ($\cap$I):

$$(\cap\mathrm{I})\ \frac{\Gamma \vdash^{\nabla} \lambda x.M : A_1 \quad \Gamma \vdash^{\nabla} \lambda x.M : A_2}{\Gamma \vdash^{\nabla} \lambda x.M : A_1 \cap A_2}.$$

As in the proof of (ii) we only consider the case $A_1 \not\sim_{\nabla} \Omega$, and $A_2 \not\sim_{\nabla} \Omega$ .By induction there are I, B_i, C_i, J, D_j, E_j such that

$$\forall i \in I.\ \Gamma, x{:}B_i \vdash^{\nabla} M : C_i, \quad \forall j \in J.\ \Gamma, x{:}D_j \vdash^{\nabla} M : E_j,$$
$$\bigcap_{i\in I}(B_i \to C_i) \leq_{\nabla} A_1 \ \ \&\ \ \bigcap_{j\in J}(D_j \to E_j) \leq_{\nabla} A_2.$$

So we are done since $(\bigcap_{i\in I}(B_i \to C_i)) \cap (\bigcap_{j\in J}(D_j \to E_j)) \leq_{\nabla} A$. ■

Special cases of this theorem have already appeared in the literature [5,12,11,19].

We conclude this section by characterising those type assignment systems for which types are preserved under β-expansion, i.e. those systems for which the following rule is admissible:

$$(\beta\text{-}exp) \quad \frac{M \longrightarrow_{\beta} N \qquad \Gamma \vdash^{\nabla} N : A}{\Gamma \vdash^{\nabla} M : A}$$

It will be convenient to consider also the rule of $\beta\mathbf{I}$-expansion, denoted by ($\beta\mathbf{I}$-*exp*), which amounts to the restriction of Rule (β-*exp*) to the case where $M \longrightarrow_{\beta} N$ is obtained by contracting only λ-$\mathbf{I}$-redexes. We recall that $(\lambda x.M)N$ is a λ-$\mathbf{I}$-redex if $x \in FV(M)$.

Theorem 1.16 (Characterization of β-expansion)

i) Rule ($\beta\mathbf{I}$-exp) is admissible in $\lambda\cap^{\nabla}$ for all theories Σ^{∇}.

ii) Rule (β-exp) is admissible in $\lambda\cap^{\nabla}_{\Omega}$ for all theories Σ^{∇}.

iii) Rule (β-exp) is not admissible in $\lambda\cap^{\nabla}_{\mathcal{B}}$ for any theory Σ^{∇}.

Proof.

i) We will only show in detail that if $\Gamma \vdash^{\nabla} M[x := N] : A$ then $\Gamma \vdash^{\nabla} (\lambda x.M)N : A$. Then by a straightforward, double induction on $\longrightarrow_\beta$ and on derivations we get the result. So assume that D is a derivation of $\Gamma \vdash^{\nabla} M[x := N] : A$. Let $\Gamma_i \vdash^{\nabla} N : B_i$ for $i \in I$ be all the statements in D whose subject is N. Without loss of generality we can assume that x does not occur in Γ. Since $x \in FV(M)$, I is non empty, hence we have that $\Gamma \subseteq \Gamma_i$ but $\Gamma \restriction FV(N) = \Gamma_i \restriction FV(N)$. So using rules (strengthening) and ($\cap$I), we have that $\Gamma \vdash^{\nabla} N : \bigcap_{i\in I} B_i$. Moreover, one can easily see, by induction on M, that $\Gamma, x : \bigcap_{i\in I} B_i \vdash^{\nabla} M : A$. Thus, by rule ($\to$I), we have $\Gamma \vdash^{\nabla} \lambda x.M : \bigcap_{i\in I} B_i \to A$. Hence, by ($\to$E) we can get $\Gamma \vdash^{\nabla} (\lambda x.M)N : A$.

ii) The proof proceeds as above except for the fact that we have to consider also the case that $x \notin FV(M)$. Using first (weakening) and then ($\to$I) we get $\Gamma \vdash^{\nabla}_{\Omega} \lambda x.M : \Omega \to A$, then using (Ax-$\Omega$) and ($\to$E) we finally get $\Gamma \vdash^{\nabla}_{\Omega} (\lambda x M)N : A$.

iii) Recall that the Generation Lemma implies that in a theory where $\Omega \notin \mathbf{C}^{\nabla}$, a term with a free variable is typable if and only if that variable occurs in the context. So for any theory ∇, such that $\Omega \notin \mathbf{C}^{\nabla}$, $(\lambda yx.x)z$ is not typable from the empty context in $\lambda\cap^{\nabla}_{\mathcal{B}}$, but clearly $\vdash^{\nabla}_{\mathcal{B}} \lambda x.x : A \to A$ for all types A. ∎

2 Some distinguished properties of λ-terms

In this section we introduce the distinguished classes of λ-terms which we shall focus on in this paper.

We shall consider first *termination properties*. In particular we shall discuss the crucial property of being *strongly normalising* and the three properties of having a *β-normal form*, of having a *head normal form*, and of having a *weak head normal form*.

Definition 2.1 (Normalization property)

i) M is a normal form*, $M \in \mathsf{NF}$, if M cannot be further reduced;*

ii) M is strongly normalising*, $M \in \mathsf{SN}$, if all reductions starting at M are finite;*

iii) M has a normal form*, $M \in \mathsf{N}$, if M reduces to a normal form;*

iv) M has a head normal form*, $M \in \mathsf{HN}$, if M reduces to a term of the form $\lambda\vec{x}.y\vec{M}$ (where possibly y appears in $\vec{x}$);*

v) M has a weak head normal form, $M \in \mathsf{WN}$, *if* M *reduces to an abstraction or to a term starting with a free variable.* ■

For each of the above properties, but SN, in the above definition, we shall consider also the corresponding *persistent* version (see Definition 2.2). *Persistently normalising* terms have been introduced in [9].

Definition 2.2 (Persistent normalisation property)

i) *A term* M *is* persistently normalising, $M \in \mathsf{PN}$, *if* $M\vec{N} \in \mathsf{N}$ *for all terms* $\vec{N}$ *in* N.

ii) *A term* M *is a* persistently normalising normal form, $M \in \mathsf{PNF}$, *if it is both persistently normalising and it is a normal form.*

iii) *A term* M *is* persistently head normalising, $M \in \mathsf{PHN}$, *if* $M\vec{N} \in \mathsf{HN}$ *for all terms* $\vec{N}$.

iv) *A term* M *is* persistently weak normalising, $M \in \mathsf{PWN}$, *if* $M\vec{N} \in \mathsf{WN}$ *for all terms* $\vec{N}$. ■

Example 2.3 *Let* $\mathbf{I} \equiv \lambda x.x,\ \Delta \equiv \lambda x.xx,\ \mathbf{Y} \equiv \lambda f.(\lambda x.f(xx))(\lambda x.f(xx)),\ \mathbf{K} \equiv \lambda xy.x$.

- $\lambda x.yx \in \mathsf{PNF}$.
- $\lambda x.x\Delta\Delta \in \mathsf{NF}$, *but* $\lambda x.x\Delta\Delta \notin \mathsf{PNF}$, *since* $(\lambda x.x\Delta\Delta)\mathbf{I} \longrightarrow\!\!\!\!\!\rightarrow_\beta \Delta\Delta \notin \mathsf{N}$.
- $\mathbf{II} \in \mathsf{SN}$, *but* $\mathbf{II} \notin \mathsf{NF}$ *and* $\mathbf{II} \notin \mathsf{PN}$, *since* $\mathbf{II}\Delta\Delta \longrightarrow\!\!\!\!\!\rightarrow_\beta \Delta\Delta \notin \mathsf{N}$.
- $(\lambda x.y)(\Delta\Delta) \in \mathsf{PN}$, *but* $(\lambda x.y)(\Delta\Delta) \notin \mathsf{PNF}$ *and* $(\lambda x.y)(\Delta\Delta) \notin \mathsf{SN}$.
- $\lambda y.(\lambda x.y\Delta)(\Delta\Delta) \in \mathsf{N}$, *but* $\lambda y.(\lambda x.y\Delta)(\Delta\Delta) \notin \mathsf{SN}$ *and* $\lambda y.(\lambda x.y\Delta)(\Delta\Delta) \notin \mathsf{PN}$, *since* $(\lambda y.(\lambda x.y\Delta)(\Delta\Delta))\Delta \longrightarrow\!\!\!\!\!\rightarrow_\beta \Delta\Delta \notin \mathsf{N}$.
- $\lambda x.y(\Delta\Delta) \in \mathsf{PHN}$, *but* $\lambda x.y(\Delta\Delta) \notin \mathsf{N}$.
- $\lambda x.x(\Delta\Delta) \in \mathsf{HN}$, *but* $\lambda x.x(\Delta\Delta) \notin \mathsf{N}$ *and* $\lambda x.x(\Delta\Delta) \notin \mathsf{PHN}$, *since* $(\lambda x.x(\Delta\Delta))\Delta \longrightarrow_\beta \Delta(\Delta\Delta) \notin \mathsf{HN}$.
- $\mathbf{YK} \in \mathsf{PWN}$, *but* $\mathbf{YK} \notin \mathsf{HN}$.
- $\lambda x.\Delta\Delta \in \mathsf{WN}$, *but* $\lambda x.\Delta\Delta \notin \mathsf{HN}$ *and* $\lambda x.\Delta\Delta \notin \mathsf{PWN}$, *since* $(\lambda x.\Delta\Delta)M \longrightarrow_\beta \Delta\Delta \notin \mathsf{WN}$.

The following proposition, represented pictorially by Figure 3, illustrates mutual implications between the above notions:

Proposition 2.4 *The following* strict *inclusions hold:*

$$\begin{gathered}\mathsf{PNF} \subsetneq \mathsf{NF} \subsetneq \mathsf{SN} \subsetneq \mathsf{N}\\ \mathsf{PNF} \subsetneq \mathsf{PN} \subsetneq \mathsf{N} \subsetneq \mathsf{HN}\\ \mathsf{PN} \subsetneq \mathsf{PHN} \subsetneq \mathsf{HN} \subsetneq \mathsf{WN}\\ \mathsf{PHN} \subsetneq \mathsf{PWN} \subsetneq \mathsf{WN}.\end{gathered}$$

No other inclusion holds between the above sets. ■

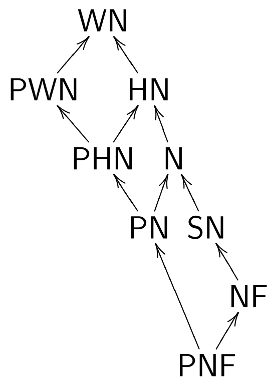

Fig. 3. Inclusion between sets of λ-terms

The following characterisation of strongly normalising terms will be very useful in the sequel.

Proposition 2.5 ([27,18]) *The set* SN *is the least set of terms closed under the following rules:*

$$\frac{M_1 \in \mathsf{SN}, \ldots, M_n \in \mathsf{SN}, (n \geq 0)}{x M_1 \ldots M_n \in \mathsf{SN}} \qquad \frac{M \in \mathsf{SN}}{\lambda x.M \in \mathsf{SN}}$$

$$\frac{M[x := N] M_1 \ldots M_n \in \mathsf{SN} \quad N \in \mathsf{SN}, (n \geq 0)}{(\lambda x.M) N M_1 \ldots M_n \in \mathsf{SN}} \ \blacksquare$$

The proof of the above proposition follows by suitable inductions.

Intersection types can be used to characterise *compositionally* also other evaluation properties of terms, which are not linked to termination. In this paper we shall consider, by way of example, the property of *reducing to a closed term.* Hence we conclude this section with the definition of:

Definition 2.6 (Closable term) *M is* closable*, $M \in \mathsf{C}$, if M reduces to a closed term.* $\blacksquare$

3 Characterising compositionally properties of λ-terms

In this section we put to use *intersection type disciplines* to give a *compositional* characterisation of evaluation properties of λ-terms. In view of Theorem 1.16(i) we can only characterise properties which are closed under, at least, $\beta\mathbf{I}$-expansion, hence we will not be able to characterise NF and PNF.

In this section we give the main result of the paper, Theorem 3.2. For each of the properties introduced in Section 2, Theorem 3.2 provides a *compositional* characterisations in terms of intersection type assignment systems. Soundness of these characterisations will be proved in the present section (and in the Appendix) and completeness will be proved in Section 4.

Some of the properties characterised in Theorem 3.2 had received already characterisations in terms of intersection type disciplines. The most significant case is that of strongly normalising terms. One of the original motivations for introducing intersection types in [25] was precisely that of achieving such a characterisation. Alternative characterisations appear in [21,4,20,17,3,18]. In [11] both normalising and persistently normalising terms had been characterised using intersection types. The type assignment system in [11] has also been discussed in [8]. Closed terms were characterised in [19]. The characterisations appearing in Theorem 3.2 strengthen and generalise all earlier results, since all previous papers consider only specific type theories, and hence in our view Theorem 3.2 appears more intrinsic.

Before giving the main theorem a last definition is necessary.

Definition 3.1 *i) A type theory* $\Sigma^\bigtriangledown$ *is an* arrow-type theory *if* $\Omega \in \mathbf{C}^\bigtriangledown$*, the axioms of* $\mathcal{B}a$ *are admissible in* $\Sigma^\bigtriangledown$ *and* $\forall \psi \in \mathbf{C}^\bigtriangledown\ \exists I, \{A_i, B_i\}_{i\in I}.\psi {\sim_\bigtriangledown} \bigcap_{i\in I}(A_i{\to}B_i)$.

ii) A type A *contains a type* B *(notation* $B \in A$*) if and only if* $A \equiv C[B]$ *for some context* $C[\,]$.

iii) A basis Γ *contains a type* A *(notation* $A \in \Gamma$*) if and only if there is* $x{:}B \in \Gamma$ *such that* $A \in B$.

iv) A type A *contains a type* B *modulo* $\sim_\bigtriangledown$ *(notation* $B \in_\bigtriangledown A$*) if and only if there is* $A' \sim_\bigtriangledown A$ *such that* $B \in A'$.

v) A basis Γ *contains a type* A *modulo* $\sim_\bigtriangledown$ *(notation* $A \in_\bigtriangledown \Gamma$*) if and only if there is* $x{:}B \in \Gamma$ *such that* $A \in_\bigtriangledown B$. ■

The theories $\Sigma^{\mathcal{S}c}$, $\Sigma^{\mathcal{P}a}$, $\Sigma^{\mathcal{CDZ}}$ and $\Sigma^{\mathcal{DHM}}$ of Figure 2 are arrow-type theories. For example, $\Omega \notin \omega$ but $\Omega \in_{\mathcal{S}c} \omega$ since $\omega \sim_{\mathcal{S}c} \Omega \to \omega$ and $\Omega \in \Omega \to \omega$.

Finally we can state the main result:

Theorem 3.2 (Characterization)

1 Normalisation properties

i) (strongly normalising terms) A λ*-term* $M \in \mathsf{SN}$ *if and only if for all type theories* $\Sigma^\bigtriangledown$ *there exist* $A \in \mathbf{T}^\bigtriangledown$ *and a* $\bigtriangledown$*-basis* Γ *such that* $\Gamma \vdash^\bigtriangledown M : A$*. Moreover in the system* $\lambda\cap^{\mathcal{B}a}_{\mathcal{B}}$ *the terms satisfying the latter property are precisely the strongly normalising ones.*

ii) (normalising terms) A λ*-term* $M \in \mathsf{N}$ *if and only if for all type theories*

Σ^∇ *such that* $\{\Omega\} \subset \mathbf{C}^\nabla$, [1] *there exist* $A \in \mathbf{T}^\nabla$ *and a* ∇*-basis* Γ *such that* $\Gamma \vdash^\nabla_\Omega M : A$ *and* $\Omega \notin A, \Gamma$. *Moreover in the system* $\lambda\cap^{\mathcal{BCD}}_\Omega$ *the terms satisfying the latter property are precisely the ones which have a normal form. Furthermore, in the system* $\lambda\cap^{\mathcal{CDZ}}_\Omega$ *the terms typable with type* φ *in the* $\mathcal{CDZ}$*-basis all of whose predicates are* ω*, are precisely the ones which have a normal form.*

iii) *(head normalising terms) A* λ*-term* $M \in \mathsf{HN}$ *if and only if for all type theories* Σ^∇ *such that* $\Omega \in \mathbf{C}^\nabla$, *and for all* $A \in \mathbf{T}^\nabla$ *there exist a* ∇*-basis* Γ *and two integers* m, n *such that* $\Gamma \vdash^\nabla_\Omega M : (\Omega^m \to A)^n \to A$. *Moreover in the system* $\lambda\cap^{\mathcal{BCD}}_\Omega$ *the terms satisfying the latter property are precisely the ones which have a head normal form. Furthermore, in the system* $\lambda\cap^{\mathcal{DHM}}_\Omega$ *the terms typable with type* φ *in the* $\mathcal{DHM}$*-basis all of whose predicates are* ω*, are precisely the ones which have a head normal form.*

iv) *(weak head normalising terms) A* λ*-term* $M \in \mathsf{WN}$ *if and only if for all type theories* Σ^∇ *such that* $\Omega \in \mathbf{C}^\nabla$, *there exists a* ∇*-basis* Γ *such that* $\Gamma \vdash^\nabla_\Omega M : \Omega \to \Omega$. *Moreover in the system* $\lambda\cap^{\mathcal{AO}}_\Omega$ *the terms satisfying the latter property are precisely the ones which have a weak head normal form.*

2 Persistent normalisation properties

i) *(persistently normalising terms) A* λ*-term* $M \in \mathsf{PN}$ *if and only if for all arrow-type theories* Σ^∇ *and all* $A \in \mathbf{T}^\nabla$ *with* $\Omega \notin_\nabla A$ *there exists a* ∇*-basis* Γ *such that* $\Omega \notin_\nabla \Gamma$ *and* $\Gamma \vdash^\nabla_\Omega M : A$. *Moreover in the system* $\lambda\cap^{\mathcal{CDZ}}_\Omega$ *the terms typable with type* ω *in the* $\mathcal{CDZ}$*-basis all of whose predicates are* ω *are precisely the persistently normalising ones.*

ii) *(persistently head normalising terms) A* λ*-term* $M \in \mathsf{PHN}$ *if and only if for all type theories* Σ^∇ *such that* $\Omega \in \mathbf{C}^\nabla$ *and all* $A \in \mathbf{T}^\nabla$ *there exists a* ∇*-basis* Γ *and an integer* n *such that* $\Gamma \vdash^\nabla_\Omega M : \Omega^n \to A$. *Moreover in the systems* $\lambda\cap^{\mathcal{SC}}_\Omega$ *and* $\lambda\cap^{\mathcal{DHM}}_\Omega$ *the terms typable with type* ω *in the basis all of whose predicates are* ω*, are precisely the persistently head normalising ones.*

iii) *(persistently weak normalising terms) A* λ*-term* $M \in \mathsf{PWN}$ *if and only if for all type theories* Σ^∇ *such that* $\Omega \in \mathbf{C}^\nabla$ *and all integers* n *there exists a* ∇*-basis* Γ *such that* $\Gamma \vdash^\nabla_\Omega M : \Omega^n \to \Omega$. *Moreover in the system* $\lambda\cap^{\mathcal{AO}}_\Omega$ *the terms satisfying the latter property are precisely the persistently weak normalising ones.*

3 Closability *(closed terms) A* λ*-term* $M \in \mathsf{C}$ *if and only if for all type theories* Σ^∇ *such that* $\Omega \in \mathbf{C}^\nabla$ *and* $\omega \sim_\nabla \omega \to \omega$ *for some* $\omega \in \mathbf{C}^\nabla$, M *is typable with type* ω*, for the empty* ∇*-basis. Moreover in the system* $\lambda\cap^{\mathcal{P}a}_\Omega$ *the terms satisfying the latter property are precisely the terms which reduce to closed terms.* ∎

[1] The condition $\{\Omega\} \subset \mathbf{C}^\nabla$ says that $\mathbf{C}^\nabla$ contains Ω and at least one other constant.

The proofs of the *only if* parts of the Theorem are mainly straightforward inductions and case split, and follow, but the case of persistently normalising terms (2.i), which is proved in the Appendix. The syntactic characterisation of the persistently normalising normal forms is quite technical. Our proof essentially follows the line of [11], but here we completely develop arguments that there were only sketched.

The proofs of the *if* parts require the set-theoretic semantics of intersection types using stable sets [7,14], which is developed in Section 4.

Proof of ($\Rightarrow$).

(1.iv) By Theorem 1.16(ii) it suffices to consider M in weak head normal form. If $M \equiv \lambda x.N$ then we get $\vdash_\Omega^\nabla N : \Omega$ by (Ax-Ω) and $\vdash_\Omega^\nabla M : \Omega \to \Omega$ by rule ($\to$I). If $M \equiv x\vec{M}$, where m is the length of $\vec{M}$, we derive $x : \Omega^{m+1} \to \Omega \vdash_\Omega^\nabla M : \Omega \to \Omega$ using (Ax-Ω) and ($\to$E).

(1.iii) Again by Theorem 1.16(ii) it suffices to consider M in head normal form. Let $M \equiv \lambda\vec{y}.x\vec{M}$ where $\vec{y}$ has length n and $\vec{M}$ has length m. We have $x{:}\Omega^m{\to}A \vdash_\Omega^\nabla x\vec{M} : A$ using rule ($\to$E). By rule ($\to$I) this implies $x{:}\Omega^m{\to}A \vdash_\Omega^\nabla M : (\Omega^m{\to}A)^n{\to}A$. For $\lambda\cap_\Omega^{\mathcal{DHM}}$ by choosing $A \equiv \omega$ we get from above $x{:}\Omega^m \to \omega \vdash_\Omega^{\mathcal{DHM}} M : (\Omega^m \to \omega)^n \to \omega$. By rules ($\leq_{\mathcal{DHM}}$) and ($\leq_{\mathcal{DHM}}$ L) this implies $x{:}\omega \vdash_\Omega^{\mathcal{DHM}} M : \varphi$ since $\omega \sim_{\mathcal{DHM}} \Omega \to \omega$, $\omega \leq_{\mathcal{DHM}} \varphi$ and $\varphi \sim_{\mathcal{DHM}} \omega \to \varphi$.

(1.ii) Similarly, it's sufficient to consider M in normal form. The proof is by induction on M. The only interesting case is $M \equiv x\vec{M}$ where $\vec{M} \equiv M_1 \ldots M_m$. By induction we have $\Gamma_j \vdash_\Omega^\nabla M_j : A_j$, for some Γ_j, A_j not containing Ω and for $j \leq m$. This implies: $\uplus_{j\leq m}\Gamma_j \uplus \{x{:}A_1{\to}\ldots{\to}A_m{\to}A\} \vdash_\Omega^\nabla x\vec{M} : A$, where A is an arbitrary type not containing Ω.
For $\lambda\cap_\Omega^{\mathcal{CDZ}}$ let $\Gamma = \{x{:}\omega \mid x \in \mathrm{FV}(M)\}$. If $M \equiv x\vec{M}$ then by induction we have $\Gamma \vdash_\Omega^{\mathcal{CDZ}} M_j : \varphi$ and this implies $\Gamma \vdash_\Omega^{\mathcal{CDZ}} x\vec{M} : \omega$, since $\omega \sim_{\mathcal{CDZ}} \varphi \to \omega$. By rule ($\leq_{\mathcal{CDZ}}$) we conclude $\Gamma \vdash_\Omega^{\mathcal{CDZ}} M : \varphi$. If $M \equiv \lambda y.N$ then by induction we have $\Gamma, y : \omega \vdash_\Omega^{\mathcal{CDZ}} N : \varphi$ and this implies $\Gamma \vdash_\Omega^{\mathcal{CDZ}} M : \omega \to \varphi$. By rule ($\leq_{\mathcal{CDZ}}$) we conclude $\Gamma \vdash_\Omega^{\mathcal{CDZ}} M : \varphi$.

(1.i) By induction on the structure of strongly normalising terms (see Proposition 2.5). The only interesting case is $M \equiv (\lambda x.R)N\vec{M}$ where m is the length of $\vec{M}$ and both $R[x := N]\vec{M}$ and N are strongly normalising. By induction hypothesis there are Γ, A, Γ', B such that $\Gamma \vdash^\nabla R[x := N]\vec{M} : A$ and $\Gamma' \vdash^\nabla N : B$. We get $\Gamma \uplus \Gamma' \vdash^\nabla R[x := N]\vec{M} : A$ and $\Gamma \uplus \Gamma' \vdash^\nabla N : B$, so if $m = 0$ we are done by a proof similar to that of Theorem 1.16(i). If $m > 0$ by iterated applications of Generation Lemma

1.15(ii) to $\Gamma \vdash^\nabla R[x := N]\vec{M} : A$ we have

$$\Gamma \vdash^\nabla R[x := N] : B_1^{(i)} \to \ldots \to B_m^{(i)} \to B^{(i)}, \quad \Gamma \vdash^\nabla M_j : B_j^{(i)}, \quad (j \leq m)$$

and $\bigcap_{i \in I} B^{(i)} \leq_\nabla A$ for some $I, B_j^{(i)} (j \leq m), B^{(i)} \in \mathbf{T}^\nabla$. As in case $m = 0$ we obtain $\Gamma \uplus \Gamma' \vdash^\nabla (\lambda x.R)N : B_1^{(i)} \to \ldots \to B_m^{(i)} \to B^{(i)}$. So we can conclude $\Gamma \uplus \Gamma' \vdash^\nabla (\lambda x.R)N\vec{M} : A$.

(2.iii) If M is persistently weak head normalising then either M is an unsolvable term of order ∞ (as defined in [2]), i.e. for all n there is N such that $M =_\beta \lambda x_1 \ldots x_n.N$, or M is a solvable term such that the head variable of its head normal form is free. In fact if M is an unsolvable term of a finite order, i.e. $M =_\beta \lambda x_1 \ldots x_n.N$ where N is unsolvable and it does not reduce to an abstraction, then $M\vec{N} \notin \mathsf{WN}$ where $\vec{N}$ are n arbitrary λ-terms. If $M =_\beta \lambda \vec{x} y \vec{z}.y\vec{N}$ we get $M\vec{x}(\Delta\Delta)\vec{z} \twoheadrightarrow_\beta \Delta\Delta\vec{N'} \notin \mathsf{WN}$, where $\vec{N'} = \vec{N}[y := \Delta\Delta]$.
If M is an unsolvable term of order ∞, i.e. for all n, there is N such that $M =_\beta \lambda x_1 \ldots x_n.N$, we can derive $\vdash_\Omega^\nabla \lambda x_1 \ldots x_n.N : \Omega^n \to \Omega$ by (Ax-Ω) and rule ($\to$I). If M is a solvable term such that the head variable of its head normal form is free, i.e. $M =_\beta \lambda\vec{x}.y\vec{N}$, we can derive for all l $y : \Omega^{m+l} \to \Omega \vdash_\Omega^\nabla \lambda\vec{x}.y\vec{N} : \Omega^{n+l} \to \Omega$, where m is the length of $\vec{N}$ and n is the length of $\vec{x}$.

(2.ii) By (2.iii) the head variable of the head normal form of M must be free. We can type a term of the shape $\lambda\vec{x}.y\vec{N}$ where $y \notin \vec{x}$ as follows $y : \Omega^m \to A \vdash_\Omega^\nabla \lambda\vec{x}.y\vec{N} : \Omega^n \to A$, where m is the length of $\vec{N}$ and n is the length of $\vec{x}$. For $\lambda\cap_\Omega^{\mathcal{DHM}}$ by choosing $A \equiv \omega$ we get $y{:}\Omega^m \to \omega \vdash_\Omega^{\mathcal{DHM}} M : \Omega^n \to \omega$, so we conclude $y{:}\omega \vdash_\Omega^{\mathcal{DHM}} M : \omega$ since $\omega \sim_{\mathcal{DHM}} \Omega \to \omega$.

(3) Let $\Gamma_\omega = \{x : \omega \mid x \in \mathsf{V}\}$. It is easy to verify by induction on the definition of λ-terms that using $\omega \sim_\nabla \omega \to \omega$ we can derive $\Gamma_\omega \vdash_\Omega^\nabla M : \omega$ for all λ-terms M. By Theorem 1.16(ii) and (strengthening) we obtain that $\vdash_\Omega^\nabla M : \omega$ whenever M reduces to a closed term. ∎

Remark 3.3 *From the proofs of* (2.iii) *and* (2.ii) *it follows that* $\mathsf{PHN} = \mathsf{PWN} \cap \mathsf{HN}$.

4 Set-theoretic semantics using stable sets

This section is devoted to prove the *if* parts of Theorem 3.2, by showing that all the given characterisations are complete.

The proof technique which we shall adopt to achieve this is uniform for all properties, and it is based on the *set theoretic semantics of intersection types*

[14]. The set-theoretic semantics of a type, for a given applicative structure, is a subset of the structure itself. *Intersection* is interpreted as *set-theoretic intersection*, $\leq$ is interpreted as *set-theoretic inclusion*, and $A \to B$ is interpreted *à la logical relation*, i.e. as a subset of the points of the structure whose functional behaviour is that of mapping all points in A into B.

In the present context, there is only one applicative structure under consideration. This is the *term structure* Λ, i.e. the applicative structure whose domain are the λ-terms and where application is just juxtaposition of terms.

In order to ensure that the interpretations of types consist of terms which satisfy appropriate properties, we need to give the set-theoretic semantics using special classes of *stable* sets, for suitable notions of stability. These stability properties amount essentially to suitable invariants for the set-theoretic operators corresponding to the type constructors. This proof technique has been used by various authors, e.g. stable sets [20], admissible relations [22], essentially in connection with strongly normalising terms. Here we develop a full-blown version of this technique, which is applicable to many other evaluation properties.

We will consider two interpretations of the arrow type constructor, the *simple semantics* and the *weak semantics*. To this end we give the following definition:

Definition 4.1 *Let* $X, Y \subseteq \Lambda$*:*

i) $X \Rightarrow Y = \{M \in \Lambda \mid \forall N \in X \; MN \in Y\}$

ii) $X \Rightarrow^{\mathsf{W}} Y = \{M \in \mathsf{WN} \mid \forall N \in X \; MN \in Y\}$. ■

Now, in accordance to the set-theoretic semantics we put:

Definition 4.2 *(Type Interpretation)*

i) The simple interpretation $[\![\ ]\!]$ *of types in* $\mathbf{T}^{\nabla}$ *induced by the type environment* $\mathcal{V} : \mathbf{C}^{\nabla} \to \mathcal{P}(\Lambda)$ *is defined by:*
(a) $[\![\Omega]\!]_{\mathcal{V}} = \Lambda$ *if* $\Omega \in \mathbf{C}^{\nabla}$*;*
(b) $[\![A]\!]_{\mathcal{V}} = \mathcal{V}(A)$ *if* $A \in \mathbf{C}^{\nabla}$ *and* $A \not\sim_{\nabla} \Omega$*;*
(c) $[\![A \to B]\!]_{\mathcal{V}} = [\![A]\!]_{\mathcal{V}} \Rightarrow [\![B]\!]_{\mathcal{V}}$*;*
(d) $[\![A \cap B]\!]_{\mathcal{V}} = [\![A]\!]_{\mathcal{V}} \cap [\![B]\!]_{\mathcal{V}}$*.*

ii) The weak interpretation $[\![\ ]\!]^{\mathsf{W}}$ *of types in* $\mathbf{T}^{\nabla}$ *induced by the type environment* $\mathcal{V} : \mathbf{C}^{\nabla} \to \mathcal{P}(\Lambda)$ *is defined as the simple interpretation but for clause (c), which now is taken to be:*
(c′) $[\![A \to B]\!]^{\mathsf{W}}_{\mathcal{V}} = [\![A]\!]^{\mathsf{W}}_{\mathcal{V}} \Rightarrow^{\mathsf{W}} [\![B]\!]^{\mathsf{W}}_{\mathcal{V}}$*.* ■

Notice that if $\Omega \in \mathbf{C}^{\nabla}$ then $[\![\Omega]\!]_{\mathcal{V}} = [\![\Omega]\!]^{\mathsf{W}}_{\mathcal{V}} = [\![\Omega \to \Omega]\!]_{\mathcal{V}} = \Lambda$ and $[\![\Omega \to \Omega]\!]^{\mathsf{W}}_{\mathcal{V}} = \mathsf{WN}$.

The interest of these semantics lies in the Soundness Theorem 4.5, below. But in order to be able to state it we need some further definitions.

Definition 4.3 *i) A type environment $\mathcal{V}$* agrees *with a type theory $\Sigma^{\bigtriangledown}$ if and only if*
(a) $\forall N \in [\![A]\!]_{\mathcal{V}}.\ M[x := N] \in [\![B]\!]_{\mathcal{V}}$ implies $\lambda x.M \in [\![A \to B]\!]_{\mathcal{V}}$;
(b) if $A \leq_{\bigtriangledown} B$ then $[\![A]\!]_{\mathcal{V}} \subseteq [\![B]\!]_{\mathcal{V}}$.

ii) A type environment $\mathcal{V}$ W-agrees *with a type theory $\Sigma^{\bigtriangledown}$ if and only if*
(a) $\forall N \in [\![A]\!]^{\mathsf{W}}_{\mathcal{V}}.\ M[x := N] \in [\![B]\!]^{\mathsf{W}}_{\mathcal{V}}$ implies $\lambda x.M \in [\![A \to B]\!]^{\mathsf{W}}_{\mathcal{V}}$;
(b) if $A \leq_{\bigtriangledown} B$ then $[\![A]\!]^{\mathsf{W}}_{\mathcal{V}} \subseteq [\![B]\!]^{\mathsf{W}}_{\mathcal{V}}$. ∎

Looking at the weak interpretations of Ω and $\Omega \to \Omega$ it is clear that no environment can W-agree with $\Sigma^{\bigtriangledown}$ whenever $\Omega \sim_{\bigtriangledown} \Omega \to \Omega$.

Definition 4.4 *(Semantic Satisfiability) Let $\rho : \mathsf{V} \to \Lambda$.*

i) $[\![M]\!]_{\rho} = M[\vec{x} := \vec{N}]$ where $\vec{x} = FV(M)$ and $\rho(\vec{x}) = \vec{N}$;

ii) $\rho, \mathcal{V} \models M : A$ if and only if $[\![M]\!]_{\rho} \in [\![A]\!]_{\mathcal{V}}$;

iii) $\rho, \mathcal{V} \models \Gamma$ if and only if $\rho, \mathcal{V} \models x : B$ for all $x{:}B \in \Gamma$;

iv) $\Gamma \models^{\bigtriangledown} M : A$ if and only if $\rho, \mathcal{V} \models \Gamma$ implies $\rho, \mathcal{V} \models M : A$ for all $\mathcal{V}$ which agree with $\Sigma^{\bigtriangledown}$, and all ρ.

v) Similarly
- *$\rho, \mathcal{V} \models_{\mathsf{W}} \Gamma$ if and only if $[\![x]\!]_{\rho} \in [\![B]\!]^{\mathsf{W}}_{\mathcal{V}}$ for all $x{:}B \in \Gamma$;*
- *$\Gamma \models^{\bigtriangledown}_{\mathsf{W}} M : A$ if and only if $\rho, \mathcal{V} \models_{\mathsf{W}} \Gamma$ implies $[\![M]\!]_{\rho} \in [\![A]\!]^{\mathsf{W}}_{\mathcal{V}}$ for all $\mathcal{V}$ which* W*-agrees with $\Sigma^{\bigtriangledown}$ and all ρ.* ∎

Finally we can give:

Theorem 4.5 (Soundness) *$\Gamma \vdash^{\bigtriangledown} M : A$ implies $\Gamma \models^{\bigtriangledown} M : A$ and $\Gamma \models^{\bigtriangledown}_{\mathsf{W}} M : A$.*

Proof. By induction on derivations. The restriction to type environments which agree with $\Sigma^{\bigtriangledown}$ is essential for the soundness of rules ($\to$I) and ($\leq_{\bigtriangledown}$). ∎

The above theorem is a very powerful tool for proving properties of typable terms, which will be constantly used in the completeness part of the proof of Theorem 3.2. Roughly the idea is the following. In order to show that a term, typable in a given type theory (or with a given type, in a given type theory) has a given property, we pick a suitable type environment which agrees with that type theory and show that all terms in the interpretations of all the types (or in the interpretation of the type in question) satisfy that property. Usually variables belong to the interpretations of types, or else we are interested only in closable terms. So, in both cases, by taking the *identity* term environment $\rho_0(x) = x$ one has that $[\![M]\!]_{\rho_0} = M$, and so, if a term is typable, then it satisfies the property in question.

The difficulty, of course, lies in showing that the properties in question are satisfied by the sets in the range of the type environments and that they are preserved by the "intersection" and the "arrow" constructions. As is normal with these inductive proofs, a possibly stronger hypothesis than the one that all terms in the interpretation of the type satisfy the property in question has to be assumed. After [20] we shall refer to these induction hypotheses as *stability* properties.

The stability properties we shall be interested in are the following:

Definition 4.6 *i) A set* $X \subseteq \mathsf{WN}$ *is* WN-type-stable *if it contains* $x\vec{M}$ *for all* $\vec{M} \in \Lambda$*, and it is closed under head expansion of redexes;*

ii) A set $X \subseteq \mathsf{HN}$ *is* HN-type-stable *if it contains* $x\vec{M}$ *for all* $\vec{M} \in \Lambda$ *and it is closed under head expansion of redexes;*

iii) A set $X \subseteq \mathsf{N}$ *is* N-type-stable *if it contains* $x\vec{M}$ *for all* $\vec{M} \in \mathsf{N}$ *and it is closed under head expansion of redexes;*

iv) A set $X \subseteq \mathsf{SN}$ *is* SN-type-stable *if it contains* $x\vec{M}$ *for all* $\vec{M} \in \mathsf{SN}$ *and it is closed under head expansion of* λ-**I**-*redexes or of* λ-**K**-*redexes*[2] *whose argument is in* SN. ■

Notice that none of the stable sets in the above definition can be empty.

The above definitions were given essentially to be able to show the following proposition, namely that the stability properties are preserved under suitable *set-theoretic* constructions. This result will imply, *inter alia*, that all sets in the range of the appropriate type interpretations satisfy the appropriate stability property.

Proposition 4.7 *Let* $\mathsf{S} \in \{\mathsf{WN}, \mathsf{HN}, \mathsf{N}, \mathsf{SN}\}$*,* $\mathsf{T} \in \{\mathsf{HN}, \mathsf{N}, \mathsf{SN}\}$*, and* $X, Y \subseteq \Lambda$*.*

i) If Y *is closed under head expansion of some kinds of redexes then both* $X \Rightarrow^{\mathsf{W}} Y$ *and* $X \Rightarrow Y$ *are closed under head expansion of the same kinds of redexes for all* $X \subseteq \Lambda$*;*

ii) If X, Y *are closed under head expansion of some kinds of redexes then* $X \cap Y$ *is closed under head expansion of the same kinds of redexes;*

iii) Each S *is* S*-type-stable;*

iv) $\Lambda \Rightarrow^{\mathsf{W}} \Lambda$ *is* WN*-type-stable;*

[2] $(\lambda x.M)N$ is a λ-**K**-redex if and only if $x \notin FV(M)$.

v) If Y is WN*-type-stable then* $\Lambda \Rightarrow^{\mathsf{W}} Y$ *is* WN*-type-stable;*

vi) If X, Y are WN*-type-stable then* $X \Rightarrow^{\mathsf{W}} Y$ *is* WN*-type-stable;*

vii) If Y is HN*-type-stable then* $\Lambda \Rightarrow Y$ *is* HN*-type-stable;*

viii) If X, Y are T*-type-stable then* $X \Rightarrow Y$ *is* T*-type-stable;*

ix) If X, Y are S*-type-stable then* $X \cap Y$ *is* S*-type-stable;*

x) If X is S*-type-stable then* $X \cap \Lambda$ *is* S*-type-stable.*

Proof. We show only (iv), (v), (vi), (vii), and (viii), the other points being immediate.

First notice that $X \Rightarrow^{\mathsf{W}} Y \subseteq \mathsf{WN}$ for all $X, Y \subseteq \Lambda$ by definition. Moreover $Mx \in \mathsf{T}$ implies $M \in \mathsf{T}$ for $\mathsf{T} \in \{\mathsf{HN}, \mathsf{N}, \mathsf{SN}\}$, and therefore from $Y \subseteq \mathsf{T}$ and $x \in X$ we get $X \Rightarrow Y \subseteq \mathsf{T}$.

If Y is Λ or it is WN-type-stable, then it contains $x\vec{M}$ for all $\vec{M} \in \Lambda$ and therefore $x\vec{M} \in X \Rightarrow^{\mathsf{W}} Y$ for all $\vec{M} \in \Lambda$ and for all $X \subseteq \Lambda$. Similarly $x\vec{M} \in X \Rightarrow Y$ for all $\vec{M} \in \mathsf{T}$ and for all $X \subseteq \Lambda$ whenever Y is T-type-stable for $\mathsf{T} \in \{\mathsf{HN}, \mathsf{N}, \mathsf{SN}\}$. We conclude using points (i) and (ii). ∎

Now we define the type environments which will be considered in the completeness part of the proof of Theorem 3.2.

Definition 4.8 *(Type Environments)*

i) The type environment $\mathcal{V}_{\mathcal{B}a}$ *is defined by:*

$$\mathcal{V}(A) = \mathsf{SN} \textit{ if } A \in \mathbf{C}_\infty.$$

ii) The type environment $\mathcal{V}^1_{\mathcal{BCD}}$ *is defined by:*

$$\mathcal{V}(A) = \mathsf{HN} \textit{ if } A \in \mathbf{C}_\infty.$$

iii) The type environment $\mathcal{V}^2_{\mathcal{BCD}}$ *is defined by:*

$$\mathcal{V}(A) = \mathsf{N} \textit{ if } A \in \mathbf{C}_\infty.$$

iv) The type environment $\mathcal{V}_{\mathcal{CDZ}}$ *is defined by:*

$$\mathcal{V}(\omega) = \mathsf{PN}; \quad \mathcal{V}(\varphi) = \mathsf{N}.$$

v) The type environment $\mathcal{V}_{\mathcal{DHM}}$ *is defined by:*

$$\mathcal{V}(\omega) = \mathsf{PHN}; \quad \mathcal{V}(\varphi) = \mathsf{HN}.$$

vi) The type environment $\mathcal{V}_{\mathcal{S}c}$ *is defined by:*

$$\mathcal{V}(\omega) = \mathsf{PHN}.$$

vii) The type environment $\mathcal{V}_{\mathcal{P}a}$ *is defined by:*

$$\mathcal{V}(\omega) = \mathsf{C}. \blacksquare$$

Notation 4.9 $\mathcal{V}_{\mathcal{BCD}}$ *stands for both* $\mathcal{V}^1_{\mathcal{BCD}}$ *and* $\mathcal{V}^2_{\mathcal{BCD}}$.

It is easy to verify, using the following Propositions 4.10 and 4.11, that each type environment $\mathcal{V}_\triangledown$ above agrees (or W-agrees) with the corresponding type theory $\Sigma^\triangledown$. Moreover all type environments agree and W-agree with the type theory $\Sigma^{\mathcal{AO}}$: this follows from Proposition 4.11(iii) taking into account the interpretations of Ω and $\Omega \to \Omega$ (see Definition 4.2 and the following sentence).

Proposition 4.10 *i)* $\mathsf{PN} = \mathsf{N} \Rightarrow \mathsf{PN}$.

ii) $\mathsf{PHN} = \Lambda \Rightarrow \mathsf{PHN}$.

iii) $\mathsf{N} = \mathsf{PN} \Rightarrow \mathsf{N}$.

iv) $\mathsf{HN} = \mathsf{PHN} \Rightarrow \mathsf{HN}$.

v) $\mathsf{C} = \mathsf{C} \Rightarrow \mathsf{C}$.

Proof. All cases are immediate but the inclusion $\mathsf{N} \subseteq \mathsf{PN} \Rightarrow \mathsf{N}$. We show that if $M \in \mathsf{PN}$ and $N \in \mathsf{NF}$ then $NM \in \mathsf{N}$. If N is λ-free, i.e. N is of the shape $x\vec{N}$, then NM is the normal form $x\vec{N}M$. Otherwise let $N \equiv \lambda x.N'$. The proof is by induction on the number of occurrences of x in N'. The basic step, that is x does not occur in N', is immediate since $NM \longrightarrow_\beta N'$. If x occurs in N', let $N' \equiv C[x]$, where the hole in $C[\,]$ identifies the left-most occurrence of x in N'. Let y be fresh: by induction $(\lambda x.C[y])M \twoheadrightarrow_\beta C'[y] \in \mathsf{NF}$. By construction there is exactly one hole in $C'[\,]$. Let $\vec{N}$ be all the terms to which $[\,]$ is applied in $C'[\,]$. Since $M \in \mathsf{PN}$, $M\vec{N} \in \mathsf{N}$ and therefore $(\lambda y.C'[y])M \in \mathsf{N}$ too. We conclude $NM \in \mathsf{N}$ since $NM =_\beta (\lambda xy.C[y])MM =_\beta (\lambda y.C'[y])M$. $\blacksquare$

Proposition 4.11 *i) For* $\triangledown \in \{\mathcal{BCD}, \mathcal{CDZ}, \mathcal{S}c, \mathcal{P}a, \mathcal{DHM}\}$ *and for all types* $A \in \mathbf{T}^\triangledown$, *all* $M, N \in \Lambda$*:*

$$\text{If } M[x := N] \in [\![A]\!]_{\mathcal{V}_\triangledown} \text{ then } (\lambda x.M)N \in [\![A]\!]_{\mathcal{V}_\triangledown}.$$

ii) For all types $A \in \mathbf{T}^{\mathcal{B}a}$ *and all* $M \in \Lambda$, *all* $N \in \mathsf{SN}$*:*

$$\text{If } M[x := N] \in [\![A]\!]_{\mathcal{V}_{\mathcal{B}a}} \text{ then } (\lambda x.M)N \in [\![A]\!]_{\mathcal{V}_{\mathcal{B}a}}.$$

iii) For all types $A \in \mathbf{T}^{\mathcal{AO}}$, *all* $M, N \in \Lambda$ *and all environments* $\mathcal{V}$*:*

$$\text{If } M[x := N] \in [\![A]\!]^{\mathsf{W}}_{\mathcal{V}} \text{ then } (\lambda x.M)N \in [\![A]\!]^{\mathsf{W}}_{\mathcal{V}}.$$

Proof. The proofs by induction on the structure of A follow from Definition 4.8 and Proposition 4.7(i),(ii). ■

Proof of Theorem 3.2(⇐). Take $\rho_0(x) = x$. Notice that $\rho_0, \mathcal{V} \models \Gamma$ and $\rho_0, \mathcal{V} \models_{\mathsf{W}} \Gamma$ for all $\mathcal{V}$ and Γ such that if $x{:}B \in \Gamma$ then either $[\![B]\!]_{\mathcal{V}}$ is Λ or $[\![B]\!]_{\mathcal{V}}$ is S-type-stable for some $\mathsf{S} \in \{\mathsf{WN}, \mathsf{HN}, \mathsf{N}, \mathsf{SN}\}$, since in both cases $[\![B]\!]_{\mathcal{V}}$ will contain all free variables.

(1.iv) It is easy to check using Proposition 4.7 that for all $A \in \mathbf{T}^{\mathcal{AO}}$ and all $\mathcal{V}$ either $A \sim_{\mathcal{AO}} \Omega$ and $[\![A]\!]^{\mathsf{W}}_{\mathcal{V}} = \Lambda$ or $A \not\sim_{\mathcal{AO}} \Omega$ and $[\![A]\!]^{\mathsf{W}}_{\mathcal{V}}$ is WN-type-stable. From above we get $\rho_0, \mathcal{V} \models_{\mathsf{W}} \Gamma$ for all $\mathcal{V}$ and Γ. Moreover $[\![\Omega \to \Omega]\!]^{\mathsf{W}}_{\mathcal{V}} = \mathsf{WN}$. Then from $\Gamma \vdash^{\mathcal{AO}}_{\Omega} M : \Omega \to \Omega$ we get by soundness $\Gamma \models^{\mathcal{AO}}_{\mathsf{W}} M : \Omega \to \Omega$, i.e. $M = [\![M]\!]_{\rho_0} \in [\![\Omega \to \Omega]\!]^{\mathsf{W}}_{\mathcal{V}} \subseteq \mathsf{WN}$, so we conclude $M \in \mathsf{WN}$.

(1.iii) For $\lambda\cap^{\mathcal{BCD}}_{\Omega}$ it is easy to check using Proposition 4.7 that for all $A \in \mathbf{T}^{\mathcal{BCD}}$ either $A \sim_{\mathcal{BCD}} \Omega$ and $[\![A]\!]_{\mathcal{V}^1_{\mathcal{BCD}}} = \Lambda$ or $A \not\sim_{\mathcal{BCD}} \Omega$ and $[\![A]\!]_{\mathcal{V}^1_{\mathcal{BCD}}}$ is HN-type-stable. So we have $\rho_0, \mathcal{V}^1_{\mathcal{BCD}} \models \Gamma$ for all Γ. From $\Gamma \vdash^{\mathcal{BCD}}_{\Omega} M : (\Omega^m \to A)^n \to A$ we get by soundness $\Gamma \models^{\mathcal{BCD}} M : (\Omega^m \to A)^n \to A$, i.e. $M = [\![M]\!]_{\rho_0} \in \mathsf{HN}$.
For $\lambda\cap^{\mathcal{DHM}}_{\Omega}$ let Γ_ω be the $\mathcal{DHM}$-basis all whose predicates are ω. By Definition 2.2 each free variable belongs to PHN and therefore $\rho_0, \mathcal{V}_{\mathcal{DHM}} \models \Gamma_\omega$. From $\Gamma_\omega \vdash^{\mathcal{DHM}} M : \varphi$ we get by soundness $M \in \mathsf{HN}$.

(1.ii) For $\lambda\cap^{\mathcal{BCD}}_{\Omega}$ observe that by Proposition 4.7 $[\![A]\!]_{\mathcal{V}^2_{\mathcal{BCD}}}$ is N-type-stable whenever Ω does not occur in A. Therefore $\rho_0, \mathcal{V}^2_{\mathcal{BCD}} \models \Gamma$, since by hypothesis Ω does not occur in Γ. So as in case (1.iii) we get by soundness $M \in \mathsf{N}$.
For $\lambda\cap^{\mathcal{CDZ}}_{\Omega}$ the proof is similar to that of case (1.iii) for $\lambda\cap^{\mathcal{DHM}}_{\Omega}$.

(1.i) The proof is similar to that of case (1.ii) for $\lambda\cap^{\mathcal{BCD}}_{\Omega}$ by observing that $[\![A]\!]_{\mathcal{V}_{\mathcal{B}a}}$ is SN-type-stable for all $A \in \mathbf{T}^{\mathcal{B}a}$.

(2.iii) By definition $[\![\Omega^{n+1} \to \Omega]\!]^{\mathsf{W}}_{\mathcal{V}} = \{M \in \mathsf{WN} \mid \forall \vec{N} \text{ of length } n, M\vec{N} \in \mathsf{WN}\}$, then $M \in \bigcap_{n \in \mathbf{N}} [\![\Omega^n \to \Omega]\!]^{\mathsf{W}}_{\mathcal{V}}$ if and only if $M \in \mathsf{PWN}$. The result follows by soundness as in case (1.iv).

(2.ii) For $\lambda\cap^{\mathcal{S}c}_{\Omega}$ first notice that $\omega \leq_{\mathcal{S}c} A$ for all $A \in \mathbf{T}^{\mathcal{S}c}$. This can be easily checked by induction on A. If $A \equiv B \to C$ then by induction $\omega \leq_{\mathcal{S}c} C$ so we get $\omega \sim_{\mathcal{S}c} \Omega \to \omega \leq_{\mathcal{S}c} B \to C$ by rule (η) since $B \leq_{\mathcal{S}c} \Omega$ by axiom (Ω). If for all $A \in \mathbf{T}^{\mathcal{S}c}$ there are a $\mathcal{S}c$-basis Γ and an integer n such that $\Gamma \vdash^{\mathcal{S}c}_{\Omega} M : \Omega^n \to A$, by choosing $A \equiv \omega$ we get that there is a $\mathcal{S}c$-basis Γ_0 such that $\Gamma_0 \vdash^{\mathcal{S}c}_{\Omega} M : \omega$ by rule $(\leq_{\mathcal{S}c})$ since $\Omega^n \to \omega \sim_{\mathcal{S}c} \omega$. This implies $\Gamma_\omega \vdash^{\mathcal{S}c}_{\Omega} M : \omega$ by rule $(\leq_{\mathcal{S}c}\,\mathrm{L})$. So it suffices to show that $\Gamma_\omega \vdash^{\mathcal{S}c}_{\Omega} M : \omega$ implies $M \in \mathsf{PHN}$. This can be proved similarly to case (1.ii) for $\lambda\cap^{\mathcal{CDZ}}_{\Omega}$ using the type interpretation $\mathcal{V}_{\mathcal{S}c}$.
For $\lambda\cap^{\mathcal{DHM}}_{\Omega}$ the proof is similar since $\omega \leq_{\mathcal{DHM}} \varphi$ and $\omega \sim_{\mathcal{DHM}} \Omega \to \omega$.

(2.i) The first observation is that $\Omega \in A$ iff $\Omega \in_{\mathcal{CDZ}} A$. We now show that

$\omega \leq_{\mathcal{CDZ}} A \leq_{\mathcal{CDZ}} \varphi$ for all $A \in \mathbf{T}^{\mathcal{CDZ}}$ such that $\Omega \notin A$ by induction on A. The only interesting case is $A \equiv B \to C$: in this case by induction $\omega \leq_{\mathcal{CDZ}} B \leq_{\mathcal{CDZ}} \varphi$, $\omega \leq_{\mathcal{CDZ}} C \leq_{\mathcal{CDZ}} \varphi$ so we get $\omega \sim_{\mathcal{CDZ}} \varphi \to \omega \leq_{\mathcal{CDZ}} B \to C \sim_{\mathcal{CDZ}} \omega \to \varphi \sim_{\mathcal{CDZ}} \varphi$ by rule (η). If for all $A \in \mathbf{T}^{\mathcal{CDZ}}$ such that $\Omega \notin A$ there is a $\mathcal{CDZ}$-basis Γ such that $\Omega \notin \Gamma$ and $\Gamma \vdash_\Omega^{\mathcal{CDZ}} M : A$, by choosing $A \equiv \omega$ we get that there is a $\mathcal{CDZ}$-basis Γ_0 such that $\Omega \notin \Gamma_0$ and $\Gamma_0 \vdash_\Omega^{\mathcal{CDZ}} M : \omega$. This implies $\Gamma_\omega \vdash_\Omega^{\mathcal{CDZ}} M : \omega$ by rule ($\leq_{\mathcal{CDZ}}$ L). So it suffices to show that $\Gamma_\omega \vdash_\Omega^{\mathcal{CDZ}} M : \omega$ implies $M \in \mathsf{PN}$. This can be proved similarly to case (1.ii) for $\lambda\cap_\Omega^{\mathcal{CDZ}}$.

(3) Clearly $\rho, \mathcal{V} \models \emptyset$ for all $\rho, \mathcal{V}$. The result follows immediately by soundness. ■

5 Concluding remarks

Two natural questions, at least, lurk behind this paper: "can we characterise in some significant way the class of evaluation properties which we can characterise using intersection types?" and "is there a method for going from a logical specification of a property to the appropriate intersection type theory?".

Regarding the first question, we have seen that the properties have to be closed, at least, under some form of β-expansion. But clearly this is not the whole story. Probably the answer to this question is linked to some very important open problems in the theory of the denotational semantics of untyped λ-calculus, like the existence of a denotational model whose theory is precisely $\lambda\beta$. As far as the latter question is concerned, we really have no idea. It seems that we are still missing something in our understanding of intersection types.

Of course there are some partial answers. For instance by looking at what happens in particular filter models, one can draw some inspiration and sometimes even provide some interesting characterisations. In this paper we discussed closable sets. Another example would have been, for instance, that of those terms which *reduce to terms of the* λ-$\mathbf{I}$-*calculus*. Here the filter model under consideration is the one in [19], generated by the theory $\Sigma^{\mathcal{HR}} = \Sigma(\{\Omega, \varphi, \omega\}, \mathcal{BCD} \cup \{(\omega\varphi), (\varphi{\to}\omega), (\omega\text{-}I)\})$, where $(\omega\text{-}I)$ is the rule $(\varphi{\to}\varphi)\cap(\omega{\to}\omega) \sim \varphi$. The terms typable with φ in $\lambda\cap_\Omega^{\mathcal{HR}}$, for the $\mathcal{HR}$-basis where all variables have type φ, are then precisely those which reduce to terms of the λ-$\mathbf{I}$-calculus [19]. These characterisations however appear quite *accidental*. And we feel that we lack yet a general theory which could allow us to streamline the approach. Given the model we can start to guess. And when we are successful, as in this case, we can achieve generality only artificially, by considering all those type theories which extend the theory of the filter model in question.

For one thing this method of drawing inspiration from filter models is interesting, in that it provides some very interesting conjectures. Perhaps the best example concerns *persistently strongly normalising* terms. These are those strongly normalising terms M, such that for all vectors $\vec{N}$ of strongly normalising terms, $M\vec{N}$ is still strongly normalising. Consider the filter model introduced in [18], generated by the type theory obtained by pruning the type theory $\Sigma^{\mathcal{CDZ}}$ of all types including Ω, i.e. generated by the theory $\Sigma^{\mathcal{HL}} = \Sigma(\{\varphi, \omega\}, \mathcal{B}a \cup \{(\omega\varphi), (\varphi{\to}\omega), (\omega{\to}\varphi)\})$. The natural conjecture is then, in analogy to what happens for persistently normalising terms, "are the terms typable with ω in $\lambda\cap^{\mathcal{HL}}_{\mathcal{B}}$, for the $\mathcal{HL}$-basis where all variables have type ω precisely the *persistently strongly normalising* ones?". Completeness is clear, but to show soundness some independent syntactical characterisation of that class of terms appears necessary. The set of persistently strongly normalising terms does not include $\mathsf{PN} \cap \mathsf{SN}$. A counter example is $M \equiv \lambda x.a((\lambda y.b)(xx))$ since $M(\lambda z.zz) \notin \mathsf{SN}$. This conjecture still resists proof.

The results and the techniques of the present paper have been widely used and developed in [13], which mainly focus on the construction of λ-models characterising computational properties of terms.

Acknowledgements

The authors are very grateful to F. Alessi for very stimulating discussions on the subject of the present paper. Moreover they like to thank the referees of MFCS and TCS submissions for their useful remarks and suggestions.

Appendix Polarised normal forms

In this Appendix we will show that, for all arrow-type theories Σ^{∇}, each persistently normalising λ-term M can be typed with an arbitrary type not containing Ω modulo $\sim_{\nabla}$ from a suitable ∇-basis. Our proof is organised as follows. First we introduce the notions of adjacent occurrences of variables, positive and negative variables, polarised normal forms, principal decorations and replacement paths. Then we show the key property (Lemma A.13):

for each normal form with adjacent occurrences of negative variables we can build a substitution such that the resulting term does not have normal form.

This fact suggests the notions of positive normal forms and strongly polarised normal forms. We conclude by showing that:

- each persistently normalising normal form is a positive normal form (Theorem A.15);

- the principal decoration of a positive normal form is a strongly polarised normal form (Proposition A.20);
- each strongly polarised normal form which is a principal decoration can be typed with an arbitrary type not containing Ω modulo $\sim_\bigtriangledown$ from a suitable $\bigtriangledown$-basis in all arrow-type theories $\Sigma^\bigtriangledown$ (Theorem A.23).

We give now some definitions concerning only terms in normal form. We do forbid α-conversion: in this way also the names of bound variables are meaningful. Moreover this leads us to consider λ-terms in which different bound variables may have the same names, and also bound and free variables may have the same name.

Definition A.1 *i) In a normal form of the shape* $x\vec{M}(\lambda\vec{z}.y\vec{N})$ *we say that the showed occurrences of* x *and* y *are* adjacent*. Notice that we can have* $x \equiv y$*.*

ii) Two (not necessary distinct!) variables have adjacent occurrences in a normal form M *if and only if they have adjacent occurrences in a subterm of* M*.*

iii) If $M \equiv xN_1 \dots N_i \dots N_m$ *we say that the subterm* N_i *is the* i*-th argument of* x *in* M*.*

iv) If $M \equiv \lambda y_1 \dots y_j \dots y_n.x\vec{N}$ *we say that:*

(a) the variables $y_1 \dots y_j \dots y_n$ *are the variable bound by the initial abstractions of* M*;*

(b) the variable y_j *is the variable bound by the* j*-th abstraction of* M*.* ■

Remark A.2 *An alternative definition of adjacent occurrences can be done using the Böhm trees of* λ*-terms as defined in [6] (Definition 10.1.4): two occurrences* x*,* y *are adjacent in* M *if and only if they correspond to two nodes father-son in the Böhm tree of* M *with labels* $\lambda\vec{z}.x$ *and* $\lambda\vec{t}.y$ *for some* $\vec{z}, \vec{t}$*.*

Example A.3 *In the normal form* $\lambda x.x(\lambda t.x)(\lambda uz.u(zt))$*:*

- *the underlined occurrences of variables are adjacent:*

$$\lambda x.\underline{x}(\lambda t.\underline{x})(\lambda uz.u(zt)) \qquad \lambda x.\underline{x}(\lambda t.x)(\lambda uz.\underline{u}(zt))$$

$$\lambda x.x(\lambda t.x)(\lambda uz.\underline{u}(\underline{z}t)) \qquad \lambda x.x(\lambda t.x)(\lambda uz.u(\underline{zt}))$$

- $\lambda uz.u(zv)$ *is the 2-th argument of* x *in* $x(\lambda t.x)(\lambda uz.u(zt))$
- x *is the variable bound by the initial abstraction of* $\lambda x.x(\lambda t.x)(\lambda uz.u(zt))$
- z *is the variable bound by the 2-th abstraction of* $\lambda uz.u(zt)$*.*

Figure 4 shows the Böhm tree of $\lambda x.x(\lambda t.x)(\lambda uz.u(zt))$*.*

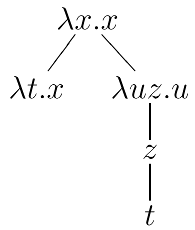

Fig. 4. Böhm tree of $\lambda x.x(\lambda t.x)(\lambda uz.u(zt))$.

We need to introduce the notion of *polarity* for term variables.

Definition A.4 (Polarised normal forms) *Assume that the variables of λ-calculus are partitioned into two infinite sets of positive and negative variables, i.e. x^+ or x^-. Let $\Lambda^\pm$ be the resulting language of polarised λ-terms, and define the set of* polarised normal forms, $\mathsf{NF}^{\mathsf{i},\mathsf{j}}$ *as follows:*

$$(+\,app)\ \frac{\vec{M} \in \mathsf{NF}^{+,+} \cup \mathsf{NF}^{+,-}}{x^+\vec{M} \in \mathsf{NF}^{+,+} \cap \mathsf{NF}^{-,+}} \quad (-\,app)\ \frac{\vec{M} \in \mathsf{NF}^{-,+} \cup \mathsf{NF}^{-,-}}{x^-\vec{M} \in \mathsf{NF}^{+,-} \cap \mathsf{NF}^{-,-}}$$

$$(+\,abs)\ \frac{M \in \mathsf{NF}^{+,\mathsf{j}}}{\lambda x^+.M \in \mathsf{NF}^{+,\mathsf{j}}} \qquad\qquad (-\,abs)\ \frac{M \in \mathsf{NF}^{-,\mathsf{j}}}{\lambda x^-.M \in \mathsf{NF}^{-,\mathsf{j}}} \quad \blacksquare$$

Notice that $\mathsf{NF}^{\mathsf{i},\mathsf{j}} \subseteq \mathsf{NF}$ for all $\mathsf{i}, \mathsf{j} \in \{+,-\}$.

Example A.5 *We can derive $\lambda x^+.x^+x^+ \in \mathsf{NF}^{+,+}$ as follows:*

$$\dfrac{\dfrac{\dfrac{}{x^+ \in \mathsf{NF}^{+,+}}\,(+\,app)}{x^+x^+ \in \mathsf{NF}^{+,+}}\,(+\,app)}{\lambda x^+.x^+x^+ \in \mathsf{NF}^{+,+}}\,(+\,abs)$$

Similarly we can derive $\lambda x^-.x^-x^- \in \mathsf{NF}^{-,-}$.

Rule $(+\,app)$ says that we can apply a positive variable only to normal forms whose initial bound variables are positive independently from the polarity of the head variables. The so obtained normal form belongs both to $\mathsf{NF}^{+,+}$ and $\mathsf{NF}^{-,+}$: in fact it is λ-free and it's head variable is positive. Similarly rule $(-\,app)$ allows to apply a negative variable to all normal forms whose initial bound variables are negative independently from the polarity of the head variables. The rules for abstractions force all consecutive abstractions to have the same sign.

In other words, $M \in \mathsf{NF}^{\mathsf{i},\mathsf{j}}$ means that the variables bound by the initial abstractions of M have polarity i, the head variable of M has polarity j and the components of M belong to $\mathsf{NF}^{\mathsf{j},+}$ or to $\mathsf{NF}^{\mathsf{j},-}$. A λ-free normal form can belong to both $\mathsf{NF}^{+,\mathsf{j}}$ and $\mathsf{NF}^{-,\mathsf{j}}$, since we do not know the polarity of missing

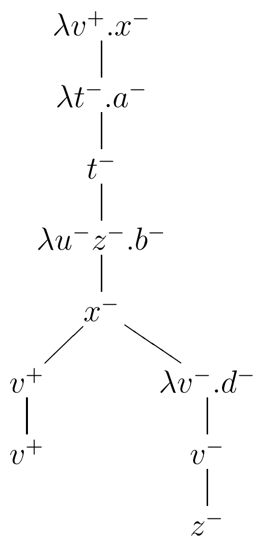

Fig. 5. Böhm tree of the principal decoration of $\lambda v.x(\lambda t.a(t(\lambda uz.b(x(vv)(\lambda v.d(vz))))))$.

bound variables.

Remark A.6 *Looking at the Böhm tree of a polarised normal form we always have that:*

- *the variables abstracted in the same node have the same polarities;*
- *if the node $\lambda\vec{z}.x^{\mathsf{i}}$ is the father of the node $\lambda t_1^{\mathsf{j}} \ldots t_n^{\mathsf{j}}.y$ then $\mathsf{i} = \mathsf{j}$.*

There is a natural way of associating polarised normal forms to normal forms.

Definition A.7 (Decoration) *A polarised normal form $N \in \Lambda^{\pm}$ is a* decoration *of a normal form $M \in \Lambda$ if and only if M is obtained from N by erasing all polarities.* ■

Example A.5 shows that a normal form can have more than one decoration. To get a one-one correspondence between polarised normal forms and normal forms it suffices to force the polarities of the free variables and of the variables bound by the initial abstractions.

Definition A.8 (Principal Decoration) *A polarised normal form $N \in \Lambda^{\pm}$ is the* principal decoration *of a normal form $M \in \Lambda$ if and only if:*

i) N is a decoration of M;

ii) all variables bound by the initial abstractions of N are positive;

iii) all free variables in N are negative. ■

Example A.9 *$\lambda x^+.x^+x^+$ is a principal decoration of $\lambda x.xx$, but $\lambda x^-.x^-x^-$ is not. The principal decoration of $\lambda v.x(\lambda t.a(t(\lambda uz.b(x(vv)(\lambda v.d(vz))))))$ is*

$$\lambda v^+.x^-(\lambda t^-.a^-(t^-(\lambda u^- z^-.b^-(x^-(v^+ v^+)(\lambda v^-.d^-(v^- z^-))))))$$

(see Figure 5).

It is easy to check the soundness of previous definition, i.e. that the principal decoration of a normal form is unique, since according to Definition A.4 the polarities of the variables abstracted in proper subterms are uniquely determined. More precisely if $x^{\mathsf{i}}\vec{N}$ is a subterm of N then all variables abstracted in the initial abstractions of the terms in $\vec{N}$ must have polarity i. Clearly all principal decorations belong $\mathsf{NF}^{+,+} \cup \mathsf{NF}^{+,-}$.

Remark A.10 *We can build the principal decoration of a normal form M by using the Böhm tree of M as follows. First we give positive polarities to all variables bound by the initial abstractions of the root and negative polarities to all free variables. Then we propagate polarities by giving the polarity i to all variables bound in a node whose father has head variable of polarity i.*

The above discussion allow us to identify normal forms with their principal decorations. So from now on until the end of this subsection **we convene that variables in normal forms have the polarities of the corresponding principal decorations**.

We need to introduce the *replacement path* of an occurrence of a negative variable in a normal form. This notion was first defined in a less formal way and for the same aim in [11]. Intuitively the replacement path of an occurrence of a negative variable says if that occurrence is free or bound, and in the last case where it is bound. This is useful in order to replace that occurrence using substitutions of free variables.

The replacement path of a free occurrence of a variable is the variable itself. The replacement path of a bound occurrence of a negative variable is the name of a free variable (hence of a negative variable) followed by a sequence of integer pairs. If the replacement path of a given occurrence of a variable y in a normal form M is $x\langle i_1, j_1\rangle \ldots \langle i_n, j_n\rangle$, then there is an occurrence of x in M such that if z_1 is the variable bound by the j_1-th abstraction of the i_1-th argument of x, then there is an occurrence of z_1 in M such that if z_2 is the variable bound by j_2-th abstraction of the i_2 argument of z_1, then ... there is an occurrence of z_{n-1} in M such that y is the variable bound by the j_n-abstraction of the i_n-th argument of z_{n-1}. A constructive definition is Definition A.11.

As usual $\mathsf{C}[\,]$ will denote a context. We convene that the hole $[\,]$ occurs only once in $\mathsf{C}[\,]$ and that $\mathsf{C}[\,]$ is in normal form (we say that $\mathsf{C}[\,]$ is a *normal context*). In this way if $M \equiv \mathsf{C}[x]$ then $\mathsf{C}[\,]$ uniquely identifies one occurrence of x in the normal form M.

$$\begin{array}{c}
v \text{ is free in } d(vz) \\ \hline
\pi(v, d([\,]z)) = v \\ \hline
\pi(v, x(vv)(\lambda v.d([\,]z))) = x\langle 2,1\rangle \\ \hline
\pi(v, b(x(vv)(\lambda v.d([\,]z)))) = x\langle 2,1\rangle \\ \hline
\pi(v, \lambda z.b(x(vv)(\lambda v.d([\,]z)))) = x\langle 2,1\rangle \\ \hline
\pi(v, \lambda uz.b(x(vv)(\lambda v.d([\,]z)))) = x\langle 2,1\rangle \\ \hline
\pi(v, t(\lambda uz.b(x(vv)(\lambda v.d([\,]z))))) = x\langle 2,1\rangle \\ \hline
\pi(v, a(t(\lambda uz.b(x(vv)(\lambda v.d([\,]z)))))) = x\langle 2,1\rangle \\ \hline
\pi(v, \lambda t.a(t(\lambda uz.b(x(vv)(\lambda v.d([\,]z)))))) = x\langle 2,1\rangle \\ \hline
\pi(v, x(\lambda t.a(t(\lambda uz.b(x(vv)(\lambda v.d([\,]z))))))) = x\langle 2,1\rangle \\ \hline
\pi(v, \lambda v.x(\lambda t.a(t(\lambda uz.b(x(vv)(\lambda v.d([\,]z))))))) = x\langle 2,1\rangle
\end{array}$$

$$\begin{array}{c}
z \text{ is free in } b(x(vv)(\lambda v.d(vz))) \\ \hline
\pi(z, b(x(vv)(\lambda v.d(v[\,])))) = z \\ \hline
\pi(z, t(\lambda uz.b(x(vv)(\lambda v.d(v[\,]))))) = t\langle 1,2\rangle \\ \hline
\pi(z, a(t(\lambda uz.b(x(vv)(\lambda v.d(v[\,])))))) = t\langle 1,2\rangle \\ \hline
\pi(z, x(\lambda t.a(t(\lambda uz.b(x(vv)(\lambda v.d(v[\,]))))))) = x\langle 1,1\rangle\langle 1,2\rangle \\ \hline
\pi(v, \lambda v.x(\lambda t.a(t(\lambda uz.b(x(vv)(\lambda v.d(v[\,]))))))) = x\langle 1,1\rangle\langle 1,2\rangle
\end{array}$$

Fig. 6. Examples of replacement paths.

Definition A.11 *i) The* replacement path $\pi(x, \mathsf{C}[\,])$ *of a variable x in a normal context $\mathsf{C}[\,]$ is defined by:*

$$\frac{\textit{if the given occurrence of } x \textit{ is free in } \mathsf{C}[x]}{\pi(x, \mathsf{C}[\,]) = x}$$

$$\frac{\textit{if } \pi(x, \mathsf{C}[\,]) \textit{ is defined}}{\pi(x, y\vec{N}\mathsf{C}[\,]\vec{N'}) = \pi(x, \mathsf{C}[\,])}$$

$$\frac{\pi(x, \mathsf{C}[\,]) = y\alpha \textit{ and } z \neq y}{\pi(x, \lambda z.\mathsf{C}[\,]) = y\alpha}$$

$$\frac{\pi(x, \mathsf{C}[\,]) = y_j\alpha}{\pi(x, zN_1 \dots N_{i-1}(\lambda y_1 \dots y_j \dots y_n.\mathsf{C}[\,])N_{i+1} \dots N_m) = z\langle i,j\rangle\alpha}$$

ii) The replacement path of a given occurrence of a variable x in a normal form M, identified by $\mathsf{C}[\,]$, is $\pi(x, \mathsf{C}[\,])$.

iii) A variable x occurs with replacement path $y\alpha$ in a normal form M if and only if $M \equiv \mathsf{C}[x]$ and $\pi(x, \mathsf{C}[\,]) = y\alpha$ for some context $\mathsf{C}[\,]$. ■

Example A.12 *Let*

$$\begin{aligned}
\mathsf{C}_1[\,] &\equiv \lambda v.[\,](\lambda t.a(t(\lambda uz.b(x(vv)(\lambda v.d(vz)))))) \\
\mathsf{C}_2[\,] &\equiv \lambda v.x(\lambda t.a(t(\lambda uz.b(x(vv)(\lambda v.d([\,]z)))))) \\
\mathsf{C}_3[\,] &\equiv \lambda v.x(\lambda t.a(t(\lambda uz.b(x(vv)(\lambda v.d(v[\,])))))) \\
\mathsf{C}_4[\,] &\equiv \lambda v.x(\lambda t.a(t(\lambda uz.b(x([\,]v)(\lambda v.d(vz)))))).
\end{aligned}$$

Then $\pi(x, \mathsf{C}_1[\,]) = x$, $\pi(v, \mathsf{C}_2[\,]) = x\langle 2,1\rangle$, $\pi(z, \mathsf{C}_3[\,]) = x\langle 1,1\rangle\langle 1,2\rangle$, while $\pi(v, \mathsf{C}_1[\,])$ and $\pi(v, \mathsf{C}_4[\,])$ are undefined. Figure 6 shows the derivations of $\pi(v, \mathsf{C}_2[\,]) = x\langle 2,1\rangle$ and $\pi(z, \mathsf{C}_3[\,]) = x\langle 1,1\rangle\langle 1,2\rangle$. Figure 5 gives the Böhm *tree of the principal decoration of* $\lambda v.x(\lambda t.a(t(\lambda uz.b(x(vv)(\lambda v.d(vz))))))$ *which is* $\mathsf{C}_1[x]$, $\mathsf{C}_2[v]$, $\mathsf{C}_3[z]$ *and* $\mathsf{C}_4[v]$.

The first case of definition A.11 is the basic step in computing replacement paths. The second and third cases simply allow to inherit replacement paths from subterms. The crucial case is the last one, which builds the replacement path in a context from that of a proper sub-context taking into account where the first variable in the given path is bound. Notice that the second and fourth cases are mutually exclusive since $\pi(x, \mathsf{C}[\ ]) = y_j\alpha$ implies that $\pi(x, \lambda y_1 \dots y_j \dots y_n.\mathsf{C}[\ ])$ is undefined, being (the current occurrence of) y_j positive in $\lambda y_1 \dots y_j \dots y_n.\mathsf{C}[x]$.

Notice that the replacement path of an occurrence can be undefined in a subterm and defined in the whole term. This comes from the last clause of Definition A.11. In Example A.12 $\pi(v, \lambda v.d([\,]z))$ is undefined, while $\pi(v, \mathsf{C}_2[\,]) = x\langle 2, 1\rangle$.

It is easy to verify that all occurrences of negative variables have defined replacement paths. Vice versa all occurrences of positive variables have undefined replacement paths

We can now put replacement paths to use in order to state and to prove the key property that if two negative variables have adjacent occurrences in a normal form M then we can replace the free variables of M by normal forms such that the so obtained term does not have a normal form.

Lemma A.13 *If there are two adjacent occurrences of the (negative) variables* z, t *in a normal form* M *with replacements paths* $x\alpha$, $y\beta$, *then there are normal forms* X, Y *such that* $M[x := X, y := Y]$ *does not have a normal form (possibly* $x \equiv y$ *and, in this case,* $X \equiv Y$*).*

Proof. Actually we prove a stronger statement, i.e. we require X and Y to be λ-terms in which all abstracted variables occur at least once as arguments of a free variable. This makes sure that all subterms of all terms obtained out of $M[x := X, y := Y]$ by reduction will never be erased. So we need only build a reduct of $M[x := X, y := Y]$ containing an unsolvable subterm.

The proof is by induction on the sum of the lengths of the current replacement paths, i.e. of α and β. We convene that (the current occurrence of) z is on the left of (the current occurrence of) t in M.

Basic step: In this case α and β are empty, and therefore $x \equiv z$ and $y \equiv t$. By definition M has a subterm of the shape $xN_1 \dots N_{i-1}(\lambda u_1 \dots u_n.yN'_1 \dots N'_m)$.

If $x \not\equiv y$ a possible choice is

$$X \equiv \lambda v_1 \dots v_i.av_1 \dots v_i(v_i u_1 \dots u_n \Delta)$$
$$Y \equiv \lambda w_1 \dots w_m w.bw_1 \dots w_m w(ww)$$

where $\Delta \equiv \lambda w.ww$. Since

$$\begin{array}{l}(xN_1 \dots N_{i-1}(\lambda u_1 \dots u_n.yN'_1 \dots N'_m))[x := X, y := Y] \\ \twoheadrightarrow_\beta a\hat{N}_1 \dots \hat{N}_{i-1}(\lambda u_1 \dots u_n w.b\hat{N}'_1 \dots \hat{N}'_m w(ww))(b\hat{N}'_1 \dots \hat{N}'_m \Delta(\Delta\Delta))\end{array}$$

where $\hat{N} = N[x := X, y := Y]$ (possibly with indexes and $'$), then $M[x := X, y := Y]$ has an unsolvable subterm.

If $x \equiv y$ we can choose

$$X \equiv \lambda v_1 \dots v_k.av_1 \dots v_k(v_{m+1}v_{m+1})(v_i u_1 \dots u_n(v_i u_1 \dots u_n)),$$

where k is the maximum between i and $m+1$. We get

$$\begin{array}{l}(xN_1 \dots N_{i-1}(\lambda u_1 \dots u_n.xN'_1 \dots N'_m))[x := X] \\ \twoheadrightarrow_\beta \lambda v_{i+1} \dots v_k.a\hat{N}_1 \dots \hat{N}_{i-1}(\lambda u_1 \dots u_n.Q)v_{i+1} \dots v_k(PP)(QQ)\end{array}$$

where $\hat{N} = N[x := X]$ (possibly with indexes and $'$), $Q \equiv \lambda v_{m+1} \dots v_k.a\hat{N}'_1 \dots \hat{N}'_m v_{m+1} \dots v_k(v_{m+1}v_{m+1})(Ru_1 \dots u_n(Ru_1 \dots u_n))$, P, R are suitable terms and

$$QQ \twoheadrightarrow_\beta \lambda v_{m+2} \dots v_k.aN'_1 \dots N'_m Q v_{m+2} \dots v_k(QQ)(Ru_1 \dots u_n(Ru_1 \dots u_n)).$$

Since QQ reduces to a term containing QQ, $M[x := X]$ does not have a normal form.

Induction step: we need to distinguish five possible cases:

i) $x \not\equiv y$ and α not empty;

ii) $x \not\equiv y$ and α empty;

iii) $x \equiv y$ and α, β both not empty;

iv) $x \equiv y$ and α empty while β not empty;

v) $x \equiv y$ and α not empty while β empty.

In all cases we exhibit a normal form X' such that there are adjacent occurrences in the normal form M' of $M[x := X']$ but the sum of the lengths of the replacement paths of these occurrences is less than the sum of the lengths of α and β. This allows us to apply the induction.

i) *Case $x \not\equiv y$ and α not empty.* Assume that $\alpha = \langle i, j\rangle\alpha'$. Then M has a subterm of the shape $xN_1 \dots N_{i-1}(\lambda u_1 \dots u_j.N')$ and $u_j\alpha'$ is the replacement path of the current occurrence of z in N'. Notice that $z \equiv u_j$ if α' is empty. Let

$$X' \equiv \lambda v_1 \dots v_i.xv_1 \dots v_i(v_iu_1 \dots u_j)$$

and M' be the normal form of $M[x := X']$ (the existence of M' comes from the fact that X' in $M[x := X']$ is only applied to free variables). On M' we can observe that:

- the occurrences which are adjacent in M remain adjacent in M';
- the variables which occur with replacement path $x\langle i, l\rangle\gamma$ where $1 \leq l \leq j$ in M occur also with replacement path $u_l\gamma$ in M';
- the variables which occur with replacement path $y\gamma$ occur with the same replacement path in M'.

By the above observations there are adjacent occurrences of z, t in M' with replacement paths respectively $u_j\alpha'$ and $y\beta$. Then induction hypothesis applies and we can find normal forms U_j, Y such that $M'[u_j := U_j, y := Y]$ does not reduce to a normal form. Therefore we can choose

$$X \equiv \lambda v_1 \dots v_i.xv_1 \dots v_i(v_iu_1 \dots U_j).$$

ii) *Case $x \not\equiv y$ and α empty.* Then $z \equiv x$. Assume that $\beta = \langle i, j\rangle\beta'$. Then M has a subterm of the shape $yN_1 \dots N_{i-1}(\lambda u_1 \dots u_j.N')$ and $u_j\beta'$ is the replacement path of the current occurrence of t in N'. Notice that $t \equiv u_j$ if β' is empty. Let

$$Y' \equiv \lambda v_1 \dots v_i.xv_1 \dots v_i(v_iu_1 \dots u_j)$$

and M' be the normal form of $M[y := Y']$. On M' we can observe that:

- the occurrences which are adjacent in M remain adjacent in M';
- the variables which occur with replacement path $y\langle i, l\rangle\gamma$ where $1 \leq l \leq j$ in M occur also with replacement path $u_l\gamma$ in M'.

By the above observations there are adjacent occurrences of x, t in M' such that the replacement path of t is $u_j\beta'$. Then induction hypothesis applies and we can find normal forms X, U_j such that $M'[x := X, u_j := U_j]$ does not reduce to a normal form. Therefore we can choose

$$Y \equiv \lambda v_1 \dots v_i . y v_1 \dots v_i (v_i u_1 \dots U_j).$$

iii) *Case $x \equiv y$ and α, β both not empty.* Assume that $\alpha = \langle i, j\rangle\alpha'$ and $\beta = \langle n, m\rangle\beta'$. Then M has a subterm of the shape $xM_1 \dots M_{i-1}(\lambda u_1 \dots u_j.M^*)$, such that $u_j\alpha'$ is the replacement path of the current occurrence of z in M^*, and a subterm of the shape $xN_1 \dots N_{n-1}(\lambda v_1 \dots v_m.N^*)$, such that $v_m\beta'$ is the replacement path of the current occurrence of t in N^* (these subterms may coincide). Observe that the current occurrences of z, t must be subterms of both M^* and N^*, and this implies that the showed occurrences of x are either nested or they coincide and $i = n$. Notice that $z \equiv u_j$ if α' is empty and $t \equiv v_m$ if β' is empty.

We first consider the sub-case $i \neq n$. This implies $u_j \not\equiv v_m$. Let

$$X' \equiv \lambda w_1 \dots w_k . x w_1 \dots w_k (w_i u_1 \dots u_j)(w_n v_1 \dots v_m),$$

where k is the maximum between i and n, and let M' be the normal form of $M[x := X']$. On M' we can observe that:

- the occurrences which are adjacent in M remain adjacent in M';
- the variables which occur with replacement path $x\langle i, l\rangle\gamma$ where $1 \leq l \leq j$ in M occur also with replacement path $u_l\gamma$ in M';
- the variables which occur with replacement path $x\langle n, h\rangle\gamma$ where $1 \leq h \leq m$ in M occur also with replacement path $v_h\gamma$ in M'.

By the above observations there are adjacent occurrences of z, t in M' with replacement paths respectively $u_j\alpha'$ and $v_m\beta'$. Then induction hypothesis applies and we can find normal forms U_j, V_m such that $M'[u_j := U_j, v_m := V_m]$ does not reduce to a normal form. Then we can choose

$$X \equiv \lambda w_1 \dots w_k . x w_1 \dots w_k (w_i u_1 \dots U_j)(w_n v_1 \dots V_m).$$

If $i = n$, let

$$X' \equiv \lambda w_1 \dots w_i . x w_1 \dots w_i (w_i u_1 \dots u_k),$$

where k is the maximum between j and m. On the normal form M' of $M[x := X']$ we can observe that:

- the occurrences which are adjacent in M remain adjacent in M';
- the variables which occur with replacement path $x\langle i, l\rangle\gamma$ where $1 \leq l \leq k$ in M occur also with replacement path $u_l\gamma$ in M'.

By the above observations if β' is not empty then there are adjacent occurrences of z, t in M' with replacement paths respectively $u_j\alpha'$ and $u_m\beta'$. If β' is empty then there are adjacent occurrences of z, u_m in M' with replacement paths respectively $u_j\alpha'$ and u_m. In both cases induction hypothesis applies and we can find normal forms U_j, U_m such that $M'[u_j := U_j, u_m := U_m]$ (or $M'[u_j := U_j]$ if $j = m$) does not reduce to a normal form. Then we can choose

$$X = \begin{cases} \lambda w_1 \dots w_i.xw_1 \dots w_i(w_iu_1 \dots U_j \dots U_m), & \text{if } j < m = k \\ \lambda w_1 \dots w_i.xw_1 \dots w_i(w_iu_1 \dots U_j), & \text{if } j = m = k \\ \lambda w_1 \dots w_i.xw_1 \dots w_i(w_iu_1 \dots U_m \dots U_j), & \text{if } m < j = k. \end{cases}$$

iv) *Case $x \equiv y$ and α empty, while β not empty.* Then $z \equiv x$ and $t \not\equiv x$. Assume that $\beta = \langle i, j\rangle\beta'$. Then M has a subterm of the shape $xM_1 \dots M_{i-1}(\lambda u_1 \dots u_j.M^*)$, such that $u_j\beta'$ is the replacement path of t in M^*, and a subterm of the shape $xN_1 \dots N_{k-1}(\lambda v_1 \dots v_n.tN'_1 \dots N'_m)$ (these subterms may coincide). Notice that $t \equiv u_j$ if β' is empty. Let

$$X' \equiv \lambda w_1 \dots w_i.xw_1 \dots w_i(w_iu_1 \dots u_j)$$

and let M' be the normal form of $M[x := X']$. On M' we can observe that:

- the occurrences which are adjacent in M remain adjacent in M';
- the variables which occur with replacement path $x\langle i, l\rangle\gamma$ where $1 \leq l \leq j$ in M occur also with replacement path $u_l\gamma$ in M'.

By the above observations there are adjacent occurrences of x, t in M' with replacement paths respectively x and $u_j\beta'$. Then induction hypothesis applies and we can find normal forms X'', U_j such that $M'[x := X'', u_j := U_j]$ does not reduce to a normal form. Notice that X'' will be built in the basic step, so

$$X'' \equiv \lambda w_1 \dots w_k.aw_1 \dots w_k(w_kv_1 \dots v_n\Delta).$$

If the subterms $xM_1 \dots M_{i-1}(\lambda u_1 \dots u_j.M^*)$ and $xN_1 \dots N_{k-1}(\lambda v_1 \dots v_n.tN'_1 \dots N'_m)$ do not coincide then the second one must be a subterm of M^*. We can choose

$$X \equiv \lambda w_1 \dots w_h.aw_1 \dots w_h(w_iu_1 \dots u_{j-1}U_j)(w_kv_1 \dots v_n\Delta)$$

where h is the maximum between i and k.

To see why this works, observe that a subterm of (a reduct of) $M[x := X]$ will be $X\hat{N}_1 \dots \hat{N}_{k-1}(\lambda v_1 \dots v_n.T\hat{N}'_1 \dots \hat{N}'_m)$, where T is a subterm of U_j and $\hat{N}$

is a substitution instance of N (possibly with indexes and $'$)[3]. Notice that T is built in the basic step, and therefore

$$T \equiv \lambda r_1 \dots r_m r.br_1 \dots r_m r(rr).$$

Now if $R \equiv \lambda v_1 \dots v_n.T\hat{N}'_1 \dots \hat{N}'_m$ we get

$$\begin{array}{l} X\hat{N}_1 \dots \hat{N}_{k-1}R \longrightarrow\!\!\!\!\!\rightarrow_\beta \lambda w_{k+1} \dots w_h.a\hat{N}_1 \dots \hat{N}_{k-1}Rw_{k+1} \dots w_h S(Rv_1 \dots v_n\Delta), \\ Rv_1 \dots v_n\Delta \longrightarrow\!\!\!\!\!\rightarrow_\beta b\hat{N}'_1 \dots \hat{N}'_m\Delta(\Delta\Delta), \end{array}$$

where S is a suitable term.

If the subterms $xM_1 \dots M_{i-1}(\lambda u_1 \dots u_j.M^*)$ and $xN_1 \dots N_{k-1}(\lambda v_1 \dots v_n.tN'_1 \dots N'_m)$ coincide we get $i = k$, $M_l \equiv N_l$ $(1 \leq l \leq i)$, $t \equiv u_j$ and $j \leq n$, $u_l \equiv v_l$ $(1 \leq l \leq j)$. In this case we can choose

$$X \equiv \lambda w_1 \dots w_i.aw_1 \dots w_i(w_i u_1 \dots u_{j-1}U_j u_{j+1} \dots u_n\Delta).$$

Notice that U_j is built in the basic step, and therefore

$$U_j \equiv \lambda r_1 \dots r_m r.br_1 \dots r_m r(rr).$$

To see why this works, observe that a subterm of (a reduct of) $M[x := X]$ will be $xN_1 \dots N_{i-1}(\lambda u_1 \dots u_n.u_j N'_1 \dots N'_m)[x := X]$. Now we get

$$\begin{array}{l} X\hat{N}_1 \dots \hat{N}_{i-1}(\lambda u_1 \dots u_n.u_j\hat{N}'_1 \dots \hat{N}'_m) \\ \longrightarrow\!\!\!\!\!\rightarrow_\beta a\hat{N}_1 \dots \hat{N}_{i-1}(\lambda u_1 \dots u_n.u_j\hat{N}'_1 \dots \hat{N}'_m)(U_j\hat{\hat{N}}'_1 \dots \hat{\hat{N}}'_m\Delta), \\ U_j\hat{\hat{N}}'_1 \dots \hat{\hat{N}}'_m\Delta \longrightarrow\!\!\!\!\!\rightarrow_\beta b\hat{\hat{N}}'_1 \dots \hat{\hat{N}}'_m\Delta(\Delta\Delta), \end{array}$$

where $\hat{N} = N[x := X]$ and $\hat{\hat{N}} = \hat{N}[u_j := U_j]$ (possibly with indexes and $'$).

v) *Case $x \equiv y$ and α not empty, while β empty.* Then $z \not\equiv x$ and $t \equiv x$. Assume that $\alpha = \langle i, j\rangle\alpha'$. Then M has a subterm of the shape $xM_1 \dots M_{i-1}(\lambda u_1 \dots u_j.M^*)$, such that the replacement path of the current occurrence of z in M^* is $u_j\alpha'$, and M^* has a subterm of the shape $zN_1 \dots N_{k-1}(\lambda v_1 \dots v_n.xN'_1 \dots N'_m)$. Notice that $z \equiv u_j$ if α' is empty. Let h be the maximum between i and $m + 1$ (so $h \geq 1$) and

$$X' \equiv \lambda w_1 \dots w_h.xw_1 \dots w_{h-1}(xw_h(w_i u_1 \dots u_j)).$$

[3] Following [6] (Definition 10.3.2) a substitution instance of a term P is the result of substituting some terms for some free variables in P.

On the normal form M' of $M[x := X']$ we can observe that:

- the occurrences which are adjacent in M remain adjacent in M';
- the variables which occur with replacement path $x\langle i, l\rangle\gamma$ where $1 \leq l \leq j$ in M occur also with replacement path $u_l\gamma$ in M'.

By the above observations there are adjacent occurrences of z, x in M' with replacement paths respectively $u_j\alpha'$ and x. Then induction hypothesis applies and we can find normal forms X'', U_j such that $M'[x := X'', u_j := U_j]$ does not reduce to a normal form. Notice that $\lambda v_1 \ldots v_n.xN'_1 \ldots N'_m[x := X']$ reduces to a normal form of the shape $\lambda v_1 \ldots v_n w_{m+1} \ldots w_h.xR_1 \ldots R_h$ for suitable normal forms $R_1, \ldots, R_h$. Since X'' will be built in the basic step, it will be

$$X'' \equiv \lambda r_1 \ldots r_{h+1}.br_1 \ldots r_{h+1}(r_{h+1}r_{h+1}).$$

This suggests us to choose

$$X \equiv \lambda s_1 \ldots s_{h+1}.xs_1 \ldots s_{h-1}(xs_hs_{h+1}(s_iu_1 \ldots u_{j-1}U_j)(s_{h+1}s_{h+1})).$$

To see why this works, observe that a subterm of (a reduct of) $M[x := X]$ will be $Z\hat{N}_1 \ldots \hat{N}_{k-1}(\lambda v_1 \ldots v_n.X\hat{N}'_1 \ldots \hat{N}'_m)$ where Z is a subterm of U_j and $\hat{N}$ is a substitution instance of N (possibly with indexes and $'$). Notice that Z is built in the basic step, and therefore

$$Z \equiv \lambda q_1 \ldots q_k.aq_1 \ldots q_k(q_kv_1 \ldots v_nw_{m+1} \ldots w_h\Delta).$$

Now if $R \equiv \lambda v_1 \ldots v_n.X\hat{N}'_1 \ldots \hat{N}'_m$ we get

$$\begin{aligned} &Z\hat{N}_1 \ldots \hat{N}_{k-1}R \twoheadrightarrow_\beta a\hat{N}_1 \ldots \hat{N}_{k-1}R(Rv_1 \ldots v_nw_{m+1} \ldots w_h\Delta),\\ &Rv_1 \ldots v_nw_{m+1} \ldots w_h\Delta \twoheadrightarrow_\beta x\hat{N}'_1 \ldots \hat{N}'_mw_{m+1} \ldots w_{h-1}(xw_h\Delta S(\Delta\Delta)), \end{aligned}$$

where S is a suitable term.

Notice that in all cases X and Y are normal forms. ■

The previous lemma suggests us to define a set of normal forms (the positive normal forms) which includes the set PNF of persistently normalising normal forms we want to characterise, as proved in Theorem A.15.

Definition A.14 (Positive normal forms) *A normal form M is a positive normal form ($M \in \mathsf{NF}^+$) if and only if*

i) the head variable of M is free (or equivalently negative) *ii) there are no adjacent occurrences of positive variables in M.* ■

.

Theorem A.15 *The persistently normalising normal forms are positive normal forms, i.e.* $\mathsf{PNF} \subseteq \mathsf{NF}^+$.

Proof. We show that if M does not belong to NF^+ then M does not belong to PNF. The easier case is when M does not belong to NF^+ since the head variable of M is bound. Let $M \equiv \lambda\vec{x}y\vec{z}.y\vec{N}$. Then clearly $M\vec{x}(\lambda\vec{t}u.u\Delta\vec{t})\vec{z}\Delta \twoheadrightarrow_\beta \Delta\Delta\vec{N}'$ where $\vec{t}$ has the same length as $\vec{N}$ and $\vec{N}' \equiv \vec{N}[y := \lambda\vec{t}u.u\Delta\vec{t}]$.

Otherwise there must be adjacent occurrences of positive variables z, t in M. Let $M \equiv \lambda\vec{x}.y\vec{N}$. From Definitions A.8 and A.11 we get that all positive variables of M are negative variables of $y\vec{N}$ and their replacement paths in $y\vec{N}$ start with one variable belonging to $\vec{x}$. Let $x_i\alpha, x_j\beta$ be the replacement paths of z, t in $y\vec{N}$ (possibly $i = j$). By Lemma A.13 there are normal forms X_i, X_j such that $y\vec{N}[x_i := X_i, x_j := X_j]$ (or, when $i = j$, one normal form X_i such that $y\vec{N}[x_i := X_i]$) does not reduce to normal form. Now by choosing

$$X_l = \begin{cases} X_i \text{ if } l = i \\ X_j \text{ if } l = j \\ x_l \quad \text{otherwise} \end{cases}$$

we get that $M\vec{X}$ does not reduce to normal form and this implies $M \notin \mathsf{PNF}$.
■

Example A.16 *If*

$$M \equiv x(\lambda y.a(y(\lambda z.b(x(\lambda t.c(\underline{zt}))))))),$$

then the underlined occurrences of the variables z, t are adjacent in M and their replacement paths in M are respectively $x\langle 1,1\rangle\langle 1,1\rangle$ and $x\langle 1,1\rangle$. Following the proof of Lemma A.13 we can consider the normal form M' of $M[x := X']$ where $X' \equiv \lambda u.xu(uy)$. We get

$$M' \equiv x(\lambda y.a(y(\lambda z.b(x(\lambda t.c(zt))(c(zy))))))(a(y(\lambda z.b(x(\lambda t.c(zt))(c(\underline{zy})))))).$$

The underlined occurrences of the variables z, y are adjacent in M' with replacement paths respectively $y\langle 1,1\rangle$ and y. Now the normal form of $M'[y := Y']$, where $Y' \equiv \lambda v.y(yv(vz))$, is

$$
\begin{aligned}
M'' \equiv\, & x(\lambda y.a(y(\lambda z.b(x(\lambda t.c(zt))(c(zy))))))\\
& (a(y(y(\lambda z.b(x(\lambda t.c(zt))(c(z(\lambda v.y(yv(vz))))))))\\
& (b(y(\lambda t.c(zt))(c(\underline{z}(\lambda v.\underline{y}(yv(vz)))))))))).
\end{aligned}
$$

In M'' the underlined occurrences of the variable z, y are adjacent with replacement paths z, y. So if we choose $Z \equiv \lambda w.aw(wv\Delta)$, $Y'' \equiv \lambda r_1 r_2.br_1r_2(r_2r_2)$ we get $M''[z := Z, y := Y'']$ which does not have a normal form. Now following the proof we build $Y \equiv \lambda s_1 s_2.y(ys_1s_2(s_1Z)(s_2s_2))$, and also $M''[y := Y]$ does not have a normal form. Lastly replacing Y to y in X' we obtain the term $X \equiv \lambda u.xu(uY)$ and one can check that the application of $\lambda x.M$ to X does not have a normal form. The crucial steps are:

$$
\begin{aligned}
&(\lambda x.M)X \twoheadrightarrow_\beta XN_1 \twoheadrightarrow_\beta xN_1(N_1Y) \text{ where } N_1 \equiv \lambda y.a(y(\lambda z.b(X(\lambda t.c(zt)))))\\
&N_1Y \twoheadrightarrow_\beta a(YN_2) \text{ where } N_2 \equiv \lambda z.b(X(\lambda t.c(zt)))\\
&YN_2 \twoheadrightarrow_\beta \lambda s_2.y(yN_2s_2(N_2Z)(s_2s_2))\\
&N_2Z \twoheadrightarrow_\beta b(X(\lambda t.c(Zt)))\\
&X(\lambda t.c(Zt)) \twoheadrightarrow_\beta xN_4(N_4Y) \text{ where } N_4 \equiv \lambda t.c(Zt)\\
&N_4Y \twoheadrightarrow_\beta c(ZY) \twoheadrightarrow_\beta c(aY(Yv\Delta))\\
&Yv\Delta \twoheadrightarrow_\beta y(yv\Delta(vZ)(\Delta\Delta)).
\end{aligned}
$$

Figure 7 shows the Böhm trees of M, M' and M''.

Theorem A.15 suggests to consider a proper subset of the polarised normal forms, i.e. the polarised normal forms not containing adjacent occurrences of positive variables. This can be obtained by the simple move of restricting the hypothesis in rule (+ *app*). We call them strongly polarised normal forms.

Definition A.17 (Strongly polarised normal forms) *The set of* strongly polarised normal forms, $\mathsf{SNF}^{\mathsf{i,j}}$ *is the subset of the set of polarised normal forms,* $\mathsf{NF}^{\mathsf{i,j}}$*, defined as follows:*

$$
(+\,appr)\ \frac{\vec{M} \in \mathsf{SNF}^{+,-}}{x^+\vec{M} \in \mathsf{SNF}^{+,+} \cap \mathsf{SNF}^{-,+}} \qquad (-\,app)\ \frac{\vec{M} \in \mathsf{SNF}^{-,+} \cup \mathsf{SNF}^{-,-}}{x^-\vec{M} \in \mathsf{SNF}^{+,-} \cap \mathsf{SNF}^{-,-}}
$$

$$
(+\,abs)\ \frac{M \in \mathsf{SNF}^{+,\mathsf{j}}}{\lambda x^+.M \in \mathsf{SNF}^{+,\mathsf{j}}} \qquad (-\,abs)\ \frac{M \in \mathsf{SNF}^{-,\mathsf{j}}}{\lambda x^-.M \in \mathsf{SNF}^{-,\mathsf{j}}} \quad \blacksquare
$$

Example A.18 *The λ-term $\lambda x^+y^+.a^-(x^+a^-)(\lambda z^-.z^-y^+)$ (see Figure 8) is a strongly polarised normal form. Instead, the principal decoration of the λ-term $\lambda v.x(\lambda t.a(t(\lambda uz.b(x(vv)(\lambda v.d(vz))))))$ (see Figure 5) is not a strongly polarised normal form since there are adjacent occurrences of the positive variable v^+.*

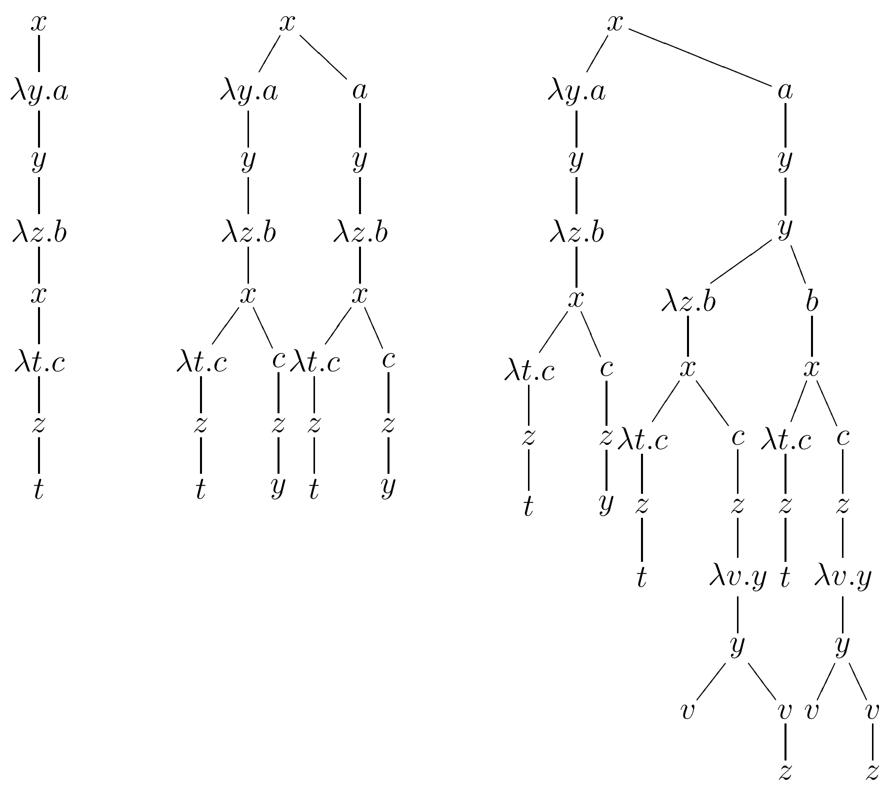

Fig. 7. Böhm trees of M, M', M'' as defined in Example A.16.

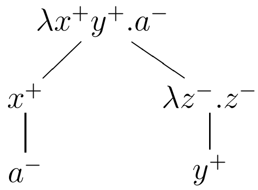

Fig. 8. Böhm tree of the strongly polarised normal form $\lambda x^+ y^+ . a^- (x^+ a^-)(\lambda z^- . z^- y^+)$.

Rule ($+$ *appr*) says that we can apply a positive variable only to normal forms whose head variable is negative. I.e. in a strongly polarised normal form we cannot have two adjacent occurrences of positive variables.

Remark A.19 *In the Böhm tree of a strongly polarised normal form all sons of a node whose head variable is positive have negative head variables.*

It is clear that not all principal decorations of normal forms are strongly polarised normal forms, take as an example $\lambda x.xx$. But we can easily see that:

Proposition A.20 *The positive normal forms are exactly the normal forms whose principal decorations belong to* $\mathsf{SNF}^{+,-}$.

We can prove that strongly polarised normal forms can be typed starting from arbitrary types not containing Ω (modulo $\sim_\bigtriangledown$) for positive variables, whenever $\Sigma^\bigtriangledown$ is an arrow type theory. Moreover under the same hypothesis the strongly polarised normal forms which are principal decorations of positive normal forms can be typed with an arbitrary type not containing Ω modulo $\sim_\bigtriangledown$ (Theorem A.23). First we need to show a property of arrow type theories (Lemma A.22).

We associate to each type the minimum number of external arrows.

Definition A.21 *Let $\Sigma^\bigtriangledown$ be a type theory. The mapping $|\ | : \mathbf{T}^\bigtriangledown \to I\!N$ is defined inductively on types as follows:*

$$\begin{array}{lll} |A| & = 0 & \textit{if } A \in \mathbf{C}^\bigtriangledown; \\ |A \to B| & = 1 + |B|; & \\ |A \cap B| & = \min\{|A|, |B|\}. \blacksquare & \end{array}$$

Lemma A.22 *Let $\Sigma^\bigtriangledown$ be an arrow type theory. For each $A \in \mathbf{T}^\bigtriangledown$ and for each integer n:*

(i) there is $A' \in \mathbf{T}^\bigtriangledown$ such that $A' \sim_\bigtriangledown A$ and $|A'| \geq n$;
(ii) there is $A' \in \mathbf{T}^\bigtriangledown$ such that $A' \sim_\bigtriangledown A$ and $A' \equiv \bigcap_{i \in I}(\vec{B}_i \to C_i)$ where $\vec{B}_i$ has length n for all $i \in I$.

Proof.

(i) It is enough to show that for each $A \in \mathbf{T}^\bigtriangledown$ we can find $A' \in \mathbf{T}^\bigtriangledown$ such that $A' \sim_\bigtriangledown A$ and $|A'| > |A|$. The proof is by induction on A. The case $A \in \mathbf{C}^\bigtriangledown$ follows immediately from the definition of arrow type theory. If $A \equiv B \to C$ then by induction there is $C' \in \mathbf{T}^\bigtriangledown$ such that $C' \sim_\bigtriangledown C$ and $|C'| > |C|$. We can choose $A' \equiv B \to C'$. If $A \equiv B \cap C$ then by induction there are $B', C' \in \mathbf{T}^\bigtriangledown$ such that $B' \sim_\bigtriangledown B$, $C' \sim_\bigtriangledown C$ and $|B'| > |B|$, $|C'| > |C|$. We can choose $A' \equiv B' \cap C'$.

(ii) By (i) it suffices to show that for each $A \in \mathbf{T}^\bigtriangledown$ with $|A| \geq n$ there is $A' \in \mathbf{T}^\bigtriangledown$ such that $A' \sim_\bigtriangledown A$ and $A' \equiv \bigcap_{i \in I}(\vec{B}_i \to C_i)$ where $\vec{B}_i$ has length n for all $i \in I$. The proof is by induction on A. The case $A \in \mathbf{C}^\bigtriangledown$ is trivial since $n = 0$. If $A \equiv B \to C$ then $|C| \geq n-1$. By induction there is $C' \in \mathbf{T}^\bigtriangledown$ such that $C' \sim_\bigtriangledown C$ and $C' \equiv \bigcap_{i \in I}(\vec{D}_i \to E_i)$ where $\vec{D}_i$ has length $n-1$ for all $i \in I$. We can choose $A' \equiv \bigcap_{i \in I}(B \to \vec{D}_i \to E_i)$, since $A' \sim_\bigtriangledown A$ by rules ($\to$-$\cap$) and (η). The case $A \equiv B \cap C$ is easy by induction. $\blacksquare$

Theorem A.23 *Let $\Sigma^\bigtriangledown$ be an arrow type theory. Let $M \in \mathsf{SNF}^{\mathsf{i,j}}$ and let $\vec{x^+}$*

and $\vec{y^-}$ be the positive and negative variables which occur free in M:

i) if $\mathsf{i} = +$ and $\mathsf{j} = -$ then for all types $\vec{A}$ with $\Omega \notin_\triangledown \vec{A}$ and for all types A with $\Omega \notin_\triangledown A$ there exist types $\vec{B}$ with $\Omega \notin_\triangledown \vec{B}$ such that $\vec{x^+} : \vec{A}, \vec{y^-} : \vec{B} \vdash^\triangledown M : A$.

ii) otherwise for all types $\vec{A}$ with $\Omega \notin_\triangledown \vec{A}$ there exist types $\vec{B}$ with $\Omega \notin_\triangledown \vec{B}$ and a type A with $\Omega \notin_\triangledown A$ such that $\vec{x^+} : \vec{A}, \vec{y^-} : \vec{B} \vdash^\triangledown M : A$.

Proof. We prove (i) and (ii) simultaneously by induction on the structure of strongly polarised normal forms. We convene that all considered types do not contain occurrences of Ω modulo $\sim_\triangledown$. By x^+ we denote an arbitrary element of $\vec{x}$. Similarly for y^-.

(i) If $M \in \mathsf{SNF}^{+,-}$ then M is of the shape $\lambda\vec{z^+}.y^-\vec{N}$ where $\vec{N} \in \mathsf{SNF}^{-,+} \cup \mathsf{SNF}^{-,-}$. Since $\Sigma^\triangledown$ is an arrow type theory, then by Lemma A.22(ii) each type is equivalent to an intersection of arrow types, each one of the shape $\vec{C} \to D$ where the length of $\vec{C}$ is an arbitrary integer. So it suffices to prove that M has all types of the shape $\vec{C} \to D$, where $\vec{C}$ has the length of $\vec{z}$. By the induction hypothesis (ii) there are types $\vec{B}$ and $\vec{E}$ such that for all types $\vec{A}$ and $\vec{C}$ we have: $\vec{x^+} : \vec{A}, \vec{z^+} : \vec{C}, \vec{y^-} : \vec{B} \vdash^\triangledown \vec{N} : \vec{E}$. Now let Γ be the $\triangledown$-basis obtained by adding the premise $y^- : \vec{E} \to D$ to $\vec{x^+} : \vec{A}, \vec{z^+} : \vec{C}, \vec{y^-} : \vec{B}$. We get $\Gamma \vdash^\triangledown y^-\vec{N} : D$ and we can conclude using rule $(\to \mathrm{I})$.

(ii) If $M \in \mathsf{SNF}^{-,-}$ then M is of the shape $\lambda\vec{z^-}.t^-\vec{N}$ where $\vec{N} \in \mathsf{SNF}^{-,+} \cup \mathsf{SNF}^{-,-}$ and $t^- \in \vec{y^-} \cup \vec{z^-}$. By the induction hypothesis (ii) there are types $\vec{B}, \vec{C}$ and $\vec{E}$ such that for all types $\vec{A}$ we get: $\vec{x^+} : \vec{A}, \vec{z^-} : \vec{C}, \vec{y^-} : \vec{B} \vdash^\triangledown \vec{N} : \vec{E}$. Now let Γ be the $\triangledown$-basis obtained by adding the premise $t^- : \vec{E} \to D$, where D is arbitrary, to $\vec{x^+} : \vec{A}, \vec{z^-} : \vec{C}, \vec{y^-} : \vec{B}$. We get $\Gamma \vdash^\triangledown t^-\vec{N} : D$ and we can conclude using rule $(\to \mathrm{I})$.

If $M \in \mathsf{SNF}^{-,+}$ then M is of the shape $\lambda\vec{z^-}.x^+\vec{N}$ where $\vec{N} \in \mathsf{SNF}^{+,-}$. Let A' be the type of the variable x^+ and n the length of $\vec{N}$. By Lemma A.22(ii) there is a type $\vec{E} \to D$ such that $\vec{E}$ has length $\geq n$ and $A' \leq_\triangledown \vec{E} \to D$. By the induction hypothesis (i) there are types $\vec{B}$ and $\vec{C}$ such that for all types $\vec{A}$ and $\vec{E}$ we have: $\vec{x^+} : \vec{A}, \vec{z^-} : \vec{C}, \vec{y^-} : \vec{B} \vdash^\triangledown \vec{N} : \vec{E}$. We get $\vec{x^+} : \vec{A}, \vec{z^-} : \vec{C}, \vec{y^-} : \vec{B} \vdash^\triangledown x^+\vec{N} : D$ and we can conclude using rule $(\to \mathrm{I})$.

If $M \in \mathsf{SNF}^{+,+}$ then M is of the shape $\lambda\vec{z^+}.t^+\vec{N}$ where $\vec{N} \in \mathsf{SNF}^{+,-}$ and $t^+ \in \vec{x^+} \cup \vec{z^+}$. If $t^+ \in \vec{x^+}$ the proof goes as in previous case. Otherwise the proof is similar, since we can assume $t^+ : \vec{E} \to D$, where $\vec{E}$ has the length of $\vec{N}$, for arbitrary types $\vec{E}, D$ and conclude as in previous case. ∎

Proof of Theorem 3.2(2.i)($\Rightarrow$). The theory of polarised normal forms has been introduced to get this result. If $M \in \mathsf{PN}$ then by definition its normal form $M' \in \mathsf{PNF}$. By Theorem A.15 $M' \in \mathsf{NF}^+$, so M' has all types not containing Ω modulo $\sim_\bigtriangledown$(from $\bigtriangledown$-bases not containing Ω modulo $\sim_\bigtriangledown$) in an arbitrary arrow type theory by Proposition A.20 and Theorem A.23. We can conclude that also M has the same types in an arbitrary arrow type theory by Theorem 1.16(ii). ∎

Remark A.24 *By Theorems 3.2(2.i)($\Leftarrow$), A.23, and Proposition A.20 we get* $\mathsf{NF}^+ \subseteq \mathsf{PNF}$. *Therefore from Theorem A.15 we can conclude that the persistently normalising normal forms are exactly the positive normal forms, i.e.* $\mathsf{PNF} = \mathsf{NF}^+$.

References

[1] S. Abramsky. Domain theory in logical form. *Ann. Pure Appl. Logic*, 51(1-2):1–77, 1991.

[2] S. Abramsky and C.-H. L. Ong. Full abstraction in the lazy lambda calculus. *Inform. and Comput.*, 105(2):159–267, 1993.

[3] R. M. Amadio and P.-L. Curien. *Domains and Lambda-calculi.* Cambridge University Press, Cambridge, 1998.

[4] S. van Bakel. Complete restrictions of the intersection type discipline. *Theoret. Comput. Sci.*, 102(1):135–163, 1992.

[5] H. Barendregt, M. Coppo, and M. Dezani-Ciancaglini. A filter lambda model and the completeness of type assignment. *J. Symbolic Logic*, 48(4):931–940, 1983.

[6] H.P. Barendregt. *The Lambda Calculus: its Syntax and Semantics.* North-Holland, Amsterdam, revised edition, 1984.

[7] H.P. Barendregt and et. al. *Typed λ-calculus and Applications.* North-Holland, 200X. (to appear).

[8] C. Berline. From computation to foundations via functions and application: The λ-calculus and its webbed models. *Theoret. Comput. Sci.*, 249:81–161, 2000.

[9] C. Böhm and M. Dezani-Ciancaglini. λ-terms as total or partial functions on normal forms. In C. Böhm, editor, *λ-calculus and Computer Science Theory*, volume 37 of *Lecture Notes in Computer Science*, pages 96–121, Berlin, 1975. Springer.

[10] M. Coppo and M. Dezani-Ciancaglini. An extension of the basic functionality theory for the λ-calculus. *Notre Dame J. Formal Logic*, 21(4):685–693, 1980.

[11] M. Coppo, M. Dezani-Ciancaglini, and M. Zacchi. Type theories, normal forms, and D_∞-lambda-models. *Inform. and Comput.*, 72(2):85–116, 1987.

[12] M. Coppo, F. Honsell, M. Dezani-Ciancaglini, and G. Longo. Extended type structures and filter lambda models. In G.Lolli, G.Longo, and A.Marcja, editors, *Logic Colloquium '82*, pages 241–262. North-Holland, Amsterdam, 1984.

[13] M. Dezani-Ciancaglini and S. Ghilezan. Two behavioural lambda models. In H.Geuvers and F.Wiedijk, editors, *Types for Proofs and Programs*, volume 2646 of *Lecture Notes in Computer Science*, pages 127–147, Berlin, 2003. Springer-Verlag.

[14] M. Dezani-Ciancaglini, F. Honsell, and F. Alessi. A complete characterization of the complete intersection-type preorders. *ACM Trans. On Comput. Logic*, 1(4):120–147, 2003.

[15] M. Dezani-Ciancaglini, F. Honsell, and Y. Motohama. Compositional characterization of λ-terms using intersection types. In M. Nielsen and B. Rovan, editors, *Mathematical Foundations of Computer Science 2000*, volume 1893 of *Lecture Notes in Computer Science*, pages 304–313. Springer, 2000.

[16] L. Egidi, F. Honsell, and S. Ronchi della Rocca. Operational, denotational and logical descriptions: a case study. *Fund. Inform.*, 16(2):149–169, 1992.

[17] S. Ghilezan. Strong normalization and typability with intersection types. *Notre Dame J. Formal Logic*, 37(1):44–52, 1996.

[18] F. Honsell and M. Lenisa. Semantical analysis of perpetual strategies in λ-calculus. *Theoret. Comput. Sci.*, 212(1-2):183–209, 1999.

[19] F. Honsell and S. Ronchi della Rocca. An approximation theorem for topological lambda models and the topological incompleteness of lambda calculus. *J. Comput. System Sci.*, 45(1):49–75, 1992.

[20] J.-L. Krivine. *Lambda-calcul Types et Modèles*. Masson, Paris, 1990. English translation: Lambda-calculus, types and models, Ellis Horwood, Crystal City, 1993.

[21] D. Leivant. Typing and computational properties of lambda expressions. *Theoret. Comput. Sci.*, 44(1):51–68, 1986.

[22] J. Mitchell. *Foundations for Programming Languages*. MIT Press, Boston, 1996.

[23] D. Park. The **Y**-combinator in Scott's λ-calculus models (revised version). Theory of Computation Report 13, Department of Computer Science, University of Warwick, 1976.

[24] G. D. Plotkin. Set-theoretical and other elementary models of the λ-calculus. *Theoret. Comput. Sci.*, 121(1-2):351–409, 1993.

[25] G. Pottinger. A type assignment for the strongly normalizable λ-terms. In J.R. Hindley and J.P. Seldin, editors, *To H. B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism*, pages 561–577. Academic Press, London, 1980.

[26] D. Scott. Continuous lattices. In F.W. Lawvere, editor, *Toposes, Algebraic Geometry and Logic*, volume 274 of *Lecture Notes in Mathematics*, pages 97–136. Springer, Berlin, 1972.

[27] P. Severi. *Normalisation in Lambda calculus and its Relation to Type Inference.* Eindhoven University of Technology, 1996. Ph D. thesis.