# Non deterministic extensions of untyped $\lambda$-calculus*

*Ugo de'Liguoro and Adolfo Piperno*
Dipartimento di Scienze dell'Informazione
Università di Roma "La Sapienza"
Via Salaria 113, I-00198 Roma
{deliguoro,piperno}@dsi-next1.ing.uniroma1.it

**Abstract**

The main concern of this paper is the study of the interplay between functionality and non determinism. Indeed the first question we ask is whether the analysis of parallelism in terms of sequentiality and non determinism, which is usual in the algebraic treatment of concurrency, remains correct in presence of functional application and abstraction.

We identify non determinism in the setting of $\lambda$-calculus with the absence of the Church-Rosser property plus the inconsistency of the equational theory obtained by the symmetric closure of the reduction relation.

We argue in favour of a distinction between non determinism and parallelism, due to the conjunctive nature of the former in contrast to the disjunctive character of the latter. This is the basis of our analysis of the operational and denotational semantics of non deterministic $\lambda$-calculus, which is the classical calculus plus a choice operator, and of our election of bounded indeterminacy as the semantical counterpart of conjunctive non determinism. This leads to operational semantics based on the idea of *must* preorder, coming from the the classical theory of solvability and from the theory of process algebras. To characterize this relation, we build a model using the inverse limit construction over non deterministic algebras, and we prove it fully abstract using a generalization of Böhm trees. We further prove conservativity theorems for the equational theory of the model and for other theories, related to non deterministic $\lambda$-calculus, with respect to classical $\lambda$-theories.

**Keywords**: Lambda-calculus, non determinism, full abstraction, Böhm trees, non deterministic algebras, applicative structures, conservativity.

*This work has been partially supported by grants from ESPRIT-BRA 3230. An extended abstract of this paper appeared in the Proceedings of CAAP'92, J.-C.Raoult (Ed.), LNCS 581, Springer Verlag 1992.

# Contents

# 1 Introduction

Multivalued functions are commonly considered as natural candidates to model the input-output behaviour of non deterministic computing devices. Plotkin and Smyth investigated this subject introducing the theory of powerdomains [43, 49]. The aim of the present paper is to give a calculus of computable multifunctions comparable to the pure lambda calculus, and then to provide a model for it based on the theory of powerdomains and non deterministic algebras.

The first contributions exploring non deterministic extensions of functional languages [5, 6] are based on Plotkin's PCF [44], which is a typed lambda calculus with constants and fixed point combinator. The proposed mathematical model however involves the elementary part of powerdomain theory only, in that just basic types are interpreted in [5] as powerdomains of flat cpo's, being functional types modeled by spaces of linear (that is additive) functions. This also causes troubles when comparing operational and denotational semantics. Introducing constraints to the calculus, Astesiano and Costa propose [6] to solve such a mismatch in a way which is not, in our opinion, completely satisfactory.

Further proposals of both typed and untyped lambda calculi extended with choice and even parallel operators are [14, 12, 15, 16, 29, 40, 47]. The concepts of non deterministic and parallel computation are often considered jointly, reflecting the prevalent position of considering non determinism as an abstract concept explaining the informal notion of parallelism. Historically this seems to arise from the idea of non deterministic Turing machine considered as an abstraction of some kind of parallel machine. Moreover the same reductionistic attitude is at the basis of the interleaving semantics as it is used in the process algebra approach to concurrency theory [36, 27, 22, 7], and directly inspires Plotkin's resumption semantics of parallelism in [43, 25].

The criticism against this approach is hardly new, mainly in the field of concurrency theory, where it motivates the so called "true concurrency" semantics. In the applicative setting Boudol [15] explicitly proposes non determinism (together with a convergency test) as a tool for extending the $\lambda$-calculus to make Plotkin's parallel or [44, 9] definable. Successively ([16]) the same author reverses his proposal resorting to a different combinator which does not allow any choice. A question naturally arises here: is the analysis of parallelism in terms of sequentiality and non determinism correct for the applicative context? To answer this, we need to make the concept of non determinism precise in rewriting systems, without resorting to the crude idea coming from elementary automata theory.

These considerations, which are developed in section 2, led us to consider absence of the Church-Rosser property not sufficient to classify a rewriting system as non deterministic. What we consider a fundamental feature of non deterministic rewriting systems is inconsistency of the equational theory obtained by the symmetric closure of their reduction relation. The intuition behind such requirement comes immediately out of the possibility of getting different values starting from a single datum: equating veritably different values naturally leads to inconsistency.

Moving from this notion, we argue for a strict distinction between non determinism and parallelism in the presence of functional application, and then we attribute a conjunctive nature to non determinism and a disjunctive character to parallelism (for similar ideas, but in a different context, see e.g. [46]). Consequently, we give a negative answer to the question about reducibility of parallelism to non determinism, at least within the framework of applicative languages.

Now, the main concern of this paper remains the analysis of the interplay between functionality and non determinism. To this aim, we introduce the non deterministic $\lambda$-calculus to be the classical calculus extended with a choice operator. The mentioned conjunctive nature of non determinism provides the basis for the analysis of the operational and denotational semantics of our calculus, which requires bounded indeterminacy. This has technical advantages; from [4] we know that unbounded countable non determinism leads to loss of continuous least fix point, fully abstract semantics using standard domain theory (but see [41] for continuous models of unbounded indeterminacy in Lehman's categorical setting). We are able to build a model which is fully abstract with respect to operational semantics based on our notion of must-solvability and head reduction; moreover the theory of this model turns out to be a conservative extension of the maximally consistent $\lambda$-theory equating all unsolvable terms, called $\mathcal{H}^*$ in [9].

Our results improve on [5, 6], in that we do not commit ourselves with additive semantics of functional application, facing the more general problem of finding a model of non deterministic type free $\lambda$-calculus. For purely syntactical aspects we get inspiration from [48], which to our knowledge remains the first work on non deterministic extensions of type free $\lambda$-calculus. Our main improvement on this work is that we base our study entirely on the reduction relation; we do not postulate any equation which could be considered reasonable among terms of our calculus: we justify Sharma's theories on the basis of our operational, algebraic and denotational semantics, giving a clean perspective to look at the relation between the non deterministic theories and the classical ones, culminating in the conservativity theorems.

Other related works follow Abramsky's proposal of lazy $\lambda$-calculus and domain logic [2]. This is the case of [15, 16, 29, 40], while [47] is concerned with a more operational analysis of lazy non deterministic $\lambda$-calculi, based on a translation into Milner's $\pi$-calculus [35, 37]. We consider the topic treated in this paper also relevant with respect to the extensions of process algebras with functional application and abstraction, as pursued by Thomsen's CHOCS [50] and, at least implicitly, by Milner's full $\pi$-calculus, that is with choice operator, which allow encodings of the $\lambda$-calculus.

The paper is organized as follows: we discuss the nature of non determinism in section 2 and there we motivate our basic choices. In section 3 we introduce the operational semantics of the calculus, based on a generalization of the classical notion of solvability. To study this semantics we introduce non deterministic unfolding trees which generalize classical Böhm trees and show that the preorder on which the operational semantics is based is included in the preorder induced by a suitable tree inclusion (semiseparability theorem). Such a notion is further investigated in section 4 from a denotational point of view, leading us to the introduction of a notion of model of non deterministic type free $\lambda$-calculus and (section 5) to a full abstraction theorem equating orders induced by the operational and tree preorders to the semantical order. We finally investigate in section 6 non deterministic theories with respect to their conservativity property on $\lambda$-theories. To avoid detours through complicated details, some technical proofs are collected in section 7.

## 2 The Non Deterministic $\lambda$-calculus

Non determinism and parallelism are often related concepts. As an example, all approaches to concurrency theory which are based on process algebras (see e.g. [36, 22, 27, 7]) rely on the analysis of parallelism in terms of sequentiality and non determinism, which is at the heart of interleaving semantics. A paradigmatic result in this setting is Milner's expansion theorem for CCS.

The question we ask here is whether such analysis remains valid in the context of applicative languages. As a first step we need to clarify the concept of non determinism in the framework of rewriting systems, which usually are the basis for the operational semantics of these languages.

In automata theory non determinism arises when more than one transition from the same state is possible. This definition does not apply to rewriting systems, since they can enjoy Church-Rosser property. This implies that every term has at most one normal form; since a normal form can be seen as the value of a term, its unicity allows for a functional, and hence deterministic interpretation. Consequently we should first ask for the absence of such unicity property when speaking of non determinism.

In our opinion, this requirement does not suffice. Term rewriting as well as combinatory rewriting systems [30] are usually seen as the asymmetric versions of equational calculi, which can be recovered taking the symmetric closure of the relation $\xrightarrow{*}$: we call such an equational theory the derived theory of the system. However, as it is argued in

[33], the rewriting paradigm is more general than the equational one, in that there are non trivial rewriting systems whose derived equational theories are inconsistent.

We believe this inconsistency is a typical feature of non determinism, which intuitively consists in the possibility of getting different values starting from a single datum: equating values which are actually different naturally leads to inconsistency. Now, from [30] (theorem 1.2.10 of chapter III), we know that untyped $\lambda$-calculus with surjective pairing is not Church-Rosser; its equational theory is nevertheless consistent (see e.g. [31]). Hence the lack of Church-Rosser property is not sufficient for being non deterministic in such an abstract sense.

As a tentative definition *we shall consider non deterministic any rewriting system which is not Church-Rosser and whose derived equational theory, given by the reflexive transitive and symmetric closure of the reduction relation, is inconsistent.*

The easiest way to get a non deterministic system is using erratic choice. Let $\oplus$ be a binary operator, and consider the following rewriting rules:

$$t \oplus t' \longrightarrow t \quad \text{and} \quad t \oplus t' \longrightarrow t'$$

where $t$ and $t'$ are arbitrary terms (more formally one should use variables as in term rewriting systems or metavariables as in $\lambda$-calculus and related systems). Then some basic questions readily arise: what kind of non determinism are we dealing with? Is it related to some notion of parallelism?

To answer these questions we resort to process algebras, where equational theories over terms are built out of equivalence relations defined over their behaviours, namely sequences of actions, as defined by a set of transition rules. In [19] one of these equivalences has been introduced (see also [22]), relying on a family of tests based on some intuition of what is observable of the behaviour of a process, and a notion of passing a test. Then two processes will be equivalent if they pass the "same" set of tests. Now, in a deterministic system this would be sufficient since given a process term $p$ and a test $e$ one would have just that $p$ passes $e$ or not. But in presence of non determinism both that $p$ passes $e$ and that $p$ fails to pass $e$ may well happen. It follows that at least two notions of testing arise, namely all transition sequences starting with $p$ will satisfy $e$ ($p$ must $e$); or at least one such sequence satisfies $e$ ($p$ may $e$).

Recent studies about laziness and $\lambda$-calculus in [2] have shown an elegant way to relate observational relations over process algebras to combinatory rewriting systems, using the notion of convergency. In such setting any (closed) $\lambda$-term $M$ is convergent iff it reduces, under lazy reduction relation, to something of the form $\lambda x.N$, called *weak normal form*: if such a weak normal form exists for $M$ write $M\!\downarrow$. Now a test is any context $C[\ ]$, and we can say that $M$ passes $C[\ ]$ iff $C[M]\!\downarrow$. This induces the following equivalence

$$M \simeq N \;\Leftrightarrow\; \forall C[\,].\, C[M]\!\downarrow \,\Leftrightarrow\; C[N]\!\downarrow .$$

Now what is "observable" in an applicative setting? Just termination of reduction(s) and the value(s) in which it terminates. It consequently does not seem improper to speak of observational equivalence and testing (even if Abramsky and Ong use "applicative bisimulation" to name their equivalence relation).

Although Abramsky's notion of convergent term is not the classical one, it is strictly related to the notion of solvable term in classical $\lambda$-calculus: indeed a pure $\lambda$-term $M$ is *solvable* iff it reduces to a *head normal form*, that is to something of the shape $\lambda x_1 \dots x_m.xM_1 \dots M_n$ (see [9]). We know further that given any term there exists at most one head normal form which can be reached by *head reduction* only (see Def. 2.3 below). Substituting the notion of solvable for that of being weakly normalizable, and using the standardization theorem to avoid restriction to head reduction only (see [9] and subsection 2.2 of this paper) the equivalence $\simeq$ introduced above is exactly the equality of the $\lambda$-theory $\mathcal{H}^*$ (see [9]).

Now this "testing" perspective is at the heart of classical operational semantics of the full $\lambda$-calculus in [39, 28, 51], as it is presented in Barendregt's book [9]. This means that the crucial point is what we actually take as a value: in case of [2] the value is a weak normal form, that is something irreducible under lazy reduction relation. In case of the classical $\lambda$-calculus a value is something which is irreducible under head reduction. In effect $M$ does not have such a value if and only if it has no influence on the converging behaviour of any term containing it, that is for any context $C[\,]$ and term $N$, if $C[M]$ has a normal form, then $C[N]$ has (the same) normal form. (see Barendregt's Genericity Lemma, in [9] §14.3). Note that a term $M$ can be solvable even if it has no normal form, so that solvability gives us a subtler and far more abstract (namely functional) notion of convergency. Henceforth we shall refer to the classical setting.

It is time to be more formal. We introduce the syntax of our calculus and the relative notion of reduction.

**Definition 2.1** *Let $Var$ be any denumerable set of variables. The set $\Lambda_\oplus$ of the terms of the non deterministic $\lambda$-calculus is the least set s.t.*

*i)* $x \in \Lambda_\oplus$ *for all* $x \in Var$;

*ii)* $M, N \in \Lambda_\oplus \Rightarrow (MN) \in \Lambda_\oplus$;

*iii)* $M \in \Lambda_\oplus, x \in Var \Rightarrow (\lambda x.M) \in \Lambda_\oplus$;

*iv)* $M, N \in \Lambda_\oplus \Rightarrow M \oplus N \in \Lambda_\oplus$.

*The set of closed terms will be denoted by* $\Lambda^0_\oplus$.

The terms are considered up to renaming of bound variables; clearly the set $\Lambda$ of classical $\lambda$-terms is a subset of $\Lambda_\oplus$.

**Definition 2.2** *Let $M[N/x]$ denote the simultaneous substitution of each occurrence of $x$ by an occurrence of $N$ in $M$, up to renaming of bound variables in $M$ to avoid variable clashes; then $\longrightarrow \subseteq \Lambda_\oplus \times \Lambda_\oplus$ is the least relation such that*

($\beta$) $(\lambda x.M)N \longrightarrow M[N/x]$;

($\mu$) $N \longrightarrow N' \Rightarrow MN \longrightarrow MN'$;

($\nu$) $M \longrightarrow M' \Rightarrow MN \longrightarrow M'N$;

($\xi$) $M \longrightarrow M' \Rightarrow \lambda x.M \longrightarrow \lambda x.M'$;

($\oplus$1) $M \oplus N \longrightarrow M$, $M \oplus N \longrightarrow N$;

($\oplus$2) $M \longrightarrow M' \Rightarrow M \oplus N \longrightarrow M' \oplus N,\ N \oplus M \longrightarrow N \oplus M'$.

*As usual $\stackrel{*}{\longrightarrow}$ is the reflexive and transitive closure of $\longrightarrow$.*

## 2.1 Must-solvability

If we consider our present non deterministic extension of the full $\lambda$-calculus by adding erratic choice, we are naturally led to two different notions of convergency, as it happens with process algebras, namely *must* and *may* convergency (see in the lazy perspective [29, 40]). It follows that we are given two testing semantics: in a first one we shall consider definitely divergent any term having a non terminating head reduction; this is "demonic" non determinism. In a second one we are satisfied if at least one reduction leads to a value, hence divergency would mean that any head reduction does not terminate; this is "angelic" non determinism.

Resuming our original question about reducibility of parallelism to non determinism in the context of functional languages, we ask whether may or must semantics will positively solve the problem.

If the idea of parallelism we have in mind is a kind of or-parallelism, in which we compute in parallel whenever an alternative is met and we don't know which is the right choice to do, then the answer seems, at first glance, affirmative. For we are satisfied if any of multiple parallel computations succeedes in getting a value, we will commit ourself with "angelic" non determinism, so that the following should obtain (see [15]:

$$(M \oplus N)\!\downarrow\ \Leftrightarrow\ M\!\downarrow \vee\ N\!\downarrow .$$

What could be the operational idea behind this? If we meet a choice $M \oplus N$, we have to start parallel computations of $M$ and $N$, deferring the choice between them until either from $M$ or from $N$ a value is reached. Of course it wouldn't be effective to decide in advance which term will converge, but it seems harmless to suppose that we will be able to detect that we have actually reached a value on one side or the other.

But is it possible to delay the choice in all cases? The answer is surely not. Let $M$ be such that $M \xrightarrow{*} \lambda x.x\omega\omega$, and $N$ such that $N \xrightarrow{*} \lambda x.x\mathbf{I}$, where $\mathbf{I} \equiv \lambda x.x$ and $\omega \equiv \lambda x.xx$; then the value of $M \oplus N$ will be $\lambda x.x\omega\omega$ or $\lambda x.x\mathbf{I}$ depending on the reduction strategy we used, that is, say, on the relative speed with which $M$ and $N$ converge. But in the reduction of $(M \oplus N)\mathbf{I}$ a choice must occur before applying the reduct of $M$ or of $N$ to $\mathbf{I}$. On the other hand we expect that $(M \oplus N)\mathbf{I}$ evaluates to $(\lambda x.x\mathbf{I})\mathbf{I}$, that is to $\mathbf{I}$, and not to $(\lambda x.x\omega\omega)\mathbf{I}$ and hence to $\omega\omega$ (that is to a typical divergent combinator). Indeed when dealing with disjunctive non determinism we ask for an effective method picking up a value if any, as it happens with deterministic simulations of non deterministic Turing machines. Now there is no effective way to guarantee that the right choice will be the actual one since the convergency predicate is not recursive. We conclude this informal argument observing that no effective normalizing strategy does exist when we allow (internal) choice operators.

This shows that what we really have to do is to avoid any choice at all; this is exactly what happens in [16], which presents a Church-Rosser calculus, which isn't non deterministic anymore. This phenomenon does not happen by chance: what the author is looking for in [16] is an analysis of Plotkin's parallel or, which is by no means a multifunction, but simply a non stable function (see [9, 11]).

Consequently, we feel that the answer to our question about reducibility of parallelism to non determinism inside the framework of applicative languages has to be negative.

If not for parallelism, what is non determinism useful for? It seems unreasonable to think of erratic choice as an operator which actually "does" something for us. Our idea is to look at $\oplus$ as a kind of declarative operator. Its meaning, being that of an internal choice, is that we are willing to abstract away from some unpredictable events (as relative reduction speeds above) in the computation. We will accept whatever of two (or more) alternatives will be the actual one, provided that a minimum is assured to us: that in any case the computation will eventually give a meaningful value, if any, that is it will converge.

From our discussion it should be clear that the semantics of erratic choice is no longer "angelic" but "demonic" non determinism, thus the following has to obtain:

$$(M \oplus N)\!\downarrow\; \Leftrightarrow\; M\!\downarrow \wedge\, N\!\downarrow .$$

Let us introduce the must convergency and must preorder predicates in a more formal way.

**Definition 2.3**

*i) A term $M$ is a* head normal form *iff* $M \equiv \lambda x_1 \ldots x_m.xM_1 \ldots M_n$*, where* $n, m \geq 0$*;* HNF *is the set of head normal forms.*

*ii) If* $M \equiv \lambda x_1 \ldots x_n.\underline{(\lambda y.P)Q}M_1 \ldots M_m$ *or* $M \equiv \lambda x_1 \ldots x_n.\underline{(P \oplus Q)}M_1 \ldots M_m$*, then the underlined subterm is called the* head redex *of* $M$*.*

*iii) If* $\Delta$ *is the head redex of* $M$*, then* $M \longrightarrow_h N$ *iff* $N$ *results from* $M$ *by contracting* $\Delta$ *(*head reduction*).*

*iv)* $M \longrightarrow_i N \;\Leftrightarrow\; M \longrightarrow N \wedge M \not\longrightarrow_h N$ *(*internal reduction*).*

*v)* $N$ *is a* head normal form *of* $M$ *iff* $N \in$ HNF *and* $M \xrightarrow{*} N$*;* $N$ *is a* principal head normal form *of* $M$ *iff* $N$ *is a head normal form of* $M$ *and* $M \xrightarrow{*}_h N$*.*

**Definition 2.4** *For* $M \in \Lambda_\oplus$ *define*

*i)* $M{\downarrow_{must}} \Leftrightarrow M$ *has no infinite head reduction,*

*ii)* $M \sqsubseteq_{must} N \Leftrightarrow \forall C[\,].\; C[M]{\downarrow_{must}} \Rightarrow C[N]{\downarrow_{must}}$*,*

*iii)* $M \simeq_{must} N \Leftrightarrow M \sqsubseteq_{must} N \sqsubseteq_{must} M$*.*

*We abbreviate* $M{\downarrow_{must}}$ *with* $M{\downarrow}$ *and we write* $M{\uparrow}$ *to mean not* $M{\downarrow}$*.*

It should be noted that $\sqsubseteq_{must}$ is by definition a precongruence, so that $\simeq_{must}$ is a congruence relation.

Summarizing the discussion above, we claim that non determinism is in our functional setting conjunctive in nature, while parallelism is disjunctive. This is the main motivation for studying the relations just defined.

As a final remark and further motivation, we note that may-solvability leads to unbounded (but still countable) indeterminacy. In fact, consider the term $H\mathbf{0}$, where

$$H \equiv \mathbf{\Theta}(\lambda hx.x \oplus h(\mathbf{Succ}\; x)),$$

$\mathbf{\Theta} \equiv (\lambda xy.y(xxy))(\lambda xy.y(xxy))$ is a fixpoint combinator and $\mathbf{0}$, $\mathbf{Succ}$ represent zero and successor in some numerical system (see [9]), whose numerals have normal form; then $H\mathbf{0}$ reduces to an infinite countable set of head normal forms. $H\mathbf{0}$ is of course may-solvable, but clearly must-unsolvable or divergent, having the infinite head reduction:

$$H\mathbf{0} \xrightarrow{*}_h \mathbf{0} \oplus H(\mathbf{Succ}\;\mathbf{0}) \longrightarrow_h H(\mathbf{Succ}\;\mathbf{0}) \xrightarrow{*}_h \mathbf{Succ}\;\mathbf{0} \oplus H(\mathbf{Succ}(\mathbf{Succ}\;\mathbf{0})) \longrightarrow_h \cdots$$

On the other hand must-solvability implies that each convergent term has finite set of *principal* head normal forms.

**Lemma 2.5** *If $M\downarrow$, then the set $\{N \mid N \in \mathrm{HNF} \wedge M \xrightarrow{*}_h N\}$ is finite and non empty.*

*Proof.* By Def. 2.4 and König lemma, the tree of head reductions being finitary. □

This is not a limitation in the expressive power of a theory based on must-solvability, because being must solvable does not imply to have a finite set of normal forms: consider e.g. the term $H'\mathbf{0}$ where

$$H' \equiv \mathbf{\Theta}(\lambda hx.x \oplus \mathbf{Succ}(hx)),$$

which has the same set of normal forms as $H\mathbf{0}$, being must-convergent.

## 2.2 A Standardization Theorem

The reason to focus on head reductions in classical $\lambda$-calculus is that any normalizing reduction can be factored into two parts, an exhaustive head reduction followed by internal reductions:

$$M \xrightarrow{*}_h L \xrightarrow{*}_i N.$$

This remains true even if we don't reach a normal form, i.e. any finite reduction can be rearranged in this way (just the head reduction may not be exhaustive, terminating in a term which is not in HNF), and, moreover, the internal part will have on subterms recursively the same structure.

This fact, which is basic in the study of reduction in the classical calculus, is usually proved using Church-Rosser property; we show however that it remains true in our calculus, and can be consequently used to develop operational semantics.

To define the notion of standard reduction some machinery is needed, basically to keep track of the redexes and of the order in which they are contracted. Given $M \in \Lambda_\oplus$, $\Delta \in M$ means that $\Delta$ is a redex occurrence in $M$; similarly, if $\mathcal{F} = \{\Delta_1, \ldots, \Delta_n\}$, then $\mathcal{F} \subseteq M$ means that $\Delta_i \in M$, for all $1 \leq i \leq n$. Finally, suppose the (binary) syntactical tree of each term $M \in \Lambda_\oplus$ labelled with strings in $\{0,1\}$ in the usual way: then $M \,@\, u$, for $u \in \{0,1\}^*$, is the subterm of $M$ rooted at $u$. The occurrence $u$, when it does not cause confusion, will be understood without any special notation.

**Definition 2.6** *If $\Delta_1, \Delta_2 \in M$, with $\Delta_1 \equiv M \,@\, u$ and $\Delta_2 \equiv M \,@\, v$; then define*

$$\Delta_1 \leq \Delta_2 \;\Leftrightarrow\; u \leq_{lex} v,$$

*where $\leq_{lex}$ is the lexicographic ordering. Now $\Delta_1/\Delta_2$ is the set of* residuals *of $\Delta_1$ after contracting $\Delta_2$, defined as the following set of redex occurrences in $M$:*

*i)* $u = v \Rightarrow \Delta_1/\Delta_2 = \emptyset$;

*ii)* $\Delta_1 < \Delta_2$ *or* $\Delta_2 < \Delta_1$ *and* $\Delta_1 \notin \Delta_2$ *and* $\Delta_2 \notin \Delta_1 \;\Rightarrow\; \Delta_1/\Delta_2 = \{\Delta_1\}$;

*iii)* $\Delta_2 \in \Delta_1 \;\Rightarrow\; \Delta_1/\Delta_2 = \{\Delta_1'\}$, *where* $\Delta_1'$ *is obtained from* $\Delta_1$ *replacing* $\Delta_2$ *with its contractum;*

*iv)* $\Delta_1 \in \Delta_2$, *then there are three subcases:*

- *a)* $\Delta_2 \equiv (\lambda x.P)Q$, $\Delta_1 \in P \;\Rightarrow\; \Delta_1/\Delta_2 = \{\Delta_1[Q/x]\}$*;*
- *b)* $\Delta_2 \equiv (\lambda x.P)Q$, $\Delta_1 \in Q \;\Rightarrow\; \Delta_1/\Delta_2 = \{\Delta^1, \ldots \Delta^r\}$ *where each* $\Delta^i$ *is a copy of* $\Delta_1$ *and* $r$ *is the number of occurrences of* $x$ *in* $P$*;*
- *c)* $\Delta_2 \equiv P \oplus Q$, $\Delta_1 \in P \;\Rightarrow\; \Delta_1/\Delta_2 = \{\Delta_1\}$, *if* $\Delta_2$ *reduces to* $P$*;* $\Delta_1/\Delta_2 = \emptyset$ *otherwise; the case* $\Delta_1 \in Q$ *is similar.*

The concept of residuals, introduced above in the case of one step reductions, can be extended to any reduction sequence $\sigma$.

**Notation**: Assume that $\sigma : M \xrightarrow{*} N$ is a finite reduction sequence; then it has the form:

$$\sigma : M \equiv M_0 \xrightarrow{\Delta_1} M_1 \xrightarrow{\Delta_2} M_2 \cdots \xrightarrow{\Delta_n} M_n \equiv N,$$

for $n \geq 0$; write $\sigma = \Delta_1 + \cdots + \Delta_n$; furthermore $\sigma_{i,j} : M_i \xrightarrow{*} M_j$ is the subreduction of $\sigma$ from $M_i$ to $M_j$. Finally $|\sigma| = n$ is the length of $\sigma$.

**Definition 2.7** *Suppose* $\Delta \in M$ *and* $\sigma = \Delta_1 + \cdots + \Delta_n$, *then the set of* residuals of $\Delta$ modulo $\sigma$, *written* $\Delta/\sigma$, *is inductively defined:*

$$\begin{array}{rcl} \sigma = \Delta_1 & \Rightarrow & \Delta/\sigma = \Delta/\Delta_1, \\ \sigma = \Delta_1 + \sigma' & \Rightarrow & \Delta/\sigma = \bigcup\{\Delta'/\sigma' \mid \Delta' \in \Delta/\Delta_1\}. \end{array}$$

It is now possible to introduce the central notion of standard reduction.

**Definition 2.8** *Suppose* $\sigma : M \xrightarrow{*} N$ *be any reduction, then* $\sigma$ *is* standard *iff*

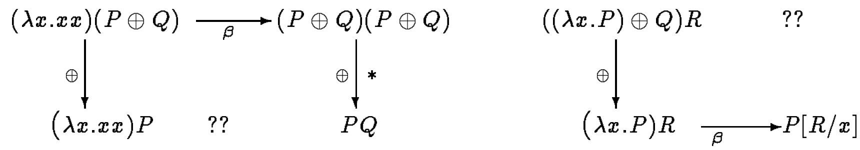
$$\forall i, j \leq |\sigma| \,.\, i < j \;\Rightarrow\; \neg \exists \Delta \in M_i.\, \Delta \leq \Delta_{i+1} \;\wedge\; \Delta_j \in \Delta/\sigma_{i,j}.$$

*We write* $\sigma : M \longrightarrow_s N$ *to mean that* $\sigma$ *is standard.*

In words, a reduction is standard iff a residual of a redex whose main operator occurs to the left of the main operator of a contracted redex is never contracted thereafter. We have:

**Theorem 2.9 (Standardization Theorem)** $\forall M, N \in \Lambda_\oplus.\, M \xrightarrow{*} N \Rightarrow M \longrightarrow_s N.$

To achieve this result one usually looks for the possibility of performing the "same" contractions, although in a different order. In the present case some difficulties arise: first the Church-Rosser theorem does not hold; second in general one cannot permute $\beta$-reductions with $\oplus$-reductions:




Indeed, the figure above shows that a $\beta$-contraction can multiply $\oplus$-redexes (left), hence the number of possible choices, and that $\oplus$-contractions may create new $\beta$-redexes (right). Nevertheless, in Section 7.1, we will give a proof of the standardization theorem following the pattern of Klop's one [30] for classical $\lambda$-calculus (see also [9], §12.3).

**Corollary 2.10** *For any reduction $\sigma : M \xrightarrow{*} N$, there exists a reduction $\sigma' : M \xrightarrow{*} N$ such that, for some $L \in \Lambda_\oplus$,*

$$\sigma' = M \xrightarrow{*}_h L \xrightarrow{*}_i N.$$

*Proof.* Immediate by the standardization theorem. □

# 3 Algebraic Semantics

In order to introduce the reader to the study of the functional behaviour of terms, let us take into account the terms

$$M \equiv \lambda x.x(P \oplus Q) \quad \text{and} \quad N \equiv (\lambda x.xP) \oplus (\lambda x.xQ).$$

If we assume the *immediate deterministic structure* of a term to be the set of terms obtained from it eliminating the choices in all possible ways, we have

$$\forall P, Q \in \Lambda. \{T \in \Lambda \mid M \xrightarrow{*}_\oplus T\} = \{T \in \Lambda \mid N \xrightarrow{*}_\oplus T\},$$

i.e. $M$ and $N$ have the same immediate deterministic structure. However, let us apply $M$ and $N$ to the term $\omega \equiv \lambda y.yy$; as shown in Fig.1, where the head reductions of $M\omega$ and $N\omega$ are illustrated, $M\omega$ can produce $PP$ or $QQ$ or $PQ$ or $QP$, while $N\omega$ can produce $PP$ or $QQ$ only. Furthermore, if we take

$$P \equiv \lambda x.x\mathbf{U}_3^3\omega \quad \text{and} \quad Q \equiv \lambda xy.yy,$$

where $\mathbf{U}_3^3 \equiv \lambda x_1x_2x_3.x_3$, a simple (head) reduction shows that

$$PP, QQ, QP \xrightarrow{*} \omega, \quad \text{while} \quad PQ \xrightarrow{*} \omega\omega \to \omega\omega \to \cdots,$$

that is $M\omega\uparrow$ while $N\omega\downarrow$; hence $N \not\sqsubseteq_{must} M$.

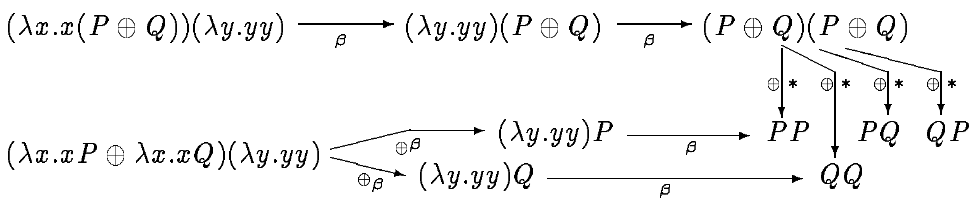



Figure 1: Different behaviours of $M\omega$ and $N\omega$

It comes out that the immediate deterministic structure of terms is not sufficient to characterize their applicative behaviour. Hence, we are led to consider not only the possibility of eliminating choices from terms, but also *how* this can be performed, thus arguing that $M$ and $N$ must have a different "choice structure". Informally speaking we refer to the choice structure as to what causes two (or more) terms to behave differently even if they have the same immediate deterministic structure.

Indeed, the distinction in behaviours of $M$ and $N$ is due to the fact that the choice $P \oplus Q$ in $M$ is duplicated by $\omega$ in $M\omega$, thus giving the opportunity of obtaining every possible combination of $P$ and $Q$. This is not possible in $N$, where duplication comes only after the choice. It seems then that some difference can be recognized in $M$ and $N$ in the relative position of abstracted variables with respect to occurrences of the choice operator.

Aiming to a formalization of the notion of choice structure of a term, it has to be noted that it cannot be immediately deduced from the syntactical tree of the term itself; consider, as a counterexample, the syntactically different terms

$$F \equiv \lambda x.(T \oplus U) \text{ and } G \equiv (\lambda x.T) \oplus (\lambda x.U);$$

to give an intuition that $F$ and $G$ have indeed the same functional behaviour, we observe that, for any $H$, $FH$ and $GH$ can produce by head reduction exactly the same objects, namely those obtenable from $T[H/x]$ and $U[H/x]$.

We will give a formalization of the notion of choice structure of a term by means of *non deterministic Böhm trees*, which extend Böhm trees of classical $\lambda$-calculus to the non deterministic case; in such a perspective, turning back to the previous examples, we will obtain different trees for the terms $M$ and $N$ and the same tree for $F$ and $G$, as shown in Fig.2.

### 3.1 Non deterministic Böhm trees

We will now introduce a kind of unfolding trees, generalizing the notion of Böhm trees of the classical $\lambda$-calculus (for this notion see [9]), and a suitable notion of approximation,

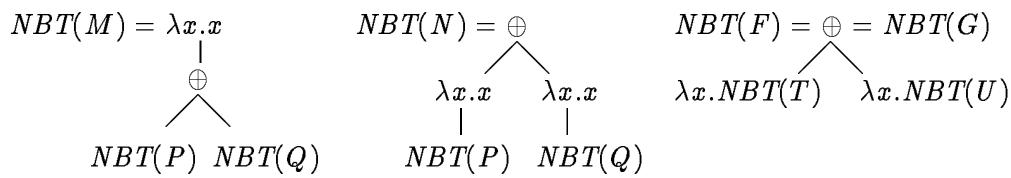



Figure 2: Examples of *non deterministic Böhm trees*

to be considered as a cut of the tree. This will give us what is called the algebraic semantics of the calculus, after e.g. [3] and [20].

The difficulty here is that, since the Church-Rosser property doesn't hold, we cannot consider our trees as a representation of the directed set of "approximated" reducts of a term: instead, we have to take into account all possible reductions, without making choices, but simply representing them in the tree.

**Definition 3.1** *We define, by mutual induction, two sets $\mathcal{S}_0$ and $\mathcal{S}_1$ of* approximants*:*

*i)* $\Omega \in \mathcal{S}_0$, *where $\Omega$ is a new constant representing divergency;*

*ii)* $M \in \mathcal{S}_0 - \{\Omega\} \Rightarrow \lambda x.M \in \mathcal{S}_0$*;*

*iii)* $\{\Omega\} \in \mathcal{S}_1$*;*

*iv)* $M_1, ..., M_n \in \mathcal{S}_0 - \{\Omega\} \Rightarrow \{M_1, ..., M_n\} \in \mathcal{S}_1 \quad$ *for* $n > 0$*;*

*v)* $\mathcal{M}_1, ..., \mathcal{M}_m \in \mathcal{S}_1, x \in Var \Rightarrow x\mathcal{M}_1 ... \mathcal{M}_m \in \mathcal{S}_0 \quad$ *for* $m \geq 0$*.*

The idea behind this notation is that an approximation is a finite set in $\mathcal{S}_1$ of elements in $\mathcal{S}_0$; this comes out from the fact that, if $M\!\downarrow$, then there is a finite set of head normal forms we derive from it: this settles the first level, and we go on recursively with the bodies of these terms.

We consider in clause (v) the application of a variable rather than a set since we have in mind head normal forms: a set is a sum, and any term with a sum in head position is not a head normal form. Similar remarks apply to clause (ii).

For every approximant, there is a term in $\Lambda_\oplus\Omega$ (i.e., $\Lambda_\oplus$ extended with the constant $\Omega$) which corresponds to it in a natural way; we define, by mutual induction, $\vartheta_0 : \mathcal{S}_0 \rightarrow \Lambda_\oplus\Omega$ and $\vartheta_1 : \mathcal{S}_1 \rightarrow \Lambda_\oplus\Omega$:

$$\vartheta_0(\Omega) = \vartheta_1(\{\Omega\}) = \Omega, \qquad \vartheta_0(\lambda x.M) = \lambda x.\vartheta_0(M),$$
$$\vartheta_0(x\mathcal{M}_1 \ldots \mathcal{M}_m) = x\vartheta_1(\mathcal{M}_1) \ldots \vartheta_1(\mathcal{M}_m);$$
$$\vartheta_1(\{M_1, \ldots, M_n\}) = \vartheta_0(M_1) \oplus \ldots \oplus \vartheta_0(M_n).$$

To simplify the notation, we will identify $\mathcal{M} \in \mathcal{S}_1$ with $\vartheta_1(\mathcal{M}) \in \Lambda_\oplus\Omega$.

**Definition 3.2** *Let $\mathcal{M} \in \mathcal{S}_1$; we define*

$$NBT(\mathcal{M}) = NBT_1(\mathcal{M}),$$

*where* $\quad NBT_0(\Omega) = \Omega$

*and* $\quad NBT_0(\lambda\vec{x}.\xi\mathcal{M}_1...\mathcal{M}_m) = \lambda\vec{x}.\xi \qquad NBT_1(\{M_1, ..., M_n\}) = \oplus$

$$\begin{array}{c} \lambda\vec{x}.\xi \\ \diagup \quad \diagdown \\ NBT_1(\mathcal{M}_1) \; \cdots \; NBT_1(\mathcal{M}_m) \end{array} \qquad \begin{array}{c} \oplus \\ \diagup \quad \diagdown \\ NBT_0(M_1) \; \cdots \; NBT_0(M_n) \end{array}$$

**Remark 3.3** Note that the symbol $\oplus$ is here just an object to be used as a label in some of the nodes of the tree, and it must not be confused with the same symbol as it has been introduced into the syntax of $\Lambda_\oplus$. Note also that the order of the subnodes of a node labelled by $\lambda\vec{x}.\xi$ is relevant, as well as multiple occurrences of the same subtree; this is not the case for sons of nodes labelled by $\oplus$.

*NBT*s may be seen as infinite $\mathcal{S}_1$-terms, that is as elements of the completion of $\mathcal{S}_1$ under the order induced on $\mathcal{S}_1$ by the relation freely generated by $\Omega \preceq M$, for all $M$, on $\mathcal{S}_0$ (see for similar constructions involving the extension of algebraic semantics to powerdomains [1, 24]). More precisely, they are the limits of those directed subsets of $\mathcal{S}_1$ generated by the following family of maps:

**Definition 3.4** *For each natural number $k$, we define a map $\omega^k$: $\Lambda_\oplus \to \mathcal{S}_1$ by:*

*i)* $\omega^0(M) = \{\Omega\}$,

*ii)* $$\omega^{k+1}(M) = \begin{cases} \{\Omega\} & \text{if } M\uparrow; \\ \{\lambda\vec{x}.\xi\,\omega^k(M_1)...\omega^k(M_m) \mid \lambda\vec{x}.\xi M_1...M_m \\ \qquad\qquad \text{is a principal hnf of } M\} & \text{otherwise.} \end{cases}$$

*Furthermore, we denote*

$$M^{[k]} = \vartheta_1 \circ \omega^k(M).$$

**Remark 3.5** We note that $M^{[k]}$ is always a $\beta$-$\Omega$-normal form in the sense of [9], Ch.14; indeed in any term of this shape no $\oplus$-redex can create new $\beta$-$\Omega$-redexes. We denote by $\mathrm{N}_\oplus^\Omega$ the set of such terms.

**Example 3.6** Let $M \equiv \lambda x.x(yx) \oplus \lambda x.xz \in \Lambda_\oplus$. Then we have:

$$\begin{array}{rcl} \omega^1(M) & = & \{\lambda x.x\,\omega^0(yx)\ ,\ \lambda x.x\,\omega^0(z)\} = \{\lambda x.x\{\Omega\}\}, \\ \omega^2(M) & = & \{\lambda x.x\,\omega^1(yx)\ ,\ \lambda x.x\,\omega^1(z)\} = \{\lambda x.x\{y\,\omega^0(x)\}\ ,\ \lambda x.x\{z\}\}, \\ & = & \{\lambda x.x\{y\{\Omega\}\}\ ,\ \lambda x.x\{z\}\}, \\ M^{[1]} & = & \lambda x.x\Omega, \\ M^{[2]} & = & \lambda x.x(y\Omega) \oplus \lambda x.xz. \end{array}$$

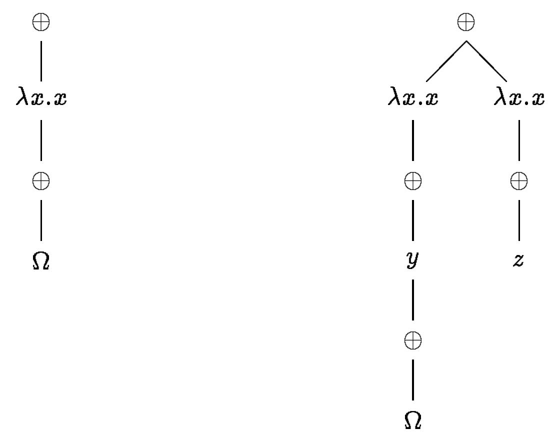



Figure 3: Non deterministic Böhm trees

The trees $NBT(\omega^1(M))$ and $NBT(\omega^2(M))$ are shown in Figure 3.

To compare two terms, that is their trees, simple inclusion doesn't suffice even in the classical $\lambda$-calculus: what we need is a generalization of the relation ${}^{\eta}\sqsubseteq^{\eta}$ in [9], or, equivalently, of $<_k^s$ in [28]: this will be achieved in several steps.

We first recall the notion of *equivalence* ($\sim$) for head normal forms [13, 9] (called *similarity* in [28]) adapting it to the set $\mathcal{S}_0$.

**Definition 3.7**

*i) let* $M \equiv \lambda x_1 \ldots x_n.\xi \mathcal{M}_1 \ldots \mathcal{M}_m \in \mathcal{S}_0$, *then* $head(M) = \xi$, $ord(M) = n$, $deg(M) = m$;

*ii) for* $M, N \in \mathcal{S}_0 - \{\Omega\}$,

$$M \sim N \;\Leftrightarrow\; deg(M) - ord(M) = deg(N) - ord(N) \text{ and } head(M) \equiv head(N).$$

The role of this relation in discriminating head normal forms is illustrated in [13, 17] for the classical $\lambda$-calculus: in our setting, sets of head normal forms must be compared, and we will consider equivalent two sets $\mathcal{M}$ and $\mathcal{N}$ iff

$$\forall X \in \mathcal{M}\; \exists Y \in \mathcal{N}.\; X \sim Y \quad \text{and} \quad \forall Y \in \mathcal{N}\; \exists X \in \mathcal{M}.\; X \sim Y$$

**Example 3.8** Let $M \equiv \lambda x_1x_2x_3.x_1x_3(x_2x_3) \oplus \lambda x_1x_2.x_1x_2x_2 \oplus \lambda x_1x_2x_3.x_1x_3x_2$ and $N \equiv \lambda x_1x_2.x_1x_1 \oplus \lambda x_1x_2x_3.x_1x_2$.

We have:

$$\begin{aligned}
\omega^2(M) &= \{\lambda x_1x_2x_3.x_1\{x_3\}\{x_2\{\Omega\}\} \ , \ \lambda x_1x_2.x_1\{x_2\}\{x_2\} \ , \\
&\quad \lambda x_1x_2x_3.x_1\{x_3\}\{x_2\}\}; \\
\omega^2(N) &= \{\lambda x_1x_2.x_1\{x_1\} \ , \ \lambda x_1x_2x_3.x_1\{x_2\}\}; \\
M^{[2]} &= \lambda x_1x_2x_3.x_1x_3(x_2\Omega) \oplus \lambda x_1x_2.x_1x_2x_2 \oplus \lambda x_1x_2x_3.x_1x_3x_2; \\
N^{[2]} &= N.
\end{aligned}$$

It comes out that there exists a $\sim$-equivalence class of $\omega^2(M)\cup\omega^2(N)$ which does not contain any element of $\omega^2(M)$. In this case, we can immediately find a context such that $C[M]$ converges while $C[N]$ does not. Indeed, take $C[\,] \equiv [\,](\lambda a_1a_2a_3.a_1)x_2x_3x_4(\omega\omega)$, then

$$\begin{aligned}
\omega^2(C[M^{[2]}]) &= \{x_4\{x_5\}, x_3\{x_4\}\{x_5\}\}, \\
\omega^2(C[N^{[2]}]) &= \{\Omega\}.
\end{aligned}$$

The previous example shows that non equivalent sets of head normal forms can be easily separated looking at the first level of their trees only. In the general case, however, while comparing sets of head normal forms, it is necessary to analyze the internal structure of their elements, which in turn encapsulate other sets of terms.

**Lemma 3.9** *Define the binary relation* $\sim_\eta \subseteq \mathcal{S}_0 \times \mathcal{S}_0$ *by*

$$M \sim_\eta \lambda x.M\{x\} \quad \textit{if } x \notin FV(M) \textit{ and } M \not\equiv \Omega.$$

*Let* $\mathcal{M} \in \mathcal{S}_1$ *and*

$$\eta(\mathcal{M}) = \{\bar{M} \mid \exists M \in \mathcal{M}.\ \bar{M} \sim_\eta M \ \wedge \ ord(\bar{M}) = \max\{ord(M') \mid M' \in \mathcal{M}\}\}.$$

*Then* $\sim_\eta \subseteq \sim$ *and* $\eta(\mathcal{M}) \in \mathcal{S}_1$ *is a finite set of* $\sim$*-equivalent objects of* $\mathcal{S}_0$*.*

*Proof.* Immediate by Def. 3.7 and from the fact that for each $M$ and $n \geq ord(M)$ there exsists exactly one $\bar{M}$ s.t. $\bar{M} \sim_\eta M$ and $ord(\bar{M}) = n$. □

**Remark 3.10** The relation $\sim_\eta$ is clearly reminiscent of $\eta$ conversion even if here we don't define it as a congruence. It turns out that, using this relation we can obtain from $\sim$-equivalent objects terms having the same order and degree. This is useful as a technical tool to simplify the construction we are carrying out.

The following definitions enable us to select those parts of $\mathcal{M}$ and $\mathcal{N}$ which must be collated to establish the relative functional behaviour of $\mathcal{M}$ and $\mathcal{N}$.

**Definition 3.11** *Let $\mathcal{M}, \mathcal{N} \in \mathcal{S}_1$, then the set* $\mathsf{Pair}(\mathcal{M}, \mathcal{N})$ *is defined as follows. Suppose that*

$$\mathcal{M} = \mathcal{M}_1 \cup \cdots \cup \mathcal{M}_k \quad and \quad \mathcal{N} = \mathcal{N}_1 \cup \cdots \cup \mathcal{N}_h$$

*where the $\mathcal{M}_i$ and $\mathcal{N}_j$ are the $\sim$-equivalence classes of the elements of $\mathcal{M}$ and $\mathcal{N}$ respectively. Furthermore, suppose that for some $i$ and $j$ $\mathcal{M}_i \cup \mathcal{N}_j$ is formed of $\sim$-equivalent objects. Take*

$$\eta(\mathcal{M}_i \cup \mathcal{N}_j) = \{\bar{M}^1, \ldots, \bar{M}^m, \bar{N}^1, \ldots, \bar{N}^n\},$$

*where the $\bar{M}^u, \bar{N}^v$ come from $\mathcal{M}_i$ and $\mathcal{N}_j$ respectively; these objects will have the shapes*

$$\bar{M}^u \equiv \lambda\vec{y}.x\mathcal{M}_1^u \ldots \mathcal{M}_l^u, \text{ for } 1 \le u \le m \quad and \quad \bar{N}^v \equiv \lambda\vec{y}.x\mathcal{N}_1^v \ldots \mathcal{N}_l^v, \text{ for } 1 \le v \le n.$$

*Then for each $1 \le p \le l$,*

$$\langle\{\mathcal{M}_p^1, \ldots, \mathcal{M}_p^m\}, \{\mathcal{N}_p^1, \ldots, \mathcal{N}_p^n\}\rangle \in \mathsf{Pair}(\mathcal{M}, \mathcal{N}).$$

Note that in the previous definition we indeed compare the following "matrices" having the same number of columns and possibly different number of rows:

$$\begin{pmatrix} \mathcal{M}_1^1 & \cdots & \mathcal{M}_l^1 \\ \\ \mathcal{M}_1^m & \cdots & \mathcal{M}_l^m \end{pmatrix} \qquad \begin{pmatrix} \mathcal{N}_1^1 & \cdots & \mathcal{N}_l^1 \\ \\ \mathcal{N}_1^n & \cdots & \mathcal{N}_l^n \end{pmatrix}$$

We take as the elements of $\mathsf{Pair}(\mathcal{M}, \mathcal{N})$ the pairs of corresponding columns. Henceforth we shall use $\mathsf{U}, \mathsf{V}$, possibly decorated by indices and primes, to denote these columns.

Given $\mathcal{M}, \mathcal{N} \in \mathcal{S}_1$, $\mathsf{Pair}(\mathcal{M}, \mathcal{N})$ selects the subterms to be compared during the first step of the analysis of the internal structure of $\mathcal{M}$ and $\mathcal{N}$. As in [28], this notion has to be extended to each level of the tree.

**Definition 3.12** *Given $\mathcal{M}$, $\mathcal{N} \in \mathcal{S}_1$, define* $\mathsf{Pair}_k(\mathcal{M}, \mathcal{N})$, *for each natural number $k$, as follows:*

*i)* $\mathsf{Pair}_1(\mathcal{M}, \mathcal{N}) = \mathsf{Pair}(\mathcal{M}, \mathcal{N})$.

*ii) Let* $\mathsf{Pair}(\mathcal{M}, \mathcal{N}) = \{\langle \mathsf{U}_1, \mathsf{V}_1\rangle, \ldots, \langle \mathsf{U}_l, \mathsf{V}_l\rangle\}$ *and*

$$\mathcal{M}'_i = \bigcup\{\mathcal{M}' \mid \mathcal{M}' \in \mathsf{U}_i\}, \qquad \mathcal{N}'_i = \bigcup\{\mathcal{N}' \mid \mathcal{N}' \in \mathsf{V}_i\},$$

*where $1 \le i \le l$: then*

$$\mathsf{Pair}_{k+1}(\mathcal{M}, \mathcal{N}) = \{\langle \mathsf{A}, \mathsf{B}\rangle \mid \exists i \le l.\ \langle \mathsf{A}, \mathsf{B}\rangle \in \mathsf{Pair}_k(\mathcal{M}'_i, \mathcal{N}'_i)\}.$$

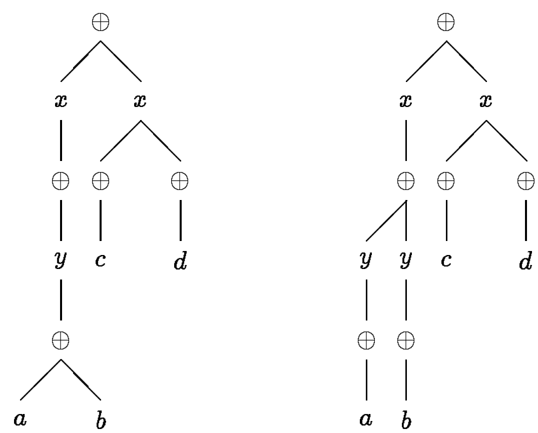


Figure 4: Respectively the trees of $\omega^3(M)$ and $\omega^3(N)$

**Remark 3.13** Note that in (ii), for $1 \le i \le l$, $\mathsf{U}_i$ and $\mathsf{V}_i$ are finite non empty sets of objects in $\mathcal{S}_1$, hence families of finite non empty sets of objects in $\mathcal{S}_0$; it follows that their unions $\mathcal{M}'_i$ and $\mathcal{N}'_i$ are again elements of $\mathcal{S}_1$.

**Example 3.14** Let $M \equiv x(y(a \oplus b)) \oplus xcd$ and $N \equiv x(ya \oplus yb) \oplus xcd$ (see Figure 4); we have

$$\begin{array}{rcl} \omega^3(M) & = & \{x\{y\{a,b\}\}\ ,\ x\{c\}\{d\}\} \\ \omega^3(N) & = & \{x\{y\{a\},y\{b\}\}\ ,\ x\{c\}\{d\}\}. \end{array}$$

From this we compute:

$$\begin{array}{rcl} \mathsf{Pair}_1(\omega^3(M),\omega^3(N)) & = \{ & \langle\{\{y\{a,b\}\}\},\{\{y\{a\},y\{b\}\}\}\rangle, \\ & & \langle\{\{c\}\},\{\{c\}\}\rangle\ ,\ \langle\{\{d\}\},\{\{d\}\}\rangle\quad \}; \\ \mathsf{Pair}_2(\omega^3(M),\omega^3(N)) & = & \mathsf{Pair}_1(\{y\{a,b\}\},\{y\{a\},y\{b\}\})\cup \\ & & \mathsf{Pair}_1(\{c\},\{c\})\cup\mathsf{Pair}_1(\{d\},\{d\}) \\ & = \{ & \langle\{\{a,b\}\},\{\{a\},\{b\}\}\rangle\quad \}. \end{array}$$

We are now ready to introduce the ordering relation $\le$ over trees.

**Definition 3.15** *Given $\mathcal{M},\mathcal{N} \in \mathcal{S}_1$ we define, for each $k$, a relation $\le_k$ by:*

$$\begin{array}{rcl} \mathcal{M} \le_1 \mathcal{N} & \Leftrightarrow & \mathcal{M} = \{\Omega\} \ \vee\ \forall N \in \mathcal{N}\ \exists M \in \mathcal{M}.\ M \sim N, \\ \mathcal{M} \le_{k+1} \mathcal{N} & \Leftrightarrow & \mathcal{M} \le_k \mathcal{N} \wedge \\ & & \forall\langle\mathsf{U},\mathsf{V}\rangle \in \mathsf{Pair}_k(\mathcal{M},\mathcal{N}).\ \mathsf{U} \sqsubseteq^\sharp \mathsf{V}, \end{array}$$

*where*

$$\mathsf{U} \sqsubseteq^\sharp \mathsf{V} \Leftrightarrow \forall\mathcal{N}_j \in \mathsf{V}\ \exists\mathcal{M}_i \in \mathsf{U}.\ \mathcal{M}_i \le_1 \mathcal{N}_j.$$

*From this we can define*

$$\mathcal{M} \leq \mathcal{N} \Leftrightarrow \forall k.\ \ \mathcal{M} \leq_k \mathcal{N}.$$

*Finally, for any* $M$, $N \in \Lambda_{\oplus}$,

$$\begin{aligned} M \leq_k N &\Leftrightarrow \omega^k(M) \leq_k \omega^k(N), \\ M \leq N &\Leftrightarrow \forall k.\ \ M \leq_k N. \end{aligned}$$

**Example 3.16** Consider the $M$ and $N$ of example 3.14. Now the following relations hold:

$$M \leq_i N \leq_i M \quad \text{for } i = 0, 1, 2 \quad .$$

Indeed

$$\omega^1(M) = \{x\{\Omega\}, x\{\Omega\}\{\Omega\}\} = \omega^1(N);$$

and

$$\omega^2(M) = \{x\{y\{\Omega\}\}, x\{c\}\{d\}\} = \omega^2(N).$$

But, looking at example 3.14, we see that in $\mathsf{Pair}_2(\omega^3(M), \omega^3(N))$ we have a (unique) pair formed by an $\mathsf{U} = \{\{a, b\}\}$ and by a $\mathsf{V} = \{\{a\}, \{b\}\}$, such that

$$\mathsf{U} \sqsubseteq^\sharp \mathsf{V} \quad \text{and} \quad \mathsf{V} \not\sqsubseteq^\sharp \mathsf{U},$$

hence $M \leq_3 N \not\leq_3 M$.

## 3.2 Discriminability and Tree Inclusion

Let $M, N \in \Lambda_{\oplus}$. In order to give a sufficient condition for the existence of a discriminating context $C[\,]$ such that $C[M]{\downarrow} \wedge C[N]{\uparrow}$, we will distinguish three cases:

$M \not\leq_1 N$. In such case we are considering terms encoding non equivalent sets of head normal forms: the existence of a discriminating context will be guaranteed by the application of the separability technique of classical $\lambda$-calculus [17], as exemplified in 3.8.

$M \not\leq_2 N$ and the previous case does not hold. In such case we are considering terms which encode equivalent sets of head normal forms, but the existence of $\langle \mathsf{U}, \mathsf{V} \rangle \in \mathsf{Pair}_2(\omega^h(M), \omega^h(N))$ (for some $h \geq 2$) such that $\mathsf{U} \not\sqsubseteq^\sharp \mathsf{V}$ implies that different sets (i.e. choices) appearing at the second level in $NBT(M)$ and $NBT(N)$ are indeed "arguments" of the same variables, namely those labelling the first level of the trees: substituting for these variables suitable combinators, we will succeed in discriminating among sets. This is the case where we can say that $M$ and $N$ exhibit a different *choice structure* at the second level of their $NBT$s.

$M \not\lesssim_k N$ where $k > 2$ and the previous cases do not hold. In such case $M$ and $N$ have an uniform structure "up to" the $k$-th level of their *NBT*s, where they eventually encode different information. Using such uniformity we shall prove that this information can be lifted up until one of the previous cases hold.

**Definition 3.17** *A context is a* head context *iff it is of the form*

$$C[\,] \equiv (\lambda x_1 \dots x_n.[\,])X_1 \dots X_n U_1 \dots U_m.$$

**Lemma 3.18** *Let $\mathcal{M} = \omega^k(M)$ for $k \geq 1$ and $M\!\downarrow$; assume $\mathcal{M}_{/\sim} = \{[M_1], \dots, [M_h]\}$; then*

*i) for each $i \in \{1, \dots, h\}$ there exists a head context $C_i[\,]$ and an integer $r_i$ s.t. for each $L \in \mathcal{M}$*

$$\omega^k(C_i[L]) = \begin{cases} \{x_i \mathcal{L}_1 \dots \mathcal{L}_{r_i}\} & \text{if } L \in [M_i] \\ \{y\} & \text{otherwise} \end{cases}$$

*where $y$ is any fixed variable;*

*ii) there exists a head context $C[\,]$ s.t.*

$$\omega^k(C[M]) = \{z_1, \dots, z_h\},$$

*where $\{z_1, \dots, z_h\}$ are new, pairwise distinct variables, and for each $i \leq h$*

$$\omega^k(C[L]) = \{z_i\} \;\Leftrightarrow\; L \in [M_i].$$

*Proof.* By standard separability techniques (see [17]). $\square$

$\lambda$-calculus encodes all recursive functions; this can be done in many different ways, choosing a suitable *numeral system.* For technical reasons, we will make use of Church's numeral system:

$$\mathbf{c}_0, \mathbf{c}_1, \dots \quad \text{where} \quad \mathbf{c}_n \equiv \lambda f x.\, \underbrace{f(\dots f}_{n}(x)\dots);$$

defining $\mathbf{Succ} \equiv \lambda xyz.y(xyz)$ we have $\mathbf{Succ}\,\mathbf{c}_n =_\beta \mathbf{c}_{n+1}$, for any $n$; such system is *adequate* [9] in that all recursive functions are representable using it. In the sequel we shall write $\mathbf{n}$ for $\mathbf{c}_n$.

It is well known that the test for equality for Church numerals is $\lambda$-definable; however we need to represent such a test with a combinator of a special shape. The proofs of the next lemmas and corollary are given in Section 7.

**Lemma 3.19** *There exists a combinator* $\mathbf{H} \in \Lambda$ *of the shape*

$$\mathbf{H} \equiv \lambda xy.xH_1 \ldots H_l,$$

*with* $x \notin FV(H_1) \cup \ldots \cup FV(H_l)$*, such that, for all non-negative integers* $n, m$*:*

$$\mathbf{Hnm} =_{\beta\eta} \begin{cases} \mathbf{1} & \text{if } n = m \\ \mathbf{0} & \text{otherwise} \end{cases}$$

**Corollary 3.20** *If* $N \equiv \mathbf{n}_1 \oplus \ldots \oplus \mathbf{n}_r$*, with* $r \geq 1$*, then, for all* $m$*,*

$$\omega^1(\mathbf{H}N\mathbf{m}) = \{\mathbf{0} \mid \exists i \leq r.\ n_i = m\} \cup \{\mathbf{1} \mid \exists j \leq r.\ n_j \neq m\}.$$

**Lemma 3.21** *For* $M, N \in \Lambda_\oplus$*,*

$$M \not\leq_2 N \ \Rightarrow\ \exists C[\,].\, C[M]\!\downarrow \ \wedge\ C[N]\!\uparrow.$$

**Example 3.22** We exhibit an example of the case where $M \leq_1 N$ but $M \not\leq_2 N$. Take $M \equiv xy \oplus xz$ and $N \equiv x(y \oplus z)$; now $\omega^2(M) = \{x\{y\}, x\{z\}\}$, while $\omega^2(N) = \{x\{y, z\}\}$; computing $\mathsf{Pair}_1(\omega^2(M), \omega^2(N))$, we get

$$\{\langle\{\{y\},\{z\}\},\{\{y,z\}\}\rangle\}$$

and it can be verified that $\{\{y\},\{z\}\} \not\sqsubseteq^\sharp \{\{y,z\}\}$. We take

$$C_0[\,] \equiv (\lambda xyz.[\,])(\lambda w.aww)\mathbf{1}\,\mathbf{2}.$$

Simple calculations give us

$$\omega^2(C_0[M]) = \{a\{\mathbf{1}\}\{\mathbf{1}\}, a\{\mathbf{2}\}\{\mathbf{2}\}\} \text{ and } \omega^2(C_0[N]) = \{a\{\mathbf{1},\,\mathbf{2}\}\{\mathbf{1},\,\mathbf{2}\}\}.$$

Taking

$$C_1[\,] \equiv (\lambda a.[\,])(\lambda uv.\mathbf{P}(\mathbf{H}u\mathbf{1})(\mathbf{H}v\mathbf{2})),$$

where $\mathbf{P}$ $\lambda$-defines multiplication, we have

$$\omega^2(C_1[C_0[M]]) = \{\mathbf{0}\} \text{ and } \omega^2(C_1[C_0[N]]) = \{\mathbf{0},\,\mathbf{1}\}.$$

Now, taking $C_2[\,] \equiv [\,](\mathbf{K}(\omega\omega))\mathbf{I}$, we have

$$\omega^2(C_2[C_1[C_0[M]]]) = \{\mathbf{I}\} \text{ while } \omega^2(C_2[C_1[C_0[N]]]) = \{\Omega\}.$$

The following lemma, whose detailed proof is deferred to Section 7 extends to the present calculus the Böhm out lemma of the classical $\lambda$-calculus (see [9]).

**Lemma 3.23** *For $M, N \in \Lambda_\oplus$ and $k \geq 2$,*

$$M \not\lesssim_k N \;\Rightarrow\; \exists C[\,].\, C[M] \not\lesssim_2 C[N].$$

We are finally in place to prove the main theorem of the present section.

**Theorem 3.24 (Semiseparability)** *For any $M$, $N \in \Lambda_\oplus$,*

$$M \sqsubseteq_{must} N \;\Rightarrow\; M \leq N.$$

*Proof.* By contraposition, we prove (see Def. 2.4)

$$\exists k.\, M \not\lesssim_k N \;\Rightarrow\; \exists C[\,].\, C[M]{\downarrow} \wedge C[N]{\uparrow}\,.$$

Indeed,

$$M \not\lesssim N \Rightarrow \exists k.\, M \not\lesssim_k N \Rightarrow \begin{cases} (k=1) & \exists C[\,].\, C[M]{\downarrow} \wedge C[N]{\uparrow} & \text{substitute } \omega\omega \text{ for } y \\ & & \text{in lemma 3.18.} \\ (k>1) & \exists C[\,].\, C[M] \not\lesssim_2 C[N] & \text{by lemma 3.23} \\ \Rightarrow & \exists C[\,], C'[\,].\, C'[C[M]]{\downarrow} & \\ & \wedge\; C'[C[N]]{\uparrow} & \text{by lemma 3.21.} \end{cases}$$

□

# 4 Models of Non Deterministic λ-calculi

λ-calculus studies functions under their applicative behaviour; consequently models of this calculus and of its variants are applicative structures, that is sets equipped with a binary operation whose intended meaning is functional application. In the present case the structure we are looking for is an applicative structure with an extra operator modeling $\oplus$. To achieve such a structure, we shall work in the framework of *non deterministic algebras.*

## 4.1 Non Deterministic Algebras

To fix notation and to keep the treatment of denotational semantics selfcontained, we briefly summarize the relevant definitions and facts about non deterministic algebras and powerdomain functors (see [21]).

**Definition 4.1** *A* non deterministic algebra *is a structure $\langle E, +, \sqsubseteq\rangle$ where $\langle E, \sqsubseteq\rangle$ is a* **CPO** *and $+$ is a binary continuous function on $E$ satisfying:*

*i)* $x + x = x$;

*ii)* $x + y = y + x$*;*

*iii)* $(x + y) + z = x + (y + z)$*;*

*A non deterministic algebra is a* Smyth algebra *iff it satisfies (i)-(iii) and* $x + y \sqsubseteq x$*; it is a* Hoare algebra *iff it satisfies (i)-(iii) and* $x \sqsubseteq x + y$*.*

Non deterministic algebras, henceforth **NDA**, do form a category, whose morphisms are Scott continuous functions $f : D \to E$ such that

$$(Lin) \qquad\qquad f(x + y) = f(x) + f(y)$$

for all $x, y \in D$. We shall write $D \to_{lin} E$ as a shorthand for $Hom_{\mathbf{NDA}}[D, E]$. This category has the trivial algebra as its terminal object, and all products $\langle D, +_D\rangle \times \langle E, +_E\rangle = \langle D \times E, +\rangle$ where $D \times E$ is the product of $D$ and $E$ as **CPO**, and

$$\langle x, y\rangle + \langle x', y'\rangle = \langle x +_D x', y +_E y'\rangle;$$

hence it is cartesian. Moreover it has exponents $\langle D \to_{lin} E, +\rangle$ where $D \to_{lin} E$ is pointwise ordered, and

$$(f + g)(x) = f(x) +_E g(x).$$

Let $\Lambda : [D_1 \times D_2 \to_{lin} D_3] \simeq [D_1 \to_{lin} [D_2 \to_{lin} D_3]]$ be defined $\Lambda(f) = \lambda y.f(x, y)$, and $eval_{D_2,D_3}$ by $eval(f, x) = f(x)$ as usual; then it is routine to verify that these are well defined (hence are linear), that $\Lambda$ is a natural isomorphism, and that the equations of cartesian closed categories are satisfied.

It is easily seen that **NDA** embeds in the category **CPO**, so that free algebras are built from cpos by the right adjoint of the forgetful functor. We shortly recall a more explicit construction of such a functor due to Plotkin and Smyth [43, 49], being these details useful in the sequel.

**Definition 4.2** *Let* $(D, \sqsubseteq)$ *be any cpo and* $M(D) = \{u \subseteq \mathcal{K}(D) \mid u \text{ is finite}, \neq \emptyset\}$*, where* $\mathcal{K}(D)$ *is the subset of Scott compact elements of* $D$*. Then define the following preorders on* $M(D)$*:*

*i)* $u \sqsubseteq^\flat v \;\Leftrightarrow\; \forall x \in u\, \exists y \in v.\, x \sqsubseteq y$*;*

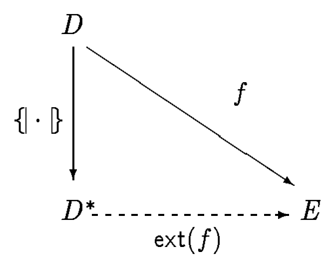
*ii)* $u \sqsubseteq^\sharp v \;\Leftrightarrow\; \forall y \in v\, \exists x \in u.\, x \sqsubseteq y$*;*

*iii)* $u \sqsubseteq^\natural v \;\Leftrightarrow\; u \sqsubseteq^\flat v \wedge u \sqsubseteq^\sharp v$*.*

For $* \in \{\flat, \sharp, \natural\}$ one has a functor $(\cdot)^* : \mathbf{CPO} \to \mathbf{NDA}$, called a *powerdomain functor*, which is the adjoint mentioned above when $* = \natural$; the functors $(\cdot)^\sharp$ and $(\cdot)^\flat$ are respectively the right adjoints of the forgetful functors from the categories **SNDA** and **HNDA**, i.e. of Smyth and Hoare non deterministic algebras, full subcategories of the category **NDA**.

The object part of each of these functors is defined by $D^* = \mathrm{Idl}(M(D), \sqsubseteq^*)$ (namely the ideal completion of $(M(D), \sqsubseteq^*)$, ordered by inclusion), which is an **NDA** object with respect to the continuous function $\uplus : D^* \times D^* \to D^*$ defined by $I \uplus J = \{u \cup v \mid u \in I \wedge v \in J\}$.

Concerning the adjointness, consider the continuous function $\{\!|\cdot|\!\} : D \to D^*$, defined by $\{\!|x|\!\} = \{u \in M(D) \mid u \sqsubseteq^* \{x\}\}$; then, given any cpos $D$ and $E$ and $f \in Hom_{\mathbf{CPO}}[D, E]$, there exists exactly one morphism $\mathsf{ext}(f) : D^* \to_{lin} E$ such that $f = \mathsf{ext}(f) \circ \{\!|\cdot|\!\}$, i.e.

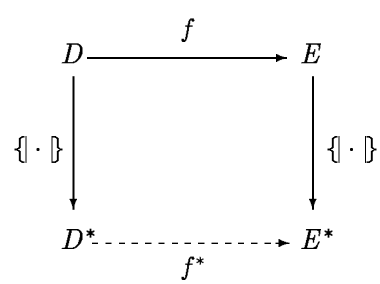



where $\mathsf{ext}(f)$ is defined

$$\mathsf{ext}(f)(I) = \bigsqcup\{\bar{f}(u) \mid u \in I\}$$

being the unique continuous extension of the function $\bar{f}(u) = f(x_1) + \cdots + f(x_n)$, for $u = \{x_1, \ldots, x_n\}$.

From this the morphism part of $(\cdot)^*$ can be defined by $f^* = \mathsf{ext}(\{\!|\cdot|\!\} \circ f)$, which implies the commutativity of the diagram




The following proposition says that $(\cdot)^*$ is actually a functor, and moreover an $\mathcal{O}$-functor as shown in [25].

**Proposition 4.3** *Let $(\cdot)^*$ be an operator among $(\cdot)^\flat$, $(\cdot)^\sharp$ and $(\cdot)^\natural$; then*

*i)* $(Id_D)^* = Id_{D^*}$*;*

*ii)* $f \circ g : D \to_{cont} E \Rightarrow (f \circ g)^* = f^* \circ g^*$;

*iii)* $f \sqsubseteq g \Rightarrow f^* \sqsubseteq g^*$ *($\mathcal{O}$-functoriality property),*

*where* $\sqsubseteq$ *is the pointwise ordering.*

It is worth to stress two relevant facts: the first one is that this construction works within the category of bounded complete cpos just for $(\cdot)^\sharp$ and $(\cdot)^\flat$. To preserve bound completeness of $D^\natural$ we have to consider more complex structures, e.g. **SFP** objects (see [43]). The second fact, which is connected to the construction to be illustrated in the sequel, is pointed out by the following proposition.

**Proposition 4.4** *For any algebraic* **CPO** *$D$ and* **NDA** *$E$*

$$D \to_{cont} E \simeq D^* \to_{lin} E \qquad \textit{for } * \in \{\flat, \sharp, \natural\}$$

*where* $\to_{cont}$ *refers to continuous functions and* $\simeq$ *is an isomorphism in the category of* **CPO**.

*Proof.* The isomorphism is given by $\mathsf{ext}$ and $\lambda g. g \circ \{\!|\cdot|\!\}$: from the universal property claimed above we immediately have that $\mathsf{ext}(f) \circ \{\!|\cdot|\!\} = f$. On the other hand, if $h = g \circ \{\!|\cdot|\!\}$ for $g : D^* \to_{lin} E$, then, for $u = \{x_1, \ldots, x_n\} \in M(D)$,

$$\begin{array}{rcl} \mathsf{ext}(h)(\downarrow u) & = & \bar{h}(u) \\ & = & h(x_1) + \cdots + h(x_n) \\ & = & g(\{\!|x_1|\!\}) + \cdots + g(\{\!|x_n|\!\}) \\ & = & g(\{\!|x_1|\!\} \uplus \cdots \uplus \{\!|x_n|\!\}) \\ & = & g(\downarrow u) \end{array}$$

where $\downarrow u = \{y \mid \exists x \in u.\, y \sqsubseteq x\}$. It follows that $\mathsf{ext}(g \circ \{\!|\cdot|\!\}) = g$. That these functions are order preserving and reversing is routine.

□

Of course this doesn't imply that the three objects $D^* \to_{lin} E$ are isomorphic as **NDA** objects: they are isomorphic just as cpos.

## 4.2 Semilinear Applicative Structures

We are now able to introduce the special kind of applicative strucure we need.

**Definition 4.5** *A* semilinear applicative structure *is a triple* $\langle X, \cdot, + \rangle$ *such that*

*i)* $\langle X, \cdot \rangle$ *is an applicative structure,*

*ii)* $+ : X^2 \to X$ *is an idempotent, commutative and associative operation,* *iii)* $\forall x, y, z \in X.\, (x + y) \cdot z = (x \cdot z) + (y \cdot z)$.

*A* linear applicative structure *is a semilinear applicative structure satisfying:*

*iv)* $\forall x, y, z \in X.\, x \cdot (y + z) = (x \cdot y) + (x \cdot z)$.

*Both semilinear and linear applicative structures are* extensional *if they are such as applicative structures, i.e.*

*v)* $\forall x, y \in X.\, (\forall z \in X.\, x \cdot z = y \cdot z) \Rightarrow x = y$.

It is used the word *semilinear* since in general the application is not right distributive with respect to the sum: i.e. it is not linear. This is due to the fact that the application will be used to model continuous functions whose arguments are "sets", that is sums, and it is not true in general that the value of these functions is the set of their values on the "elements" of the argument.

To model our calculus we shall construct an applicative structure in the category **SNDA**, in consequence of the fact that, operationally and algebraically, we have that $M \oplus N$ is less then $M$.

The usual method to construct applicative structures in the category of cpos is solving the equation $D = D \to D$, say by an inverse limit construction. The trouble is that the applicative structure arising from such a construction is surely linear. On the contrary the study of the operational and algebraic semantics in the previous sections suggests that the equation $(M \oplus N)L = ML \oplus NL$ should obtain in the model, but $M(N \oplus L) = MN \oplus ML$ should not. Therefore we have to find a semilinear applicative structure which is not a linear one.

The solution we propose is as follows: first we solve the equation $D = D^\sharp \to_{lin} D$ in the category **SNDA** via an isomorphism $\Phi$ (recall that powerdomain functors are $\mathcal{O}$-functors and locally continuous in the sense of [45], so that the projection method works); second, we define a binary continuous operation $\cdot : D \times D \to D$ as $d \cdot e = \Phi(d)\{\!|e|\!\}$. Now being $\Phi$ an isomorphism of **SNDA** it is linear itself, that is $\Phi(d + d') = \Phi(d) + \Phi(d')$, hence $(d + d') \cdot e = (d \cdot e) + (d' \cdot e)$. On the other hand, being $\{\!|e + e'|\!\}$ strictly included in $\{\!|e|\!\} \uplus \{\!|e'|\!\}$ in $D^\sharp$ as soon as $e$ and $e'$ are distinct in $D$, we have

$$d \cdot (e + e') = \Phi(d)\{\!|e + e'|\!\} \leq \Phi(d)(\{\!|e|\!\} \uplus \{\!|e'|\!\}) = \Phi(d)\{\!|e|\!\} + \Phi(d)\{\!|e'|\!\} = (d \cdot e) + (d \cdot e')$$

i.e. an inequality which depends on $d$ and in general is strict. Indeed this implies that representable functions will be continuous, but in general not linear (see proposition 4.4).

**Definition 4.6** *Take $D_0$ as any non trivial Smyth algebra (e.g. $(\mathbf{2})^\sharp$), and $D_{n+1} = [(D_n)^\sharp \to_{lin} D_n]$; then inductively define $\varphi_n : D_n \to_{lin} D_{n+1}$ and $\psi_n : D_{n+1} \to_{lin} D_n$ as follows:*

*i)* $\varphi_0(x) = \lambda y.x,\ \psi_0(y) = y(\bot)$,

*ii)* $\varphi_{n+1}(x) = \varphi_n \circ x \circ (\psi_n)^\sharp,\ \psi_{n+1}(y) = \psi_n \circ y \circ (\varphi_n)^\sharp$.

**Proposition 4.7** *The mappings $\varphi_n$, $\psi_n$ are well defined, that is they are linear; furthermore for each natural number $n$ $\langle \varphi_n, \psi_n \rangle$ is an embedding-projection pair, that is*

*i)* $\psi_n \circ \varphi_n = Id_n$;

*ii)* $\varphi_n \circ \psi_n \sqsubseteq Id_n$.

*Proof.* Linearity is proved by induction on $n$: for $n = 0$

$$\varphi_0(x + y)(z) = z = z + z = \varphi_0(x)(z) + \varphi_0(y)(z);$$

and

$$\psi_0(f + g) = (f + g)(\bot) = f(\bot) + g(\bot) = \psi_0(f) + \psi_0(g).$$

If $n > 0$ then the thesis follows from the inductive hypothesis, since for any (continuous) $f$, $f^\sharp$ is always linear and composition of linear functions is linear. To prove that $\langle \varphi_n, \psi_n \rangle$ is an embedding-projection pair we again make induction on $n$. If $n = 0$:

$$\psi_0 \circ \varphi_0(x) = \psi_0(\lambda y.x) = (\lambda y.x)(\bot) = x$$

and

$$\varphi_0 \circ \psi_0(f) = \varphi_0(f(\bot)) = \lambda z.f(\bot) \sqsubseteq f.$$

For the inductive step:

$$\begin{aligned}
\psi_{n+1} \circ \varphi_{n+1} &= \psi_{n+1}(\varphi_n \circ x \circ (\psi_n)^\sharp) \\
&= \psi_n \circ (\varphi_n \circ x \circ (\psi_n)^\sharp) \circ (\varphi_n)^\sharp \\
&= (\psi_n \circ \varphi_n) \circ x \circ (\psi_n \circ \varphi_n)^\sharp && \text{by prop. 4.3} \\
&= Id_n \circ x \circ (Id_n)^\sharp && \text{by ind. hyp.} \\
&= Id_n \circ x \circ Id_{D_n^\sharp} && \text{by prop. 4.3} \\
&= x
\end{aligned}$$

and (ii) is proved similarly, this time using the $\mathcal{O}$-functoriality of $(\cdot)^\sharp$. □

**Definition 4.8**

*i)* $D_* = \lim_{\leftarrow}(D_n, \psi_n) = \{x \in \prod_n D_n \mid \forall n \in \omega.\ \psi_n(x_{n+1}) = x_n\}$ *with coordinate-wise ordering;*

*ii)* $\Phi_{m,n} : D_m \to_{lin} D_n$ *is defined:*

$$\Phi_{m,n} = \begin{cases} \varphi_{n-1} \circ \ldots \circ \varphi_m & \textit{if } m < n \\ Id & \textit{if } m = n \\ \psi_n \circ \ldots \circ \psi_{m-1} & \textit{if } n < m \end{cases}$$

*iii)* $\Phi_{*,n} : D_* \to_{lin} D_n$ *and* $\Phi_{n,*} : D_n \to_{lin} D_*$ *are defined:*

$$\begin{array}{rcl} \Phi_{*,n}(x) & = & x_n \\ \Phi_{n,*}(y) & = & \langle \Phi_{n,m}(y) \rangle_{m \in \omega} \end{array}$$

As usual with inverse limit constructions, each $D_n$ embeds into $D_*$, by the $\langle \Phi_{n,*}, \Phi_{*,n} \rangle$ embedding-projection pair; we will write $x_n = \Phi_{n,*} \circ \Phi_{*,n}(x)$ for $x \in D_*$ and $a_n = (\Phi_{n,*} \circ \Phi_{*,n})^\sharp(a) = \Phi^\sharp_{n,*} \circ \Phi^\sharp_{*,n}(a)$ for $a \in D^\sharp_*$

Once we have the family of functions of the above definition we can explicitly define the desired isomorphism, together with the application.

**Definition 4.9**

*i) The maps* $F : D_* \to [D^\sharp_* \to_{lin} D_*]$ *and* $\tilde{F} : D_* \to [D_* \to_{cont} D_*]$ *are defined by:*

$$F(x) = \lambda a \in D^\sharp_*. \bigsqcup_n x_{n+1}(a_n) \textit{ and } \tilde{F} = (\lambda g. g \circ \{\!| \cdot |\!\}) \circ F.$$

*ii) The maps* $G : [D^\sharp_* \to_{lin} D_*] \to D_*$ *and* $\tilde{G} : [D_* \to_{cont} D_*] \to D_*$ *are defined by:*

$$G(f) = \bigsqcup_n (\lambda a \in D^\sharp_n . (f(\Phi^\sharp_{n,*}(a)))_n) \textit{ and } \tilde{G} = G \circ \mathsf{ext}.$$

*iii) The operation of application* $\cdot : D \times D \to D$ *is defined by:*

$$x \cdot y = \tilde{F}(x)(y) = F(x)\{\!| y |\!\}.$$

We list in the following lemma some relevant properties of the domain $D_*$

**Lemma 4.10** *For any* $x, y, z \in D_*$ *and* $a \in D^\sharp_*$,

*i)* $(x_m)_n = x_{min(n,m)}$;

*ii)* $x = \bigsqcup_n x_n$, $a = \bigsqcup_n a_n$;

*iii)* $x_{n+1} \cdot y_n = x_{n+1}(\{\!| y_n |\!\})$;

*iv)* $x_{n+1} \cdot y = x_{n+1} \cdot y_n = (x \cdot y_n)_n$;

*v)* $x_0 \cdot y = x_0 = (x \cdot \bot)_0$;

*vi)* $(x+y)_n = x_n + y_n = (x_n + y_n)_n$.

*Proof.*

(i) Consequence of the fact that $\langle \Phi_{n,*}, \Phi_{*,n} \rangle$ is an embedding-projection pair.

(ii) $x = \bigsqcup_n x_n$ is standard in inverse limit constructions; to see $a = \bigsqcup_n a_n$:

a) Let $\langle \varphi, \psi \rangle$ be an injection-projection pair from some $D$ to some $E$: then

$$\varphi(\mathcal{K}(D)) \subseteq \mathcal{K}(E).$$

Indeed, let $x \in \mathcal{K}(D)$ then for any directed $Y \subseteq E$

$$\begin{array}{rcl} \varphi(x) \sqsubseteq \bigsqcup Y & \Rightarrow & x \sqsubseteq \psi(\bigsqcup Y) = \bigsqcup \psi(Y) \\ & \Rightarrow & \exists y \in Y.\, x \sqsubseteq \psi(y) \\ & \Rightarrow & \exists y \in Y.\, \varphi(x) \sqsubseteq \varphi \circ \psi(y) \sqsubseteq y. \end{array}$$

b) $\mathcal{K}(D_*) = \bigcup_n \mathcal{K}(D_n)$: indeed if $d \in \mathcal{K}(D_n)$ then $d \in \mathcal{K}(D_*)$ follows from (a) and the fact that $\langle \Phi_{n,*}, \Phi_{*,n} \rangle$ is an embedding-projection pair; on the other hand, from $d \in \mathcal{K}(D_*)$ it follows

$$d = \bigsqcup_n d_n \Rightarrow \exists m.\, d = d_m.$$

Now, given any directed $S \subseteq D_m$

$$\begin{array}{rcl} \Phi_{*,m}(d) \sqsubseteq \bigsqcup S & \Rightarrow & d = \Phi_{m,*} \circ \Phi_{*,m}(d) \sqsubseteq \Phi_{m,*}(\bigsqcup S) \\ & \Rightarrow & d \sqsubseteq \bigsqcup \Phi_{m,*}(S) \\ & \Rightarrow & \exists s \in S.\, d \sqsubseteq \Phi_{m,*}(s) \\ & \Rightarrow & \exists s \in S.\, \Phi_{*,m}(d) \sqsubseteq s. \end{array}$$

That $a_n \sqsubseteq a$ for all $n$ is immediate. Vice versa, let $u \in a$, then $u = \{d^1, \ldots, d^r\} \in \mathrm{M}(D_*)$; using (b) we know that each $d_i$ is compact in some $D_{m_i}$, then we choose $m = max\{m_i \mid 1 \leq i \leq r\}$: by (a) $u \in \mathrm{M}(D_m)$. On the other hand

$$\Phi^\sharp_{*,m}(a) = \bigcup \{\bar{\Phi}_{*,m}(v) \mid v \in a\}$$

but

$$\begin{array}{rcl} \bar{\Phi}_{*,m}(u) & = & \{\!|\Phi_{*,m}(d^1)|\!\} \uplus \ldots \uplus \{\!|\Phi_{*,m}(d^r)|\!\} \\ & = & \{\!|d^1|\!\} \uplus \ldots \uplus \{\!|d^r|\!\} \\ & = & \downarrow u \end{array}$$

so that $u \in \downarrow u \subseteq a_m$, from which we conclude that $a \sqsubseteq \bigsqcup_n a_n$.

(iii) Let us note preliminarily that, by the very definition of $(\cdot)^\sharp$:

$$\{\!|y|\!\}_n = \Phi^\sharp_{*,n}\{\!|y|\!\} = \{\!|\Phi_{*,n}(y)|\!\} = \{\!|y_n|\!\}.$$

Now

$$\begin{array}{rcl} x_{n+1}\cdot y_n & = & F(x_{n+1})\{\!|y_n|\!\} \\ & = & \bigsqcup_m (x_{n+1})_{m+1}\{\!|y_n|\!\}_m \\ & = & \bigsqcup_m (x_{n+1})_{m+1}(\{\!|y|\!\}_n)_m \\ & = & x_{n+1}\{\!|y|\!\}_n \\ & = & x_{n+1}\{\!|y_n|\!\}. \end{array}$$

(iv)-(v) Similar to the proof of the corresponding properties for Scott $D_\infty$ models.

(vi) By linearity of $\Phi^\sharp_{*,n}$ we immediately have $(x+y)_n = x_n + y_n$. On the other hand

$$\begin{array}{rcl} (x_n+y_n)_n & = & (x_n)_n + (y_n)_n \\ & = & x_n + y_n. \end{array}$$

□

**Lemma 4.11** *The mappings $F$ and $G$ are continuous and linear, that is they are* **NDA** *morphisms. Furthermore the structure*

$$\langle D_*, \cdot, + \rangle$$

*is a semilinear applicative structure.*

*Proof.* Let $x, y \in D_*$ and $a \in D^\sharp_*$, then

$$\begin{array}{rcll} \bigsqcup_n (x+y)_{n+1}(a_n) & = & \bigsqcup_n (x_{n+1}+y_{n+1})(a_n) & \text{by lemma 4.10 (vi)} \\ & = & \bigsqcup_n (x_{n+1}(a_n) + y_{n+1}(a_n)) & \\ & = & \bigsqcup_n x_{n+1}(a_n) + \bigsqcup_n y_{n+1}(a_n) & \text{by continuity of } + \end{array}$$

hence

$$\begin{array}{rcl} F(x+y) & = & \lambda a \in D^\sharp_*. \bigsqcup (x+y)_{n+1}(a_n) \\ & = & \lambda a \in D^\sharp_*. \bigsqcup_n x_{n+1}(a_n) + \bigsqcup_n y_{n+1}(a_n) \\ & = & (\lambda a \in D^\sharp_*. \bigsqcup_n x_{n+1}(a_n)) + (\lambda a \in D^\sharp_*. \bigsqcup_n y_{n+1}(a_n)) \\ & = & F(x) + F(y). \end{array}$$

Let $f, g \in [D^\sharp_* \to_{lin} D_*]$ and $a \in D^\sharp_*$, then

$$((f+g)(a))_n = (f(a)+g(a))_n = (f(a))_n + (g(a))_n$$

by lemma 4.10 (vi); it follows

$$\begin{array}{rcl} G(f+g) & = & \bigsqcup_n (\lambda b \in D^\sharp_n.((f+g)(\Phi^\sharp_{n,*}(b)))_n) \\ & = & \bigsqcup_n (\lambda b \in D^\sharp_n.(f(\Phi^\sharp_{n,*}(b)))_n + (g(\Phi^\sharp_{n,*}(b)))_n) \\ & = & \bigsqcup_n ((\lambda b \in D^\sharp_n.(f(\Phi^\sharp_{n,*}(b)))_n) + (\lambda b \in D^\sharp_n.(g(\Phi^\sharp_{n,*}(b)))_n)) \\ & = & \bigsqcup_n (\lambda b \in D^\sharp_n.(f(\Phi^\sharp_{n,*}(b)))_n) + \bigsqcup_n (\lambda b \in D^\sharp_n.(g(\Phi^\sharp_{n,*}(b)))_n) \\ & = & G(f) + G(g). \end{array}$$

This establishes the linearity property; the continuity property is proved in the same way as in the category of **CPO**.
Finally, let $x, y, z \in D_*$:

$$\begin{array}{rcll} (x+y)\cdot z & = & \tilde{F}(x+y)(z) & \\ & = & F(x+y)(\{\!|z|\!\}) & \\ & = & (F(x)+F(y))(\{\!|z|\!\}) & \text{by linearity of } F \\ & = & F(x)(\{\!|z|\!\}) + F(y)(\{\!|z|\!\}) & \\ & = & (x\cdot z)+(y\cdot z). & \end{array}$$

□

**Theorem 4.12** *The domain $D_*$ satisfies the equation*

$$D \simeq [D^\sharp \to_{lin} D]$$

*in the category of* **NDA** *and consequently in that of* **SNDA**; *it satisfies also the equation*

$$D \simeq [D \to_{cont} D]$$

*in the category of* **CPO** *as pictured in the diagram*

$$D_* \underset{F}{\overset{G}{\leftrightarrows}} [D_*^\sharp \to_{lin} D_*] \underset{\lambda g.g \circ \{\!|\cdot|\!\}}{\overset{\mathsf{ext}}{\leftrightarrows}} [D_* \to_{cont} D_*]$$

*We conclude that* $\langle D_*, \cdot, + \rangle$ *is an* extensional *semilinear applicative structure.*

*Proof.* To prove the theorem it remains to show that $F$ and $G$ are mutually inverse: actually the second isomorphism will follow from this one and from corollary 4.4, which applies to the $(\cdot)^\sharp$ functor as well.

a) $G \circ F = Id$: by definition $(G \circ F)(x) = G(f)$ where

$$f = \lambda a \in D_*^\sharp . \bigsqcup_n x_{n+1}(a_n);$$

now we observe that if $y$ is in (the image of) $D_n$ the $y = y_n$, and similarly if $a$ is in the image if $D_n^\sharp$; now given such an $a$

$$\begin{array}{rcl} (f(a))_n & = & (\bigsqcup_m x_{m+1}(a_m))_n \\ & = & (x_{n+1}(a))_n \\ & = & x_{n+1}(a). \end{array}$$

It follows that

$$\begin{array}{rcl} G(f) & = & \bigsqcup_n(\lambda a \in D_n^\sharp . x_{n+1}(a)) \\ & = & \bigsqcup_n x_{n+1} \\ & = & x. \end{array}$$

b) $F \circ G = Id$: we note that

$$\begin{array}{rcl} G(f)_{n+1}(a_n) & = & (f(\Phi^{\sharp}_{n,*}(a_n)))_n \\ & = & (f(a))_n \end{array}$$

so that

$$\begin{array}{rcl} (F \circ G)(f)(a) & = & \bigsqcup_n (f(a))_n \\ & = & f(a), \end{array}$$

that is $(F \circ G)(f) = (f)$.

To prove extensionality:

$$\begin{array}{rcl} \forall z.\, x \cdot z = y \cdot z & \Rightarrow & \tilde{F}(x)(z)\tilde{F}(y)(z) \\ & \Rightarrow & \tilde{F}(x) = \tilde{F}(y) \\ & \Rightarrow & x = \tilde{G} \circ \tilde{F}(x) = \tilde{G} \circ \tilde{F}(y) = y. \end{array}$$

□

### 4.3 Syntactical Models

We present a notion of *model*, which actually does not directly interpret the relation $\longrightarrow$, but the equivalence relation $\simeq_{must}$. More precisely we extend the classical notion of syntactical model of lambda calculus (see [26]) to the case of the non deterministic calculus and show that the algebra $D_*$, together with a suitable interpretation map, is an instance of such a model.

**Definition 4.13** *A* syntactical model *is a semilinear applicative structure* $\mathcal{M} = \langle X, \cdot, + \rangle$ *equipped with a map* $[\![\cdot]\!] : \Lambda_\oplus \to (Env \to X)$, *such that the triple* $\langle X, \cdot, [\![\cdot]\!] \rangle$, *for any* $\rho \in Env = Var \to X$, *satisfies:*

*i)* $[\![x]\!]_\rho = \rho(x)$;

*ii)* $[\![MN]\!]_\rho = [\![M]\!]_\rho \cdot [\![N]\!]_\rho$;

*iii)* $[\![\lambda x.M]\!]_\rho \cdot d = [\![M]\!]_{\rho[d/x]}$ *for all* $d \in X$;

*iv)* $\rho \lceil \mathrm{FV}(M) = \rho' \lceil \mathrm{FV}(M) \Rightarrow [\![M]\!]_\rho = [\![M]\!]_{\rho'}$;

*v)* $[\![\lambda x.M]\!]_\rho = [\![\lambda y.M[y/x]]\!]_\rho$ *if* $y \notin \mathrm{FV}(M)$;

*vi)* $(\forall d \in X.\, [\![M]\!]_{\rho[d/x]} = [\![N]\!]_{\rho[d/x]}) \Rightarrow [\![\lambda x.M]\!]_\rho = [\![\lambda x.N]\!]_\rho$;

*which are the clauses of the classical definition of syntactical* $\lambda$*-model of* [26], *and furthermore*

*vii)* $[\![M \oplus N]\!]_\rho = [\![M]\!]_\rho + [\![N]\!]_\rho$.

*Finally we call* extensional *any syntactical model whose underlying semilinear applicative structure is extensional.*

**Lemma 4.14** *If $\langle X, \cdot, +, [\![\cdot]\!]\rangle$ is an extensional syntactical model, then for any $M, N \in \Lambda_\oplus$ and for all $\rho \in Env$:*

$$[\![\lambda x.M \oplus N]\!]_\rho = [\![(\lambda x.M) \oplus (\lambda x.N)]\!]_\rho.$$

*Proof.* Let $d \in X$ be an arbitrary element; then, for any $\rho \in Env$,

$$\begin{array}{rcll} [\![\lambda x.M \oplus N]\!]_\rho \cdot d &=& [\![M \oplus N]\!]_{\rho[d/x]} & \text{by def. 4.13 (iii)} \\ &=& [\![M]\!]_{\rho[d/x]} + [\![N]\!]_{\rho[d/x]} & \text{by def. 4.13 (vii)} \\ &=& [\![\lambda x.M]\!]_\rho \cdot d + [\![\lambda x.N]\!]_\rho \cdot d & \text{by def. 4.13 (iii)} \\ &=& ([\![\lambda x.M]\!]_\rho + [\![\lambda x.N]\!]_\rho) \cdot d & \text{by semilinearity.} \end{array}$$

Since $d$ is arbitrary, it follows that

$$\begin{array}{rcll} [\![\lambda x.M \oplus N]\!]_\rho &=& [\![\lambda x.M]\!]_\rho + [\![\lambda x.N]\!]_\rho & \text{by def. 4.13 (vi)} \\ &=& [\![(\lambda x.M) \oplus (\lambda x.N)]\!]_\rho & \text{by def. 4.13 (vii).} \end{array}$$

□

**Definition 4.15** *Given the structure $\langle D_*, \cdot, +\rangle$ and $\rho \in Env = Var \to D_*$, we define the map $[\![\cdot]\!] : \Lambda_\oplus \to (Env \to D_*)$ as follows:*

*i)* $[\![x]\!]_\rho = \rho(x)$,

*ii)* $[\![MN]\!]_\rho = [\![M]\!]_\rho \cdot [\![N]\!]_\rho$,

*iii)* $[\![\lambda x.M]\!]_\rho = \tilde{G}(\lambda d \in D_*.[\![M]\!]_{\rho[d/x]})$,

*iv)* $[\![M \oplus N]\!]_\rho = [\![M]\!]_\rho + [\![N]\!]_\rho$.

This is a good definition, since in (iii) the continuity and linearity of application, abstraction and $+$ ensure that the function $\lambda d \in D_*.[\![M]\!]_{\rho[d/x]}$ is continuous and linear as well.

**Proposition 4.16** *The quadruple $\langle D_*, \cdot, +, [\![\cdot]\!]\rangle$ is a syntactical model, furthermore it is extensional.*

*Proof.* By theorem 4.12 the structure $\langle D_*, \cdot, +\rangle$ is an extensional semilinear applicative structure. The rest is routine; e.g.

$$\begin{array}{rcl} [\![\lambda x.M]\!]_\rho \cdot d &=& \tilde{F}(\tilde{G}(\lambda d'.[\![M]\!]_{\rho[d'/x]}))(d) \\ &=& (\lambda d'.[\![M]\!]_{\rho[d'/x]})(d) \\ &=& [\![M]\!]_{\rho[d/x]}, \end{array}$$

hence definition 4.13 (iii) is verified.

□

## 5 Full Abstraction

The main result of this section is a theorem stating that the operational and denotational semantics constructed so far coincide. Such kind of result is an instance of that property which is called in the literature "full abstraction property" (see [44, 34]).

In effect it could be questioned whether our reduction relation is actually "operational" since it doesn't formalize an evaluation mechanism, that is it doesn't force any reduction strategy.

In such a case one should refer to the main result of this section as a characterization theorem of the local structure of the model $D_*$ (see [9] where the local structure of a $\lambda$-model is its equational theory, as opposed to the global structure of properties which are not equationally expressible).

However, in view of the fact that we put the emphasis of our treatment on the notion of head reduction, which actually is an evaluation mechanism, we feel entitled to speak of full abstraction.

**Theorem 5.1 (Full Abstraction Theorem)**
*For all* $M, N \in \Lambda_\oplus$

$$M \sqsubseteq_{must} N \;\Leftrightarrow\; \forall \rho.\, [\![M]\!]_\rho \sqsubseteq [\![N]\!]_\rho.$$

To prove the theorem we shall use ideas from classical $\lambda$-calculus.

Following Wadsworth, Hyland and Levy (see [28, 32, 51]), we introduce an indexing notion $\mathcal{I}$ assigning a natural number to each term. This corresponds semantically to the projection of the denotation of any term $M$ in the algebra $D_{\mathcal{I}(M)}$, so that the denotation of $M$ is the limit of all its denotations under any indexing function $\mathcal{I}$.

It turns out that the set of interpretations of the terms $M^{[k]}$ has a cofinality property with respect to the set of interpretations of $M^{\mathcal{I}}$, giving us the main lemma to establish the theorem.

In the sequel the intended interpretation is $D_*$.

**Definition 5.2** *Let $\mathcal{I}$ be an* indexing function, *that is a map $\mathcal{I} : \Lambda_\oplus\Omega \to \mathbb{N}$; then, writing $M^{\mathcal{I}}$ to denote the term $M$ labelled with the index $\mathcal{I}(M)$, $[\![M^{\mathcal{I}}]\!]$ is defined:*

*i)* $[\![\Omega^{\mathcal{I}}]\!]_\rho = \bot_{\mathcal{I}(\Omega)} = \bot$;

*ii)* $[\![x^{\mathcal{I}}]\!]_\rho = (\rho(x))_{\mathcal{I}(x)}$;

*iii)* $[\![(MN)^{\mathcal{I}}]\!]_\rho = ([\![M^{\mathcal{I}}]\!]_\rho \cdot [\![N^{\mathcal{I}}]\!]_\rho)_{\mathcal{I}(MN)}$;

*iv)* $[\![(\lambda x.M)^{\mathcal{I}}]\!]_\rho = (\tilde{G}(\lambda d.[\![M^{\mathcal{I}}]\!]_{\rho[d/x]}))_{\mathcal{I}(\lambda x.M)}$;

*v)* $[\![(M \oplus N)^{\mathcal{I}}]\!]_\rho = ([\![M^{\mathcal{I}}]\!]_\rho + [\![N^{\mathcal{I}}]\!]_\rho)_{\mathcal{I}(M\oplus N)}$.

**Lemma 5.3** *For any $M \in \Lambda_\oplus$ and all $\rho \in Env$:*

$$[\![M]\!]_\rho = \bigsqcup_{\mathcal{I}} [\![M^{\mathcal{I}}]\!]_\rho.$$

*Proof.* By induction on $M$ using the equation $x = \bigsqcup_n x_n$ of lemma 4.10. □

In the sequel we call terms together with their indexes modulo some indexing function *indexed terms*. For the sake of symbolic manipulation we allow multiple indexed terms e.g. $(M^n)^m$, whose intended meaning is $M^{\min(n,m)}$, and make this minimalization over indexes into an explicit reduction step (see [9] chapter 14).

**Definition 5.4** *First extend the definition of substitution to indexed terms inductively from the base clause $x^m[N^n/x] \equiv (N^n)^m$. Now define the following binary relation $\rhd$ over indexed terms:*

| | | | |
|---|---|---|---|
| *i)* | $(\lambda x.M)^{n+1}N \rhd (M[N^n/x])^n$; | *vii)* | $M \oplus \Omega^n \rhd \Omega^0$; |
| *ii)* | $(\lambda x.M)^0 N \rhd (M[\Omega^0/x])^0$; | *viii)* | $(M \oplus N)^{n+1}L \rhd (ML^n \oplus NL^n)^n$; |
| *iii)* | $\Omega^n \rhd \Omega^0$; | *ix)* | $(M \oplus N)^0 L \rhd (M\Omega^0 \oplus N\Omega^0)^0$; |
| *iv)* | $\lambda x.\Omega^n \rhd \Omega^0$; | *x)* | $(M^m)^n \rhd M^{min(m,n)}$; |
| *v)* | $\Omega^n M \rhd \Omega^0$; | *xi)* | $M^m \rhd N^n \Rightarrow C[M^m] \rhd C[N^n]$. |
| *vi)* | $\Omega^n \oplus M \rhd \Omega^0$; | | |

**Lemma 5.5** *The relation $\rhd$ is strongly normalizing and Church-Rosser.*

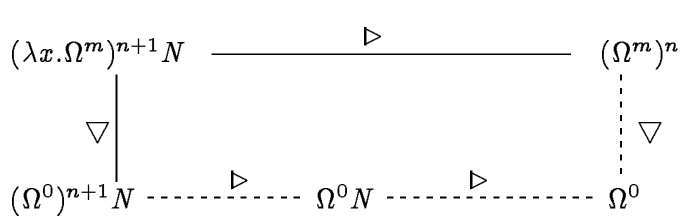
*Proof.* An easy extension of the classical proof of strong normalization of the labelled $\lambda$-calculus (see [9]) establishes the strong normalizability property: just note the decreasing indexes in clauses (i) and (viii) of the Def. 5.4, and that the length of the term decreases in the other cases.

Now by case inspection of overlapping right hand sides in the definition 5.4 one sees that $\rhd$ is weakly Church-Rosser that is

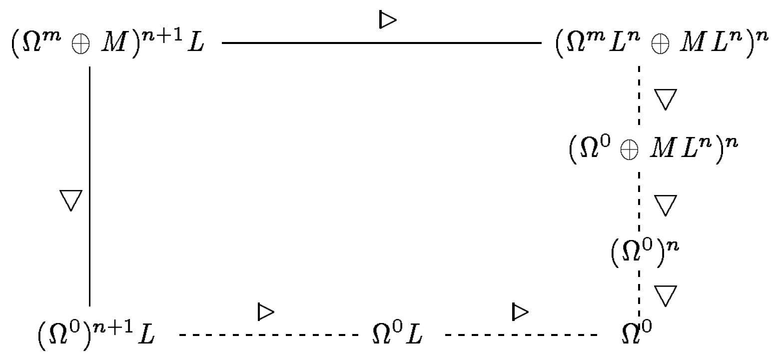
$$M^n \rhd N^p \ \wedge \ M^n \rhd L^q \Rightarrow \exists T^r.\ N^p \rhd^* T^r \ \wedge \ L^q \rhd^* T^r.$$

We illustrate two relevant cases.

Case 1: (i)-(iv)



Case 2: (iii)-(viii)




Hence $\rhd$ is Church-Rosser by Newman lemma. □

**Corollary 5.6**

*i)* $\mathrm{N}_{\oplus}^{\Omega} = \{|M| \mid M$ *in* $\rhd$*-normal form*$\}$, *where* $|\cdot|$ *is the index erasing map;*

*ii)* $\forall M \in \Lambda_{\oplus}\, \forall \mathcal{I}\, \exists N \in \mathrm{N}_{\oplus}^{\Omega}\, \exists \mathcal{J}.\, M^{\mathcal{I}} \rhd N^{\mathcal{J}}$.

*Proof.* Recall that $\mathrm{N}_{\oplus}^{\Omega} = \{M^{[k]} = \vartheta_1 \circ \omega^k(M) \mid M \in \Lambda_{\oplus}, k \in \mathbb{N}\}$. Let us observe that, after the very definition of $\vartheta_1$, this set could be inductively defined by:

i) $\Omega \in \mathrm{N}_{\oplus}^{\Omega}$;

ii) $M_1, \ldots, M_m \in \mathrm{N}_{\oplus}^{\Omega} \wedge x_1, \ldots, x_n, x \in Var \Rightarrow \lambda x_1 \ldots x_n.xM_1 \ldots M_m \in \mathrm{N}_{\oplus}^{\Omega}$;

iii) $M, N \in \mathrm{N}_{\oplus}^{\Omega} - \{\Omega\} \Rightarrow M \oplus N \in \mathrm{N}_{\oplus}^{\Omega}$;

now to prove (i) is routine. (ii) follows from (i) and the previous lemma. □

**Lemma 5.7**

$$M \in \mathrm{N}_{\oplus}^{\Omega} \Rightarrow \exists k \in \mathbb{N}\, \forall h \geq k.\, \omega^h(M) = \omega^k(M);$$

*hence, defining* $\mathsf{height}(M)$ *as the minimal* $k$ *satisfying the above statement,*

$$M \leq N \iff M \leq_{\mathsf{height}(M)} N \iff M \leq N^{[\mathsf{height}(M)]}.$$

*Proof.* Note that, since $M \in \mathrm{N}^{\Omega}_{\oplus}$, $NBT(M)$ differs from the syntactical tree only in that the operator $\oplus$ is treated as a set constructor, and some abstractions are pushed into sums: e.g.

$$\omega^k(\lambda x.P \oplus Q) = \omega^k(\lambda x.P) \cup \omega^k(\lambda x.Q) = \omega^k(\lambda x.P \oplus \lambda x.Q).$$

Now take as $k$ the depth of the syntactical tree of $M$. □

We state here the main lemma to prove the full abstraction theorem; its proof is deferred to section 7.

**Lemma 5.8** *For any $M \in \Lambda_{\oplus}$ and natural number $k$,*

$$\forall \rho \in Env.\, [\![M]\!]_\rho = \bigsqcup_k [\![M^{[k]}]\!]_\rho.$$

**Theorem 5.9** *For all $M, N \in \Lambda_{\oplus}$,*

$$\forall \rho \in Env.\, [\![M]\!]_\rho \sqsubseteq [\![N]\!]_\rho \Rightarrow M \sqsubseteq_{must} N.$$

*Proof.*

$$\begin{array}{rcl} M^{[1]} = \Omega & \Leftrightarrow & \forall k.\, M^{[k]} = \Omega \\ & \Leftrightarrow & \forall k.\, [\![M^{[k]}]\!] = \bot \\ & \Leftrightarrow & [\![M]\!] = \bigsqcup_k [\![M^{[k]}]\!] = \bot \quad \text{by lemma 5.8} \end{array}$$

since $[\![\Omega]\!] = \bot$; hence

$$\begin{array}{rcl} M \not\sqsubseteq_{must} N & \Rightarrow & \exists C[\,].\, C[M]\!\downarrow \wedge C[N]\!\uparrow \\ & \Rightarrow & \exists C[\,].\, \omega^1(C[M]) \neq \{\Omega\} = \omega^1(C[N]) \\ & \Rightarrow & \exists C[\,].\, [\![C[M]]\!] \neq \bot = [\![C[N]]\!] \\ & \Rightarrow & [\![M]\!] \not\sqsubseteq [\![N]\!], \end{array}$$

being the context operation the composition of abstraction, application and $+$, that is a monotonic function. □

**Corollary 5.10** *For all $M, N \in \Lambda_{\oplus}$,*

*i)* $M \leq N \Rightarrow [\![M]\!] \sqsubseteq [\![N]\!]$,

*ii)* $M \sqsubseteq_{must} N \Leftrightarrow M \leq N \Leftrightarrow [\![M]\!] \sqsubseteq [\![N]\!]$.

*Proof.* To establish (i):

$$\begin{array}{rcl} M \leq N & \Rightarrow & \forall k.\, M^{[k]} \leq M \leq N \\ & \Rightarrow & \forall k.\, [\![M^{[k]}]\!] \sqsubseteq [\![N]\!] \\ & \Rightarrow & [\![M]\!] \sqsubseteq [\![N]\!]. \end{array}$$

Now (ii) follows from (i) and theorems 3.24 and 5.9. □

## 6 Non Deterministic Theories and Conservativity

In this section we address the question of relationship between the non deterministic calculus studied so far and the classical one, seen as its deterministic subcalculus.

In [5] a non deterministic extension of Plotkin's *PCF* was introduced, consisting in a simply typed $\lambda$-calculus with recursion and a choice operator. The authors proposed an operational semantics based on a reduction relation forcing outermost evaluation, and a denotational semantics in which powerdomain functors were used just to interpret basic types, functional types being restricted to linear (that is additive) functions over non deterministic algebras.

As observed in [6], this results in a theory of the equivalence of terms which doesn't preserve the equivalences holding in the deterministic subcalculus. Indeed to recover such a conservativity property while preserving an additive semantics, Astesiano and Costa were led in [6] to an operational semantics based on a rewriting system, which follows an outermost strategy with sharing mechanism for arguments.

It should be noted that this is a high cost solution: this is due not just to the complexity of both syntax and semantics, but much to the strong limitation it imposes to the power of non determinism. Indeed it corresponds to forcing a uniform behaviour of multiplied instances of the same choice redexes.

This problem can be recasted in the perspective of call-by-name and call-by-value $\lambda$-calculi, whose study has been initiated in [42], and recently received a categorical foundation in [38].

In [48] the task of finding correct non deterministic extensions of the notion of $\lambda$-theory in case of the untyped calculus has been carried out distinguishing two value passing mechanisms (namely $\beta$-rules), which are viewed as relativizing call-by-name and call-by-value to the choice operator. The first rule allows unrestricted substitution, in such a way that choices can be multiplied and performed at any time during evaluation; this has been called run-time-choice in [23]. The second rule forces passed arguments to be "deterministic", that is to perform the reduction $(\lambda x.M)N \longrightarrow M[N/x]$, $N$ needs to be free of any occurrence of the choice operator. This is called call-time-choice in [23].

**Definition 6.1** *The binary relations $\longrightarrow_r, \longrightarrow_c \subseteq \Lambda_\oplus^2$ are obtained from the following three clauses, closing under contexts:*

*($\beta_r$)* $(\lambda x.M)N \longrightarrow_r M[N/x]$;

*($\beta_c$)* $(\lambda x.M)N \longrightarrow_c M[N/x]$ *if* $N \in \Lambda$;

*($\oplus$)* $M \oplus N \longrightarrow_* M$, $M \oplus N \longrightarrow_* N$, *for* $* = r, c$.

The relation $\longrightarrow_r$ is clearly the reduction relation studied in the previous sections of this paper. The relation $\longrightarrow_c$ induces a non deterministic calculus which, as we will see, is equivalent to a calculus of finite sets of classical terms.

In the next subsections we shall consider Sharma's theories and and give new proofs of their consistency, relying on our semantical results for the case of run-time-choice theory, here called $\lambda_r$. Moreover we will prove that both $\lambda_r$ and $\lambda_c$ (that is the call-time-choice theory) are conservative extensions of the theory $\lambda$, and that the theory of the model $D_*$, we call $\mathcal{T}_{must}$, is conservative with respect to the maximal consistent $\lambda$-theory $\mathcal{H}^*$. Since $\lambda_r$ is a subtheory of $\mathcal{T}_{must}$, this can be considered as an alternative proof of its consistency.

## 6.1 Call-time Choice $\lambda$-calculus

**Definition 6.2** *The theory $\lambda_c$ is the equational theory over $\Lambda_\oplus$ whose axioms and rules are as follows*

| | | | |
|---|---|---|---|
| $(\beta_c)$ | $(\lambda x.M)N = M[N/x]$ *if* $N \in \Lambda$ | $(\zeta_1)$ | $M \oplus M = M$ |
| $(\rho)$ | $M = M$ | $(\zeta_2)$ | $M \oplus N = N \oplus M$ |
| $(\sigma)$ | $M = N \Rightarrow N = M$ | $(\zeta_3)$ | $(M \oplus N) \oplus L = M \oplus (N \oplus L)$ |
| $(\tau)$ | $M = N, N = L \Rightarrow M = L$ | $(\varepsilon)$ | $M = N \Rightarrow M \oplus L = N \oplus L$ |
| $(\mu)$ | $M = N \Rightarrow LM = LN$ | $(\delta)$ | $(M \oplus N)L = ML \oplus NL$ |
| $(\nu)$ | $M = N \Rightarrow ML = NL$ | $(\theta)$ | $L(M \oplus N) = LM \oplus LN$ |
| $(\xi)$ | $M = N \Rightarrow \lambda x.M = \lambda x.N$ | $(\gamma)$ | $\lambda x.M \oplus N = \lambda x.M \oplus \lambda x.N$ |

In [48] it was actually proved that a subtheory of $\lambda_c$ was consistent, namely the theory obtained deleting axiom $(\gamma)$. The result was established by defining a notion of reduction (different from $\longrightarrow_c$) essentially by orienting from left to right the axioms of $\lambda_c$, and then proving a Church-Rosser theorem.

The proof of the consistency theorem was however very long, and the difficulty with the axiom $(\gamma)$ couldn't be overcome. On the contrary we give here a very short proof of the consistency of the whole theory, in a way that, in our opinion, enlightens the fact that the $\lambda_c$-calculus is nothing more than a calculus of (closures under $\beta$-conversion of) finite sets of classical terms.

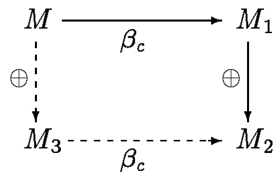
As a first step we prove a simple property of the reduction relation $\longrightarrow_c$, that fails in case of $\longrightarrow_r$.

**Notation**: We will write $\longrightarrow_{\beta_c}$ when a one step $\beta_c$-contraction occurs; similarly we write $\longrightarrow_\oplus$ when only one $\oplus$-contraction occurs. $\stackrel{*}{\longrightarrow}_{\beta_c}$ and $\stackrel{*}{\longrightarrow}_\oplus$ are their reflexive and transitive closures respectively.

**Lemma 6.3**

$$\forall M, M_1, M_2 \in \Lambda_\oplus.\ M \longrightarrow_{\beta_c} M_1 \longrightarrow_\oplus M_2 \ \Rightarrow\ \exists M_3 \in \Lambda_\oplus.\ M \longrightarrow_\oplus M_3 \longrightarrow_{\beta_c} M_2$$

*that is*




*Proof.* By induction on $M$, and then by cases. The only interesting case is when $M \equiv (\lambda x.M')M''$ and $M_1 \equiv M'[M''/x]$; in $M_1 \longrightarrow_\oplus M_2$ the only possibility is that a (residual of a) $\oplus$ redex in $M'$ is contracted, since it must be the case that $M'' \in \Lambda$. It follows that $M' \equiv C[P_1 \oplus P_2]$ for some $P_1$ and $P_2$ and $M_1 \equiv C'[P_1[M''/x] \oplus P_2[M''/x]]$ if $x$ is not bound above in $C[\,]$ and $C'[\,]$ results from $C'[\,]$ substituting $M''$ for all free occurrences of $x$; in this case $M_2 \equiv C'[P_i[M''/x]]$ for $i = 1$ or 2. Then

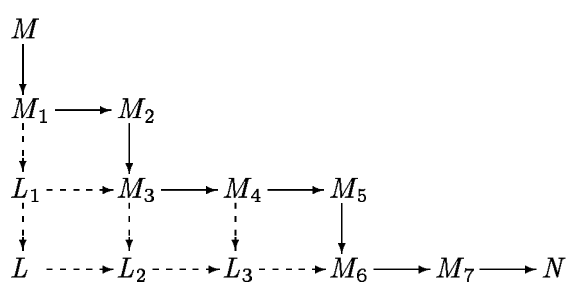
$$(\lambda x.M')M'' \longrightarrow_\oplus (\lambda x.C[P_i])M'' \longrightarrow_{\beta_c} (C[P_i])[M''/x] \equiv C'[P_i[M''/x]]$$

so that we take $M_3 \equiv (\lambda x.C[P_i])M''$. If $x$ is bound above the hole $[\,]$ in the context $C[\,]$ the proof is similar and easier. □

**Corollary 6.4**

$$\forall M, N \in \Lambda_\oplus.\ M \xrightarrow{*}_c N \Rightarrow \exists L \in \Lambda_\oplus.\ M \xrightarrow{*}_\oplus L \xrightarrow{*}_{\beta_c} N.$$

*Proof.* The proof is illustrated in the following picture, where vertical arrows represent one-step $\oplus$-reductions, horizontal arrows represent one-step $\beta_c$-reductions, and each square is an application of lemma 6.3:




□

**Definition 6.5** *Let $\mathcal{A} \subseteq \Lambda$ and $M, N \in \Lambda_\oplus$, then*

*i)* $\mathcal{A}^+ = \{M \mid \exists N \in \mathcal{A}.\ M =_\beta N\}$*;*

*ii)* $\mathsf{det}(M) = \{L \in \Lambda \mid M \xrightarrow{*}_c L\}^+$;

*iii)* $M \subseteq_c N \;\Leftrightarrow\; \mathsf{det}(M) \subseteq \mathsf{det}(N)$;

*iv)* $M =_c N \;\Leftrightarrow\; M \subseteq_c N \subseteq_c M$.

The operation $(\cdot)^+$ is the usual closure under $\beta$-conversion; the intuitive meaning of $\mathsf{det}(M)$ is "the set of deterministic values of $M$".

**Definition 6.6** *Let $\mathcal{A} \subseteq \Lambda$; $\mathcal{A}$ is $\beta$-closed iff $\mathcal{A} = \mathcal{A}^+$. Furthermore if $\mathcal{A}, \mathcal{B}$ are $\beta$-closed then*

*i)* $\mathcal{A}\mathcal{B} = \{MN \mid M \in \mathcal{A}, N \in \mathcal{B}\}^+$;

*ii)* $\lambda x.\mathcal{A} = \{\lambda x.M \mid M \in \mathcal{A}\}^+$;

*iii)* $\mathcal{A}[\mathcal{B}/x] = \{M[N/x] \mid M \in \mathcal{A}, N \in \mathcal{B}\}^+$.

**Lemma 6.7** *For any $M, N \in \Lambda_\oplus$*

*i)* $\mathsf{det}(M \oplus N) = \mathsf{det}(M) \cup \mathsf{det}(N)$;

*ii)* $\mathsf{det}(\lambda x.M) = \lambda x.\mathsf{det}(M)$;

*iii)* $\mathsf{det}(MN) = \mathsf{det}(M)\mathsf{det}(N)$.

*Proof.* Parts (i) and (ii) are clear. To see (iii):

$$L \in \mathsf{det}(MN) \Rightarrow \exists L' \in \Lambda.\ MN \xrightarrow{*}_c L' =_\beta L$$

by definition; by corollary 6.4 there is a $P \in \Lambda_\oplus$ s.t.

$$MN \xrightarrow{*}_\oplus P \xrightarrow{*}_{\beta_c} L';$$

now $MN \xrightarrow{*}_\oplus P$ implies that $P \equiv M'N'$ where $M \xrightarrow{*}_\oplus M'$ and $N \xrightarrow{*}_\oplus N'$. On the other hand we note that no $\beta_c$ contraction can delete an occurrence of a $\oplus$ e.g.:

$$\mathbf{K}L(M \oplus N) \not\xrightarrow{*}_{\beta_c} L$$

since $M \oplus N \notin \Lambda$. It follows that $P \xrightarrow{*}_{\beta_c} L'$ implies $P \in \Lambda$ being $L' \in \Lambda$. We conclude that $L \in \mathsf{det}(M)\mathsf{det}(N)$, that is $\mathsf{det}(MN) \subseteq \mathsf{det}(M)\mathsf{det}(N)$. The inverse inclusion is clear. □

**Lemma 6.8**

$$M \in \Lambda_\oplus, N \in \Lambda \Rightarrow \mathsf{det}(M[N/x]) = \mathsf{det}(M)[\mathsf{det}(N)/x].$$

*Proof.* By induction on $M$ using lemma 6.7. The only nontrivial case is when $M \equiv M_1 M_2$; now

$$\begin{array}{rcll} \mathsf{det}((M_1 M_2)[N/x]) & = & \mathsf{det}(M_1[N/x] M_2[N/x]) & \\ & = & \mathsf{det}(M_1[N/x])\mathsf{det}(M_2[N/x]) & \text{by lemma 6.7 (iii)} \\ & = & \mathsf{det}(M_1)[\mathsf{det}(N)/x]\mathsf{det}(M_2)[\mathsf{det}(N)/x] & \text{by ind. hyp.} \\ & = & \mathsf{det}(M_1 M_2)[\mathsf{det}(N)/x] & \text{since } N \in \Lambda \end{array}$$

where in the last step above we observe that, since $N \in \Lambda$, $\mathsf{det}(N)$ is a set of $\beta$-convertible terms; now in the classical calculus we know that

$$Q_1 =_\beta Q_2 \Rightarrow P[Q_1/x] =_\beta P[Q_2/x]$$

from which it follows that

$$\begin{array}{rcl} (P_1[Q_1/x])(P_2[Q_2/x]) & =_\beta & (P_1 P_2)[Q_1/x] \\ & =_\beta & (P_1 P_2)[Q_2/x]. \end{array}$$

□

**Theorem 6.9** *For any* $M, N \in \Lambda_\oplus$

$$\lambda_c \vdash M = N \Rightarrow M =_c N.$$

*Proof.* Using lemma 6.7 and 6.8, we check that axioms of $\lambda_c$ are satisfied by the relation $=_c$; indeed the only interesting case is that of $(\beta_c)$: let $M \in \Lambda_\oplus$ and $N \in \Lambda$, then using lemma 6.7 we have

$$\begin{array}{rcl} \mathsf{det}((\lambda x.M)N) & = & \mathsf{det}(\lambda x.M)\mathsf{det}(N) \\ & = & (\lambda x.\mathsf{det}(M))\mathsf{det}(N) \\ & = & \mathcal{A} \end{array}$$

say, then

$$\begin{array}{rcl} P \in \mathcal{A} & \Leftrightarrow & \exists Q \in \mathsf{det}(M).\ P =_\beta (\lambda x.Q)N =_\beta Q[N/x] \\ & \Leftrightarrow & P \in \mathsf{det}(M)[\mathsf{det}(N)/x] = \mathsf{det}(M[N/x]), \end{array}$$

by lemma 6.8. □

We conjecture that $M =_c N \Rightarrow \lambda_c \vdash M = N$.

**Corollary 6.10** *The theory* $\lambda_c$ *is consistent.*

*Proof.* For any $M, N \in \Lambda$

$$M =_c N \ \Leftrightarrow\ \mathsf{det}(M) = \mathsf{det}(N) \ \Leftrightarrow\ M =_\beta N,$$

that is $=_c$ restricted to $\Lambda$ coincides with $=_\beta$. This implies that the theory induced by $=_c$ is a conservative extension of $\lambda$: hence it is consistent. Then, by the theorem, $\lambda_c$ is consistent. □

In force of corollary 6.4 it follows that $\mathsf{det}(M)$ is essentially the "immediate deterministic structure" of the discussion at the beginning of section 3. Therefore the meaning of theorem 6.9 is that the choice structure gets definitely lost in this calculus.

## 6.2 Run-time Choice $\lambda$-calculus

In previous sections we studied the properties of the relation $\longrightarrow_r$ and gave both operational and denotational characterizations of the equivalence it induces over terms. In this section our aim is to present an axiomatization of this equivalence (even if not a complete one), allowing to compare this relation with the $\beta$-convertibility relation of the classical $\lambda$-calculus. Here too, as in the case of the theory $\lambda_c$, we get inspiration from [48] and [16].

**Definition 6.11** *The theory $\lambda_r$ is the equational theory over $\Lambda_\oplus$ which results from $\lambda_c$ substituting ($\beta_c$) with the unrestricted axiom*

$$(\beta_r) \qquad (\lambda x.M)N = M[N/x]$$

*and eliminating axioms ($\theta$) and ($\gamma$).*

This theory was proved consistent in [48] with syntactical methods: for us it is actually a corollary of previous results.

**Theorem 6.12** *The syntactical model $D_*$ is a non trivial model of $\lambda_r$, hence $\lambda_r$ is consistent.*

*Proof.* By Def. 4.13, lemma 4.14 and proposition 4.16 we know that $D_*$ is a syntactical model (actually an extensional one); now the proof that the equations and rules up to ($\xi$) are valid runs as in the classical way; the rest is an immediate consequence of the definition of the interpretation of $\oplus$ and of the semilinearity of $D_*$. □

In his proof Sharma didn't prove consistency of the full theory, which in his formulation had among its axioms also

$$(\gamma) \qquad \lambda x.M \oplus N = \lambda x.M \oplus \lambda x.N.$$

We do not include this axiom in the theory $\lambda_r$ because of its special status, illustrated in the following proposition. Note however that ($\gamma$) holds in $D_*$, hence the previous theorem readily extends to the theory $\lambda_r + \gamma$.

Another equation which appeared in the literature (see [5]) is

$$(\iota) \qquad M \oplus N = \lambda x.Mx \oplus Nx \quad \text{if } x \notin \mathrm{FV}(M \oplus N);$$

it is clearly connected (actually equivalent) with the classical axiom $\eta$.

**Proposition 6.13**

*i)* $\lambda_r + \eta \vdash \gamma$;

*ii)* $\lambda_r + \iota \vdash \eta$.

*Proof.*

(i) Let $y \notin \mathrm{FV}(M) \cup \mathrm{FV}(N)$:

$$\begin{array}{rcll} (\lambda x.M \oplus N)y & = & M[y/x] \oplus N[y/x] & \text{by } (\beta_r) \\ & = & (\lambda x.M)y \oplus (\lambda x.N)y & \text{by } (\beta_r) \\ & = & (\lambda x.M \oplus \lambda x.N)y & \text{by } (\delta) \end{array}$$

from which $\lambda x.M \oplus N = \lambda x.M \oplus \lambda x.N$ follows by $(\eta)$.

(ii) Let $x \notin \mathrm{FV}(M)$:

$$\begin{array}{rcll} M & = & M \oplus M & \text{by } (\zeta_1) \\ & = & \lambda x.Mx \oplus Mx & \text{by } (\iota) \\ & = & \lambda x.Mx & \text{by } (\zeta_1). \end{array}$$

□

**Remark 6.14** the independence of the axiom $(\gamma)$ from the theory $\lambda_r$ has been proved in [18]: this has been shown building a model $N_* = N_*^\sharp \to N_*^\sharp$ of the theory $\lambda_r$ in the category of **CPO**, using Moggi's strong monads (see [38]); this model, which is of course non-extensional, doesn't satisfy $(\gamma)$.

We leave as an open question whether the theory $\lambda_r + \gamma$ is extensional, but we conjecture it is not.

## 6.3 Conservativity of the theory $\mathcal{T}_{must}$

In this section we finally compare the theory induced by the equivalence studied throughout this paper with the theories $\lambda + \eta$ and $\mathcal{H}^*$. The main theorem is a conservativity result; it is readily seen that this can be considered as an alternative (syntactical) proof of the consistency for the theory $\mathcal{T}_{must}$ defined below.

**Definition 6.15** $\mathcal{T}_{must} = \{M = N \mid M, N \in \Lambda^0_\oplus,\ M \simeq_{must} N\}$.

We know from the semantic construction in the previous sections that:

**Proposition 6.16** *The theory $\mathcal{T}_{must}$ is the theory of the model $D_*$, hence it is consistent.*

*Proof.* Immediate consequence of the full abstraction theorem. □

This fact is not illuminating, however, with respect to the question of conservativity. To establish this result we are going to prove a "simulation lemma" which says that contexts containing the $\oplus$ operator do not discriminate more w.r.t. must-convergency than classical contexts do.

**Definition 6.17** *Let $M \in \Lambda_\oplus$, $\mathcal{F} \subseteq M$ a subset of the set of redexes occurring in $M$, and $\sigma$ a head reduction starting with $M$; then*

*i) $\sigma$ is* finitely often *in $\mathcal{F}$ iff*

$$\exists m \, \forall n \geq m.\ \sigma_{0,n+1} : M \xrightarrow{*}_r M_n \xrightarrow{\Delta}_r M_{n+1} \Rightarrow \Delta \notin \mathcal{F}/\sigma_{0,n};$$

*ii) let $\sigma$ be any head reduction of $M$ finitely often in $\mathcal{F}$ then*

$$\deg(\mathcal{F}, \sigma) = \max\{n \mid \exists m \in \omega \, \exists R \in \mathcal{F}/\sigma_{0,m}.\ M \xrightarrow{m}_h \lambda\vec{x}.RM_1 \ldots M_n\};$$

*iii) let $\sigma_1, \ldots, \sigma_n$ be head reductions of $M$ finitely often in $\mathcal{F}$ then*

$$\deg(\mathcal{F}, \sigma_1, \ldots, \sigma_n) = \max\{\deg(\mathcal{F}, \sigma_i) \mid 1 \leq i \leq n\}.$$

**Remark 6.18** $\sigma$ is *finitely often* in $\mathcal{F}$ iff it contracts at most a finite number of (residuals of) redexes in $\mathcal{F}$.

The next lemma is based on the idea that, if a context containing the operator $\oplus$ converges on a term $M$, while it diverges on a term $N$, the choices caused by the $\oplus$'s inside the context which are essential for this convergency-divergency property, are bounded above by those which are necessary to converge on $M$. On the other hand we know that all the reductions on $M$ will converge, while there is a diverging reduction on $N$. The point is to simulate the choices of this last reduction, encoding them in a classical context. Because of its complexity we defer the proof of the next lemma to section 7.

**Lemma 6.19 (Simulation Lemma)** *Given $M, N \in \Lambda_\oplus$*

$$\exists D[\,] \in \Lambda_\oplus[\,].\ D[M]\!\downarrow \wedge D[N]\!\uparrow \Rightarrow\ \exists C[\,] \in \Lambda[\,].\ C[M]\!\downarrow \wedge C[N]\!\uparrow.$$

We remind the reader that SOL is the set of classical terms reducing to a head normal form, and that

$$\mathcal{H}^* = \{M = N \mid M, N \in \Lambda^0, \forall C[\,] \in \Lambda[\,].\ C[M] \in \text{SOL} \ \Leftrightarrow\ C[N] \in \text{SOL}\}.$$

It is known that $\mathcal{H}^*$ is the theory of the models $D_\infty$. We conclude the present section with the conservativity theorem for $\mathcal{T}_{must}$.

**Theorem 6.20**

*i)* $\lambda_r + \gamma \subseteq \mathcal{T}_{must}$,

*ii)* $\mathcal{T}_{must}$ *is a conservative extension of* $\mathcal{H}^*$,

*iii)* $\mathcal{T}_{must}$ *is a conservative extension of* $\lambda + \eta$.

*Proof.* To prove the first part, simply note that $\mathcal{T}_{must}$ is the theory of a model of $\lambda_r$, and that this model validates $\gamma$ since it is extensional by lemma 4.14 (alternatively one can use the algebraic semantics to prove that the axioms of $\lambda_r$ are included in $\mathcal{T}_{must}$). As to (ii): let $M,\ N \in \Lambda^0$ be such that $\mathcal{T}_{must} \not\vdash M = N$; then there is a context $D[\,] \in \Lambda_\oplus[\,]$ such that, say, $D[M]\!\downarrow$ and $D[N]\!\uparrow$. By lemma 6.19, there is a context $C[\,] \in \Lambda[\,]$ such that $C[M]\!\downarrow$, that is $C[M] \in \mathrm{SOL}$ and $C[N] \notin \mathrm{SOL}$; hence $\mathcal{H}^* \not\vdash M = N$: it follows that $\mathcal{H}^* \subseteq \mathcal{T}_{must}$. On the other hand, and *a fortiori*, if $M,\ N \in \Lambda^0$, then

$$\begin{array}{rcl} \mathcal{T}_{must} \vdash M = N & \Rightarrow & \forall D[\,] \in \Lambda_\oplus[\,].\ D[M]\!\downarrow\ \Leftrightarrow\ D[N]\!\downarrow \\ & \Rightarrow & \forall C[\,] \in \Lambda[\,].\ C[M] \in \mathrm{SOL}\ \Leftrightarrow\ C[N] \in \mathrm{SOL} \\ & \Rightarrow & \mathcal{H}^* \vdash M = N, \end{array}$$

since the restriction of the predicate $\downarrow$ to $\Lambda$ coincides with the set SOL.
Finally (iii) follows from (ii) and the fact that $\mathcal{H}^*$ is an extensional $\lambda$-theory.

□

# 7 Detailed proofs

In the present section we present in detail those proofs which, for the sake of readability, have been omitted from the text.

## 7.1 The Standardization Theorem

**Definition 7.1** *Let* $\sigma : M \xrightarrow{*} N$ *be a finite reduction, where* $\sigma = \Delta_1 + \cdots + \Delta_n$*; then*

*i)* $\Delta \in M$ is contracted *in* $\sigma$ *iff for some* $k \leq n$, $\Delta_k \in \Delta/\sigma_{0,k-1}$*;*

*ii)* $lmc(\sigma)$ *is the leftmost redex* $\Delta \in M$ *which is contracted in* $\sigma$*;*

*iii)* $lmcpos(\sigma)$ *is the (actually unique) index* $k$ *such that* $\Delta_k \in lmc(\sigma)/\sigma_{0,k-1}$.

**Lemma 7.2** *Let* $\sigma : M \xrightarrow{*} N$ *be a finite reduction. Then*

$$lmc(\sigma) + \sigma/lmc(\sigma) : M \xrightarrow{*} N$$

*is a finite reduction.*

*Proof.* By induction on $lmcpos(\sigma)$. Let $\sigma = \Delta + \sigma'$: the basic case $lmc(\sigma) = \Delta$ is trivial; the inductive step is illustrated in the following diagram:

$$\begin{array}{ccccc} M \equiv M_0 & \xrightarrow{\Delta} & M_1 & \xrightarrow{\sigma'} & M_n \equiv N \\ \downarrow{\scriptstyle lmc(\sigma)} & & \downarrow{\scriptstyle lmc(\sigma')} & & \\ M'_0 & \overset{*}{\dashrightarrow} & M'_1 & & \end{array}$$

We distinguish 2 cases:

Case 1: $lmc(\sigma) = (\lambda x.U)V$.

subcase a: $\Delta \notin U, \Delta \notin V$: then $M'_0 \xrightarrow{\Delta} M'_1$.

subcase b: $\Delta \in U$: if $x \notin \Delta$, then $M'_0 \xrightarrow{\Delta} M'_1$, else $M'_0 \xrightarrow{\Delta'} M'_1$, where $\Delta' = \Delta[V/x]$.

subcase c: $\Delta \in V$: if $x$ occurs $k$ ($\geq 0$) times in $U$, then $M'_0 \xrightarrow{\rho} M'_1$, where $\rho = \Delta^1 + \ldots + \Delta^k$ is the reduction of $k$ copies of $\Delta$.

Case 2: $lmc(\sigma) = U \oplus V$. Let $U \oplus V \to U$; if $\Delta \notin V$, then $M'_0 \xrightarrow{\Delta} M'_1$, otherwise $M'_0 \equiv M'_1$. Similarly for the symmetrical case.

Now, the thesis follows observing that $lmcpos(\sigma') = lmcpos(\sigma) - 1$. □

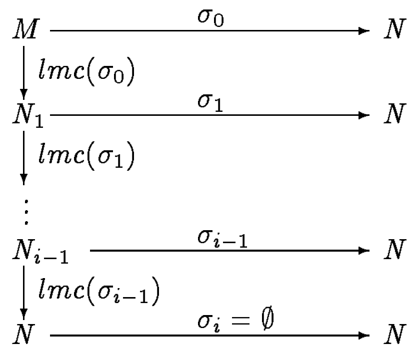
**Definition 7.3** *([9], 12.3.3) Let $\sigma : M \xrightarrow{*} N$ be a given reduction and let*

$$\begin{array}{rcl} \sigma_0 & = & \sigma; \\ \Delta_0 & = & lmc(\sigma_0); \\ \sigma_{k+1} & = & \sigma_k / \Delta_k; \\ \Delta_{k+1} & = & lmc(\sigma_{k+1}). \end{array}$$

*If there exists $i \geq 0$ such that $\sigma_i = \emptyset$, then the* standardization *of $\sigma$, notation $\sigma_S$, is defined to be*

$$\sigma_S = \Delta_0 + \ldots + \Delta_{i-1}.$$

*The construction of $\sigma_S$ from $\sigma$ uses the property stated by lemma 7.2, and is illustrated in the following figure.*




**Remark 7.4** Given the reduction $\sigma$, let $\sigma_S = \Delta_1 + \ldots + \Delta_i$, for some $i$. Note that $\forall j,k.j < k \Rightarrow \Delta_{j+1} = lmc((\sigma_S)_{j,k})$. It comes out that, for each reduction $\sigma$, its standardization $\sigma_S$, whenever it exists, is a standard reduction.

In order to complete the proof of the standardization theorem, we need to show that, if $\sigma$ is a finite reduction, then $\sigma_S$ exists. To this aim, following [51, 28] (see also [9], Ch.14), we introduce a labelled calculus:

**Definition 7.5** *The set of non-deterministic labelled $\lambda$-terms, $\Lambda^N_\oplus$, is defined as follows:*

*i)* $x \in \Lambda^N_\oplus$ *for all* $x \in Var$*;*

*ii)* $M, N \in \Lambda^N_\oplus \Rightarrow (MN) \in \Lambda^N_\oplus$*;*

*iii)* $M \in \Lambda^N_\oplus, x \in Var \Rightarrow (\lambda x.M) \in \Lambda^N_\oplus$*;*

*iv)* $M, N \in \Lambda^N_\oplus \Rightarrow M \oplus N \in \Lambda^N_\oplus$*.*

*v)* $M \in \Lambda^N_\oplus \Rightarrow (M^n) \in \Lambda^N_\oplus$ *for every natural number* $n$*.*

**Definition 7.6** *As in [9], the following notions of reduction are introduced on* $\Lambda^N_\oplus$*:*

*i)* $(\beta_+)$*:* $(\lambda x.M)^{n+1}N \to (M[N^n/x])^n$*;*

*ii) label:* $(M^n)^m \to M^{min(n,m)}$*;*

*iii)* $\oplus$*:* $M \oplus N \to M \quad , \quad M \oplus N \to N$*;*

*iv)* $lab.\beta.\oplus$*:* $\beta_+ \cup \oplus \cup$ *label.*

**Lemma 7.7** *Let* $M \in \Lambda^N_\oplus$*. Then every* $lab.\beta.\oplus$*-reduction starting with* $M$ *terminates.*

*Proof.* It is a trivial extension of the proof that $\beta_+ \cup$ *label* has the strong normalization property (see [9],§14.1). □

**Lemma 7.8** *Define* $|\cdot|: \Lambda_\oplus^N \to \Lambda_\oplus$ *to be the map that erases all labels from a term in* $\Lambda_\oplus^N$. *Let* $M, N \in \Lambda_\oplus$ *and* $\sigma : M \xrightarrow{*} N$. *There exist* $M', N' \in \Lambda_\oplus^N$ *such that* $|M'| = M, |N'| = N$ *and* $M' \xrightarrow{*}_{lab.\beta.\oplus} N'$.

*Proof.* Simply give all subterms of $M$ having the shape $(\lambda x.N)$ the right labels (see [9], 14.2.1), e.g. $(\lambda x.N)^{|\sigma|}$. □

**Theorem 7.9** (See Theorem 2.9) *For all* $M, N \in \Lambda_\oplus$. $M \xrightarrow{*} N \Rightarrow M \longrightarrow_s N$.

*Proof.* As in [30], by lemma 7.8, take a labelling for the given reduction $\sigma : M \xrightarrow{*} N$; observe that any residual of an indexed redex $\Delta$ has the same index as $\Delta$; it follows that the indexing of $\sigma$ determines an indexing on $\sigma_S$, which, by lemma 7.7, terminates. □

## 7.2 Semi-separability

### 7.2.1 Head contexts

**Definition 7.10**

*i) A context is a* head context *iff it is of the form*

$$C[\,] \equiv (\lambda x_1 \dots x_n.[\,])X_1 \dots X_n U_1 \dots U_m;$$

*ii) abbreviate* $x_1 \dots x_n$ *with* $\vec{x}$, $X_1 \dots X_n$ *with* $\vec{X}$ *and* $U_1 \dots U_m$ *with* $\vec{U}$; *similarly consider a context* $D[\,] \equiv (\lambda y_1 \dots y_h.[\,])Y_1 \dots Y_h V_1 \dots V_k$, *abbreviated* $(\lambda\vec{y}.[\,])\vec{Y}\vec{V}$, *then define*

$$D \bullet C[\,] \equiv (\lambda \vec{x}\vec{z}.[\,])\vec{X}^\circ \vec{Z}\vec{U}^\circ \vec{V}$$

*where* $\circ = [\vec{Y}/\vec{y}]$, *and* $\vec{z} \equiv z_1 \dots z_r \equiv y_{i_1} \dots y_{i_r}$, *if* $\{y_{i_1} \dots y_{i_r}\} = \{y_1, \dots, y_h\} - \{x_1, \dots, x_n\}$ *and* $\vec{Z} \equiv Y_{i_1} \dots Y_{i_r}$.

**Lemma 7.11** *Let* $C[\,], D[\,]$ *be head contexts; then, for any* $M \in \Lambda_\oplus$ *and* $k \geq 1$*:*
*if* $\omega^k(M) = \{M_1, \dots, M_l\}$ *then:*

*i)* $\omega^k(C[M]) = \begin{cases} \omega^k(C[M_1]) \cup \dots \cup \omega^k(C[M_l]) & \textit{if } C[M]\downarrow \\ \{\Omega\} & \textit{otherwise} \end{cases}$

*ii)* $\omega^k(D \bullet C[M]) = \omega^k(D[C[M]])$.

*Proof.* Let $C[\,] \equiv (\lambda x_1 \dots x_n.[\,])X_1 \dots X_n U_1 \dots U_m$, then

$$\begin{array}{rcl} C[M] & \xrightarrow{*}_h & M^* U_1 \dots U_m \\ & \xrightarrow{*}_h & M_i^* U_1 \dots U_m \\ & {}_h\!\xleftarrow{*} & C[M_i]; \end{array}$$

for $i = 1, \dots, l$, where $* = [X_1/x_1, \dots, X_n/x_n]$. It follows that $L$ is a principal hnf of $C[M]$ iff for some $i \leq l$, $L$ is a principal hnf of $C[M_i]$, establishing (i).

To prove (ii) let $D[\,] \equiv (\lambda \vec{y}.[\,])\vec{Y}\vec{V}$, and $C[\,]$ as above; then, reasoning as for (i), we have

$$\begin{array}{rcl} D[C[M]] & \equiv & (\lambda \vec{y}.(\lambda \vec{x}.M)\vec{X}\vec{U})\vec{Y}\vec{V} \\ & \xrightarrow{*}_h & (\lambda \vec{x}.M)^\circ \vec{X}^\circ \vec{U}^\circ \vec{V} \\ & \equiv & (\lambda \vec{x}.M^\diamond)\vec{X}^\circ \vec{U}^\circ \vec{V} \\ & \xrightarrow{*}_h & M^{\diamond\star}\vec{U}^\circ \vec{V} \\ & {}_h\!\xleftarrow{*} & (\lambda \vec{x}\vec{z}.M)\vec{X}^\circ \vec{Z}\vec{U}^\circ \vec{V} \\ & \equiv & D \bullet C[M], \end{array}$$

where $\circ = [\vec{Y}/\vec{y}]$, $\diamond = [\vec{Z}/\vec{z}]$, $\star = [\vec{X}^\circ/\vec{x}]$, and $\vec{Z}$, $\vec{z}$ are as in the definition 7.10. □

### 7.2.2 Test for equality

**Lemma 7.12** (See Lemma 3.19) *There exists a combinator* $\mathbf{H} \in \Lambda$ *of the shape*

$$\mathbf{H} \equiv \lambda xy.xH_1 \dots H_l,$$

*with* $x \notin FV(H_1) \cup \dots \cup FV(H_l)$, *such that, for all non-negative integers* $n, m$*:*

$$\mathbf{Hnm} =_{\beta\eta} \begin{cases} \mathbf{1} & \text{if } n = m \\ \mathbf{0} & \text{otherwise} \end{cases}$$

*Proof.* To build $\mathbf{H}$ we solve the following system of equations in the theory $\lambda\beta\eta$:

$$\begin{cases} \mathbf{H00} & = & \mathbf{1} \\ \mathbf{H0}(\mathbf{Succ}\, y) & = & \mathbf{0} \\ \mathbf{H}(\mathbf{Succ}\, x)\mathbf{0} & = & \mathbf{0} \\ \mathbf{H}(\mathbf{Succ}\, x)(\mathbf{Succ}\, y) & = & \mathbf{H}xy \end{cases}$$

Let $\mathbf{H} \equiv \lambda uv.uPQ(vRT)$, where the combinators $P, Q, R, T$ will be specified later. We compute

$$\begin{array}{rcl} \mathbf{H00} & = & \mathbf{0}PQ(\mathbf{0}RT) \\ & = & QT; \\ \mathbf{H0}(\mathbf{Succ}\, y) & = & \mathbf{0}PQ(\mathbf{Succ}\, yRT) \\ & = & Q(R(yRT)); \\ \mathbf{H}(\mathbf{Succ}\, x)\mathbf{0} & = & \mathbf{Succ}\, xPQ(\mathbf{0}RT) \\ & = & P(xPQ)T; \\ \mathbf{H}(\mathbf{Succ}\, x)(\mathbf{Succ}\, y) & = & \mathbf{Succ}\, xPQ(\mathbf{Succ}\, yRT) \\ & = & P(xPQ)(R(yRT)). \end{array}$$

Now we choose

$$\begin{array}{rcl} P & \equiv & \lambda ab.b\mathbf{O}a \\ Q & \equiv & \lambda a.a\mathbf{K} \\ R & \equiv & \lambda ab.b(\mathbf{K0})\mathbf{C}_*a \\ T & \equiv & \lambda a.a\mathbf{1}(\mathbf{K0}) \end{array}$$

where $\mathbf{C}_* \equiv \lambda ab.ba$; it is straightforward to see that these choices give the desired result: in particular for the fourth equation we have

$$\begin{array}{rcl} P(xPQ)(R(yRT)) & = & R(yRT)\mathbf{O}(xPQ) \\ & = & \mathbf{O}(\mathbf{K0})\mathbf{C}_*(yRT)(xPQ) \\ & = & \mathbf{C}_*(yRT)(xPQ) \\ & = & xPQ(yRT) \\ & = & \mathbf{H}xy. \end{array}$$

□

**Corollary 7.13** (See Corollary 3.20) *If* $N \equiv \mathbf{n}_1 \oplus \ldots \oplus \mathbf{n}_r$, *with* $r \geq 1$, *then, for all* $m$,

$$\omega^1(\mathbf{H}N\mathbf{m}) = \{\mathbf{0} \mid \exists i \leq r.\, n_i = m\} \cup \{\mathbf{1} \mid \exists j \leq r.\, n_j \neq m\}.$$

*Proof.* From the shape of $\mathbf{H}$ one sees that, when applied to closed terms, it behaves like the head context $\lambda y.[\,]H_1 \ldots H_l$; on the other hand the $=_{\beta\eta}$ in the lemma is actually $\longrightarrow_{\beta\eta}$, because the numerals $\mathbf{0}$ and $\mathbf{1}$ are normal forms. Since $\longrightarrow_{\beta\eta} \subseteq \longrightarrow$, and using the standardization theorem we have that each exhaustive head reduction of $\mathbf{H}N\mathbf{m}$ has to start with

$$\begin{array}{rcl} \mathbf{H}N\mathbf{m} & \longrightarrow & NH_1[\mathbf{m}/y] \ldots H_l[\mathbf{m}/y] \\ & \longrightarrow & \mathbf{n}_i H_1[\mathbf{m}/y] \ldots H_l[\mathbf{m}/y]; \end{array}$$

for some $1 \leq i \leq r$; now the corollary follows from lemma 7.12 and lemma 7.11. □

### 7.2.3 Semi-separability lemmas

**Lemma 7.14** (See Lemma 3.21) *For* $M, N \in \Lambda_\oplus$,

$$M \not\lesssim_2 N \;\Rightarrow\; \exists C[\,].\, C[M]\!\downarrow \;\wedge\; C[N]\!\uparrow.$$

*Proof.* By cases.

Case 1: when $M \not\lesssim_1 N$: then either $\omega^1(N) = \mathcal{N} = \{\Omega\}$ and $\omega^1(M) = \mathcal{M} \neq \{\Omega\}$, so that there is nothing to prove, or there exists $N' \in \mathcal{N}$ such that, for all $M' \in \mathcal{M}$, $M' \not\sim N'$. If $(\mathcal{M} \cup \mathcal{N})_{/\sim} = \{[L_1], \ldots, [L_h]\}$, then there is an $i$ such that $N' \in [L_i]$ and $[L_i] \cap \mathcal{M} = \emptyset$; by lemma 3.18 (ii) we know that there is a context $C[\,]$ such that $\omega^1(C[N']) = \{z_i\}$ for some variable $z_i$, and, for all $M' \in \mathcal{M}$ there is a $j \neq i$ s.t. $\omega^1(C[M']) = \{z_j\}$, where $z_i \not\equiv z_j$; now let

$$C'[\,] \equiv (\lambda z_1 \ldots z_h.[\,])z_1 \ldots z_{i-1}(\omega\omega)z_{i+1} \ldots z_h;$$

it is a head context, hence by lemma 7.11 and by the fact that head contexts are closed under composition we conclude that $\omega^1(C' \bullet C[M])$ is a subset of $\{z_1, \ldots, z_h\} - \{z_i\}$, and that $\omega^1(C' \bullet C[N]) = \omega^1(C' \bullet C[N']) = \{\Omega\}$.

Case 2: when $M \leq_1 N$ but $M \not\leq_2 N$: then by definition we have

$$M \leq_1 N \wedge M \not\leq_2 N \;\Rightarrow\; \exists \langle \mathsf{U}, \mathsf{V} \rangle \in \mathsf{Pair}_1(\mathcal{M}, \mathcal{N}) = \mathsf{Pair}(\mathcal{M}, \mathcal{N}).\ \mathsf{U} \not\sqsubseteq^\sharp \mathsf{V};$$

this means that, for some $[P] \in (\mathcal{M} \cup \mathcal{N})_{/\sim}$, letting $\mathcal{M}_{[P]} = [P] \cap \mathcal{M}$ and similarly $\mathcal{N}_{[P]} = [P] \cap \mathcal{N}$, we have

$$\langle \mathsf{U}, \mathsf{V} \rangle \in \mathsf{Pair}(\mathcal{M}_{[P]}, \mathcal{N}_{[P]}).$$

By lemma 3.18 (i) there exists a context $C[\,]$ and an integer $r$ such that

$$\begin{aligned} \omega^2(C[M]) &= \omega^2(C[\mathcal{M} - \mathcal{M}_{[P]}]) \cup \omega^2(C[\mathcal{M}_{[P]}]) \\ &= \{y\} \cup \{x\mathcal{M}_1^1 \ldots \mathcal{M}_r^1, \ldots, x\mathcal{M}_1^n \ldots \mathcal{M}_r^n\} \end{aligned}$$

and

$$\begin{aligned} \omega^2(C[N]) &= \omega^2(C[\mathcal{N} - \mathcal{N}_{[P]}]) \cup \omega^2(C[\mathcal{N}_{[P]}]) \\ &= \{y\} \cup \{x\mathcal{N}_1^1 \ldots \mathcal{N}_r^1, \ldots, x\mathcal{N}_1^m \ldots \mathcal{N}_r^m\} \end{aligned},$$

so that, for some $1 \leq i \leq r$, it must be the case that

$$\mathsf{U} = \{\mathcal{M}_i^1, \ldots, \mathcal{M}_i^n\} \quad \text{and} \quad \mathsf{V} = \{\mathcal{N}_i^1, \ldots, \mathcal{N}_i^m\}.$$

In the sequel we assume that

$$\forall j \leq n, k \leq m.\ x \notin FV(\mathcal{M}_i^j) \wedge x \notin FV(\mathcal{N}_i^k):$$

there is no theoretical loss, since as in the classical $\lambda$-calculus, one can use the technique of Böhm transformations to make these occurrences harmless (see [9], §10.3).

Since $\mathsf{U} \not\sqsubseteq^\sharp \mathsf{V}$, there exists $1 \leq k \leq m$ such that, for all $1 \leq j \leq n$, we have $\mathcal{M}_i^j \not\leq_1 \mathcal{N}_i^k$: then $\mathcal{M}_i^j \neq \{\Omega\}$ for all $j$. Now we have two subcases.

Subcase 2.1: $\mathcal{N}_i^k = \{\Omega\}$: it follows that, taking $C'[\,] \equiv (\lambda x.[\,])(\lambda a_1 \ldots a_r.a_i)$ we have

$$\omega^2(C'[\{x\mathcal{M}_1^1 \ldots \mathcal{M}_r^1, \ldots, x\mathcal{M}_1^n \ldots \mathcal{M}_r^n\}]) = \{\mathcal{M}_i^1, \ldots, \mathcal{M}_i^n\},$$

while

$$\omega^2(C'[\{x\mathcal{N}_1^1 \ldots \mathcal{N}_r^1, \ldots, x\mathcal{N}_1^m \ldots \mathcal{N}_r^m\}]) = \{\Omega\};$$

it follows that $C' \bullet C[M]{\downarrow}$ and $C' \bullet C[N]{\uparrow}$.

Subcase 2.2: $\mathcal{N}_i^k \neq \{\Omega\}$: then for each $1 \leq j \leq n$ there is a $[Q_j] \in (\bigcup \mathsf{U} \cup \mathcal{N}_i^k)_{/\sim}$ s.t.

$$[Q_j] \subseteq \mathcal{N}_i^k - \mathcal{M}_i^j. \tag{1}$$

Using lemma 3.18 (ii) we know that there exists a head context $\tilde{C}[\,]$ transforming $\bigcup \mathsf{U} \cup \mathcal{N}$ into a set of variables having the same cardinality as $(\bigcup \mathcal{U} \cup \mathcal{N}_i^k)_{/\sim}$: say $\{z_1, \ldots, z_h\}$. (1) now implies that

$$\forall j\, \exists l \leq h. z_l \in \omega^1(\tilde{C}[\mathcal{N}_i^k]) - \omega^1(\tilde{C}[\mathcal{M}_i^j]). \tag{2}$$

We define $\bar{C}[\,]$ as the composition $((\lambda z_1 \ldots z_h.[\,])\mathbf{1} \ldots \mathbf{h}) \bullet \tilde{C}[\,]$.
Say that $|(\mathcal{N}_i^k)_{/\sim}| = l$: then we take

$$C'[\,] \equiv (\lambda x.[\,])(\lambda y_1 \ldots y_l.v\, \underbrace{\bar{C}[y_i] \ldots \bar{C}[y_i]}_{l}).$$

Set $\mathcal{X}^j = \omega^h(\bar{C}[\mathcal{M}_i^j])$ and $\mathcal{Y} = \omega^h(\bar{C}[\mathcal{N}_i^k])$; they are sets of Church numerals, and, by (2), no $\mathcal{X}^j$ contains all the numerals in $\mathcal{Y}$. Now

$$\begin{array}{rcl} \omega^h(C' \bullet C[M]) & = & \{y, v\, \underbrace{\mathcal{X}^1 \ldots \mathcal{X}^1}_{l}, \ldots, v\, \underbrace{\mathcal{X}^n \ldots \mathcal{X}^n}_{l}\}, \\ \omega^h(C' \bullet C[N]) & = & \{y, v\, \underbrace{\mathcal{Y} \ldots \mathcal{Y}}_{l}\}. \end{array}$$

Using the combinator $\mathbf{H}$ of lemma 3.19, we finally define

$$C''[\,] \equiv (\lambda v.[\,])(\lambda v_1 \ldots v_l.\mathbf{P}_l(\mathbf{H}v_1\mathbf{1}) \ldots (\mathbf{H}v_l\mathbf{l}))(\mathbf{K}(\omega\omega))\mathbf{I},$$

where $\mathbf{P}_l$ $\lambda$-defines the numeric function $\prod_{i=1}^{l} n_l$. It follows that, by corollary 3.20,

$$\omega^1(C'' \bullet C' \bullet C[M]) = \{\mathbf{0}\}(\mathbf{K}(\omega\omega))\mathbf{I} = \{\mathbf{I}\},$$

while

$$\begin{array}{rcl} \omega^1(C'' \bullet C' \bullet C[N]) & = & \left\{ \begin{array}{l} \{\mathbf{0}, \mathbf{1}\}(\mathbf{K}(\omega\omega))\mathbf{I} \\ \{\mathbf{1}\}(\mathbf{K}(\omega\omega))\mathbf{I} \end{array} \right. \\ & = & \{\Omega\} \end{array}$$

according to the numerals in $\mathcal{Y}$.

□

**Lemma 7.15** (See Lemma 3.23) *For $M, N \in \Lambda_\oplus$ and $k \geq 2$,*

$$M \not\leq_k N \;\Rightarrow\; \exists C[\,].\, C[M] \not\leq_2 C[N].$$

*Proof.* By induction on $k$. Suppose $k > 2$: then, letting $\mathcal{M} = \omega^k(M)$ and $\mathcal{N} = \omega^k(N)$, there are two subacses: either $\mathcal{M} \not\leq_{k-1} \mathcal{N}$, in which case the thesis follows directly from the inductive hypothesis, or $\mathcal{M} \leq_{k-1} \mathcal{N}$ and $\mathcal{M} \not\leq_k \mathcal{N}$. In the last case by definition we have that

$$\exists \langle \mathsf{U}, \mathsf{V} \rangle \in \mathsf{Pair}_{k-1}(\mathcal{M}, \mathcal{N}).\ \mathsf{U} \not\sqsubseteq^\sharp \mathsf{V};$$

this implies the existence of $\langle \mathsf{U}', \mathsf{V}' \rangle \in \mathsf{Pair}_1(\mathcal{M}, \mathcal{N})$ s.t. $\langle \mathsf{U}, \mathsf{V} \rangle \in \mathsf{Pair}_{k-2}(\bigcup \mathsf{U}', \bigcup \mathsf{V}')$; it follows that, for some $[P] \in (\mathcal{M} \cup \mathcal{N})_{/\sim}$ the sets $\mathcal{M}_{[P]}$ and $\mathcal{N}_{[P]}$, defined as in the proof of the previous lemma, are non empty and

$$\langle \mathsf{U}, \mathsf{V} \rangle \in \mathsf{Pair}_{k-2}(\mathcal{M}_{[P]}, \mathcal{N}_{[P]}).$$

Using lemma 3.18, we can find a context $C[\,]$ "selecting" $P$, thus we get

$$\begin{array}{rcl} \omega^k(C[M]) & = & \{y, x\mathcal{M}_1^1 \ldots \mathcal{M}_r^1, \ldots, x\mathcal{M}_1^n \ldots \mathcal{M}_r^n\} \\ \omega^k(C[N]) & = & \{y, x\mathcal{N}_1^1 \ldots \mathcal{N}_r^1, \ldots, x\mathcal{N}_1^m \ldots \mathcal{N}_r^m\} \end{array}$$

and, for some $1 \leq i \leq r$,

$$\mathsf{U}' = \{\mathcal{M}_i^1, \ldots, \mathcal{M}_i^n\}, \qquad \mathsf{V}' = \{\mathcal{N}_i^1, \ldots, \mathcal{N}_i^m\}.$$

Again, w.l.o.g. we suppose that $x$ doesn't occur free in any term in $\mathsf{U}'$ or in $\mathsf{V}'$, and we choose $C'[\,] \equiv (\lambda x.[\,])(\lambda a_1 \ldots a_r.a_i)$; so that by lemma 7.11,

$$\begin{array}{rcll} \omega^k(C' \bullet C[M]) & = & \{y\} \cup \mathcal{M}_i^1 \cup \ldots \cup \mathcal{M}_i^n & =_{def} \bar{\mathcal{M}} \\ \omega^k(C' \bullet C[M]) & = & \{y\} \cup \mathcal{N}_i^1 \cup \ldots \cup \mathcal{N}_i^m & =_{def} \bar{\mathcal{N}}; \end{array}$$

hence $\langle \mathsf{U}, \mathsf{V} \rangle \in \mathsf{Pair}_{k-2}(\bar{\mathcal{M}}, \bar{\mathcal{N}})$, from which we conclude that $C' \bullet C[M] \not\leq_{k-1} C' \bullet C[M]$. The inductive hypothesis now applies. □

### 7.3 Full Abstraction Lemmas

In what follows we sometime omit the environment $\rho$ from $[\![M]\!]_\rho$, when it is not necessary.

**Lemma 7.16** *For any $M, N \in \Lambda_\oplus$ and any indexing functions $\mathcal{I}$ and $\mathcal{J}$:*

*i)* $M^{\mathcal{I}} \rhd^* N^{\mathcal{J}} \in \mathrm{N}_\oplus^\Omega \Rightarrow N \leq M$;

*ii)* $M^{\mathcal{I}} \rhd N^{\mathcal{J}} \Rightarrow \forall \rho.\ [\![M^{\mathcal{I}}]\!]_\rho = [\![N^{\mathcal{J}}]\!]_\rho$.

*Proof.* (i) $NBT(M)$ is the same as $NBT(N)$ with the possible exception of some nodes labelled with $\Omega$; since $N \in \mathrm{N}_\oplus^\Omega$ the thesis follows by induction on the height of $N$.
To prove (ii) one checks the clauses in definition 5.4 along the equations of lemma 4.10; e.g.

$$\begin{array}{rcl} [\![(M \oplus N)^{n+1} L]\!] & = & [\![(M \oplus N)^{n+1}]\!] \cdot [\![L]\!] \\ & = & [\![M \oplus N]\!]_{n+1} \cdot [\![L]\!] \\ & = & ([\![M]\!] + [\![N]\!])_{n+1} \cdot [\![L]\!]. \end{array}$$

Now call $x = [\![M]\!]$, $y = [\![N]\!]$, $z = [\![L]\!]$:

$$\begin{array}{rcl}(x+y)_{n+1} \cdot z & = & (x_{n+1} + y_{n+1}) \cdot z \\ & = & x_{n+1} \cdot z + y_{n+1} \cdot z \\ & = & (x \cdot z_n)_n + (y \cdot z_n)_n \\ & = & (x \cdot z_n + y \cdot z_n)_n \\ & = & [\![(ML^n \oplus NL^n)^n]\!].\end{array}$$

□

**Lemma 7.17** *For any $M \in \Lambda_\oplus$, if $L \in \mathrm{N}^\Omega_\oplus$ and $L \le M$, then $[\![L]\!] \sqsubseteq [\![M]\!]$.*

*Proof.* Using the inductive definition of $\mathrm{N}^\Omega_\oplus$.

Case 1: $L \equiv \Omega$, then

$$[\![L]\!] = \bot \sqsubseteq [\![M]\!].$$

In the sequel, since $L \not\equiv \Omega$, $L \le M$ implies that $M\!\downarrow$: let $\{M_1, \ldots, M_r\}$ be the principal hnfs of $M$.

Case 2: $L \equiv x$; now

$$x \le_1 M \Rightarrow x \le_1 M_1 \wedge \ldots \wedge x \le_1 M_r,$$

hence, for $i = 1, \ldots, r$,

$$M_i \equiv \lambda y_1 \ldots y_{n_i}.xM^i_1 \ldots M^i_{n_i}.$$

By lemma 5.3 we know that $[\![x]\!] = \bigsqcup_{\mathcal{I}} [\![x^{\mathcal{I}}]\!]$; hence we proceed by induction on $q = \mathcal{I}(x)$.

Subcase 2.1: $q = 0$, then, by lemma 4.10 (v), for $i = 1, \ldots, r$:

$$\begin{array}{rcl}[\![x^0]\!] & = & [\![x]\!]_0 \\ & = & [\![\lambda y_1 \ldots y_{n_i}.x^0 \underbrace{\Omega \ldots \Omega}_{n_i}]\!] \\ & \sqsubseteq & [\![M_i]\!].\end{array}$$

We conclude $[\![x^0]\!] \sqsubseteq [\![M_1]\!] + \ldots + [\![M_r]\!] = [\![M]\!]$.

Subcase 2.2: $q > 0$, then, by lemma 4.10 (iv), for $i = 1, \ldots, r$:

$$[\![x^q]\!] = [\![\lambda y_1 \ldots y_{n_i}.x^q y_1^{q-1} \ldots y_{n_i}^{q-n_i}]\!].$$

Now each pair in $\mathsf{Pair}_1(\omega^2(L), \omega^2(M_i))$ will have the shape:

$$\langle \{\{y_j^{q-j}\}\}, \{\mathcal{M}_1, \ldots, \mathcal{M}_s\} \rangle$$

and $\{y_j^{q-j}\} \leq \mathcal{M}_h$ for $h = 1, \ldots, s$. By ind. hyp.

$$[\![y_j^{q-j}]\!] \sqsubseteq [\![N_1]\!] + \ldots + [\![N_t]\!] = [\![\mathcal{M}_h]\!]$$

given that $\mathcal{M}_h = \{N_1, \ldots, N_t\}$. This means that again $[\![x^q]\!] \sqsubseteq [\![M_i]\!]$ for each $i$, so that $[\![x^q]\!] \sqsubseteq [\![M]\!]$

Case 3: $L \equiv \lambda x_1 \ldots x_m.xL_1 \ldots L_n$, then the pairs in $\mathsf{Pair}_1(\omega^k(L), \omega^k(M))$, where $k = \mathsf{height}(L)$, are of the form

$$\langle \{\mathcal{L}_j\}, \{\mathcal{M}_j^1, \ldots, \mathcal{M}_j^r\} \rangle,$$

where $\mathcal{L}_j = \omega^{k-1}(L_j)$. By ind. hyp. $[\![\mathcal{L}_j]\!] \sqsubseteq [\![\mathcal{M}_j^h]\!]$ for $h = 1, \ldots, r$; we conclude, as at the end of subcase 2.2, that $[\![L]\!] \sqsubseteq [\![M]\!]$.

Case 4: $L \equiv L_1 \oplus \ldots \oplus L_n$ where we can suppose that the $L_i$ are not sums. Let again $k = \mathsf{height}(L) > 0$. From the definition of $\leq$ we know that

a) $\forall [P] \in (\mathcal{L} \cup \mathcal{M})_{/\sim}. \, [P] \cap \mathcal{L} \neq \emptyset$;

b) $\forall \langle \mathsf{U}, \mathsf{V} \rangle \in \mathsf{Pair}(\mathcal{L}, \mathcal{M}). \, \mathsf{U} \sqsubseteq^\sharp \mathsf{V}$.

where $\mathcal{L} = \omega^k(L)$ and $\mathcal{M} = \omega^k(M)$. For sake of simplicity suppose that

$$\mathcal{L} = \{x\mathcal{L}_1, x\mathcal{L}_2\} \text{ and } \mathcal{M} = \{x\mathcal{M}_1, x\mathcal{M}_2, x\mathcal{M}_3\}; \tag{3}$$

then $\mathsf{Pair}_1(\mathcal{L}, \mathcal{M})$ contains only the pair

$$\langle \{\mathcal{L}_1, \mathcal{L}_2\}, \{\mathcal{M}_1, \mathcal{M}_2, \mathcal{M}_3\} \rangle.$$

We know that

$$\forall j \leq 3 \; \exists i \leq 2. \; \mathcal{L}_i \leq_{k-1} \mathcal{M}_j,$$

so that the inductive hypothesis applies giving that

$$\forall j \leq 3 \; \exists i \leq 2 \, . [\![\mathcal{L}_i]\!] \sqsubseteq [\![\mathcal{M}_j]\!],$$

that is

$$\forall j \leq 3 \; \exists i \leq 2 \, . [\![x\mathcal{L}_i]\!] \sqsubseteq [\![x\mathcal{M}_j]\!].$$

Since in any Smyth algebra $D$, for any $a_1, \ldots, a_n, b_1, \ldots, b_m \in D$

$$\forall j \leq m \; \exists i \leq n. \; a_i \sqsubseteq b_j \Rightarrow a_1 + \ldots + a_n \sqsubseteq^\sharp b_1 + \ldots + b_m,$$

we get the thesis observing that

$$[\![L]\!] = [\![x\mathcal{L}_1]\!] + [\![x\mathcal{L}_2]\!] \sqsubseteq [\![x\mathcal{M}_1]\!] + [\![x\mathcal{M}_2]\!] + [\![x\mathcal{M}_3]\!] = [\![M]\!].$$

The general case is a trivial extension of (3). $\square$

**Lemma 7.18** (See Lemma 5.8) *For any* $M \in \Lambda_\oplus$ *and natural number* $k$*,*

$$\forall \rho \in Env.\ [\![M]\!]_\rho = \bigsqcup_k [\![M^{[k]}]\!]_\rho.$$

*Proof.* For any $k \in \mathbb{N}$, $M^{[k]} \in \mathrm{N}_\oplus^\Omega$ and $M^{[k]} \leq M$, hence by lemma 7.17

$$[\![M^{[k]}]\!] \sqsubseteq [\![M]\!]$$

that is

$$\bigsqcup_k [\![M^{[k]}]\!] \sqsubseteq [\![M]\!].$$

Let $\mathcal{I}$ be any indexing map, then by corollary 5.6 there exist $\mathcal{J}$ and $L \in \mathrm{N}_\oplus^\Omega$ such that $M^\mathcal{I} \rhd^* L^\mathcal{J}$. By lemma 7.16 $[\![M^\mathcal{I}]\!] = [\![L^\mathcal{J}]\!]$ and $L \leq M$; now $[\![L^\mathcal{J}]\!] \sqsubseteq [\![L]\!]$ and, being $L \in \mathrm{N}_\oplus^\Omega$, $L \leq M^{[k]}$, where $k = \mathsf{height}(L)$, by lemma 5.7; again by lemma 7.17 it follows that

$$[\![L]\!] \sqsubseteq [\![M^{[k]}]\!],$$

hence

$$[\![M^\mathcal{I}]\!] = [\![L^\mathcal{J}]\!] \sqsubseteq [\![L]\!] \sqsubseteq [\![M^{[k]}]\!].$$

From this we conclude, by lemma 5.3,

$$[\![M]\!] = \bigsqcup_\mathcal{I} [\![M^\mathcal{I}]\!] = \bigsqcup_k [\![M^{[k]}]\!].$$

$\square$

## 7.4 The Simulation Lemma

**Lemma 7.19** (See Lemma 6.19) *Given* $M, N \in \Lambda_\oplus$

$$\exists D[\,] \in \Lambda_\oplus[\,].\ D[M]\!\downarrow \wedge D[N]\!\uparrow \Rightarrow\ \exists C[\,] \in \Lambda[\,].\ C[M]\!\downarrow \wedge C[N]\!\uparrow.$$

*Proof.* W.l.o.g. let us suppose that $M, N \in \Lambda_\oplus^0$; then exists $F \in \Lambda_\oplus$ s.t. $FM\!\downarrow$ and $FN\!\uparrow$. Let $F'$ be the term in $\Lambda$ obtained from $F$ by substituting all occurrences of a subterm of the form $P \oplus Q$ with an occurrence of $xPQ$, where $x$ is a fresh variable. For any $r \in \mathbb{N}$ define

$$T_r \equiv \lambda xyz_1 \ldots z_r w.w(xz_1 \ldots z_r)(yz_1 \ldots z_r).$$

We show that there exists an $F'' \in \Lambda$ and a vector $\vec{L} \in (Var \cup \{\mathbf{K}, \mathbf{O}, \omega\omega\})^*$ s.t.

$$F''M\vec{L}\!\downarrow \quad \text{and} \quad F''N\vec{L}\!\uparrow.$$

Let $\tau_1, \ldots, \tau_m$ be the set of the head reductions of $FM$, and $\sigma$ be any divergent head reduction of $FN$; let $\mathcal{F}$ be the set of $\oplus$ redexes of $F$.

Case 1: $\sigma$ doesn't contract any redex in $\mathcal{F}$: then choose $F'' \equiv F'$ and $\vec{L}$ is empty.

Case 2: $\tau_1, \ldots, \tau_m$ do not contract any redex in $\mathcal{F}$: then choose $F'' \equiv F'[\omega\omega/x]$.

Case 3: both $\sigma$ and $\tau_1, \ldots, \tau_m$ contract redexes in $\mathcal{F}$. Since $\tau_1, \ldots, \tau_m$ are finite, they are finitely often in $\mathcal{F}$, hence $k = \deg(\mathcal{F}, \tau_1, \ldots, \tau_m)$ for some $k$. We proceed as follows

- we choose an $r \geq k$ and take $F'' \equiv F'[T_r/x]$;
- we perform all possible head reductions of $F''M$ until either a head normal form is reached, or a term with $T_r$ in head position;
- we reduce $F''N$ until a term with $T_r$ in head position is reached: this must happen since $\sigma$ reduces some redex in $\mathcal{F}$ and no head normal form can be reached, otherwise we would have $FN\!\downarrow$.

Suppose that the term obtained in the reduction of $F''N$ is

$$\lambda\vec{x}.(T_r PQ)N_1 \ldots N_m$$

and, supposing $r$ choosen greater than $m$, the next steps in the head reduction of $F''N$ will give

$$U_0 \equiv \lambda\vec{x}z_{m+1}\ldots z_r w.w(PN_1\ldots N_m z_{m+1}\ldots z_r)(QN_1\ldots N_m z_{m+1}\ldots z_r).$$

We note that $w \notin \mathrm{FV}(PN_1\ldots N_m) \cup \mathrm{FV}(QN_1\ldots N_m)$. Correspondingly from the reductions of $F''M$ we get

$$\begin{array}{lcl} U_1 & \equiv & \lambda\vec{x}_1.(T_r P_1 Q_1)M_{1,1}\ldots M_{1,m_1} \\ & \cdots & \\ U_q & \equiv & \lambda\vec{x}_q.(T_r P_q Q_q)M_{q,1}\ldots M_{q,m_q} \\ U_{q+1} & \equiv & \lambda\vec{x}_{q+1}.\xi_{q+1}M_{q+1,1}\ldots M_{q+1,m_{q+1}} \\ & \cdots & \\ U_p & \equiv & \lambda\vec{x}_p.\xi_p M_{p,1}\ldots M_{p,m_p} \end{array}$$

and from these, for $1 \leq i \leq q$, the head reductions proceed giving certain $U_i'$ of the form

$$\lambda\vec{x}_i z_{m_i+1}\ldots z_r w.w(P_i M_{i,1}\ldots M_{i,m_i} z_{m_i+1}\ldots z_r)(Q_i M_{i,1}\ldots M_{i,m_i} z_{m_i+1}\ldots z_r),$$

where we make a similar remark about the $w$ as for $U_0$. Because of our assumptions all head variables appearing in the terms above are bound variables, hence they must occur in the prefixed string of abstractions of the respective terms. For any closed term in head normal form define its "head distance" as follows:

$$\mathrm{hd}(\lambda v_1\ldots v_n.\xi R_1\ldots R_m) = i \quad \text{if } \xi \equiv v_i.$$

Now we can always assume that for all $q+1 \leq i \leq p$

$$\mathrm{hd}(U_0) \neq \mathrm{hd}(U_i),$$

because we simply suppose the $r$ to be choosen suitably large. If this condition is satisfied also for $1 \leq i \leq q$ then we take $F'' \equiv F'[T_r/x]$ and $\vec{L} \equiv y_1 \ldots y_{h-1}(\omega\omega)$, where $h = \mathrm{hd}(U_0)$, and we are done.

However nothing prevents us from having some $U'_j$, where $1 \leq j \leq q$, s.t. $\mathrm{hd}(U'_j) = \mathrm{hd}(U_0)$, and of course this cannot be settled with a choice of $r$, since both head distances will depend on it. In this case suppose that the original reduction $\sigma$ has, after $\lambda\vec{x}.(P \oplus Q)N_1 \ldots N_m$, a choice to the left, namely it continues with $\lambda\vec{x}.PN_1 \ldots N_m$. In this case, if $l = \max\{|\vec{x}|, |\vec{x}_{q+1}|, \ldots, |\vec{x}_p|\}$, then take

$$\vec{L} \equiv y_1 \ldots y_{h-1}\mathbf{K}y_{h+1} \ldots y_l\vec{L'},$$

where $\vec{L'}$ remains to be determined. (Clearly, if the choice is to the right, we take $\mathbf{O}$ instead of $\mathbf{K}$). By the way

$$U'_i\vec{L}\downarrow \qquad \text{if } i \neq 0 \text{ and } \mathrm{hd}(U'_i) \neq \mathrm{hd}(U_0),$$

since the head variable will be replaced by some $y$, and the rest can be ignored; otherwise

$$U_0\vec{L} \xrightarrow{*}_h PN_1 \ldots N_m\vec{y},$$

and

$$U'_i\vec{L} \xrightarrow{*}_h P_iM_{i,1} \ldots M_{i,m_i}\vec{y}\,'$$

where $\vec{y}, \vec{y}\,' \subseteq y_1 \ldots y_{h-1}y_{h+1} \cdots y_l$.

If either $\sigma$ or $\tau_1, \ldots, \tau_m$ do not contract any other redex in $\mathcal{F}$, we are in a case similar to case 1 or to case 2: consequently we shall choose the $\vec{L'}$ accordingly. Otherwise the present case applies, and we repeat the same reasoning. This process, however, is bounded because the $\tau_1, \ldots, \tau_m$ were finitely often in $\mathcal{F}$. This implies that we must reach a point in which either (the simulation of) $\sigma$ definitely diverges, or all the reducts obtained from (the simulation of) $\tau_1, \ldots, \tau_m$ are similar to the $U_i$ above, when $q+1 \leq i \leq p$: that is we can suppose that they all have a different head distance from that of the term coming from $\sigma$. In the former case we add nothing to the $\vec{L}$ constructed up to that point; in the last case we add

$$w_1 \ldots w_s(\omega\omega),$$

supposing $s$ to be the head distance of the term coming from $\sigma$.

□

## 8 Conclusions and Further Work

We have studied non deterministic extensions of pure $\lambda$-calculus in their operational, denotational and axiomatic aspects. We have given some evidence of the conjunctive nature of the non determinism in the functional setting in contrast to the disjunctive nature of parallelism; furthermore, we have shown the possibility of grasping the former in a framework which is not disturbing, but rather consistently extending what has been known for the classical calculus.

The point of a consistent treatment of both concepts of non determinism and parallelism has been pursued to some extent in [40] in the perspective of lazy $\lambda$-calculus and we think it could be studied in the full calculus as well.

To take a step forward let us recall the idea underlying our treatment of the convergency predicate. We considered as convergent any term such that any (head) reduction starting with it gives something which can be considered as a value; for "sums" of terms this implies that they satisfy the convergency property if and only if all of the summands do. This can be generalized to any property, and formalized in a system which uses e.g. types to express properties of terms so that one could say that $M \oplus N$ has type $\sigma$ if and only if both $M$ and $N$ have type $\sigma$.

On the other hand we have argued for the disjunctive nature of parallelism, so that, if $M \parallel N$ represents the parallelization of $M$ and $N$, we say that $M \parallel N$ has type $\sigma$ provided that either $M$ or $N$ (or both) does.

In a system like Curry's simple types this does not seem to be expressive; we are instead considering, as a further development of the present work, a system like the one presented in [8], where one has both conjunctive and disjunctive types. Consistently with our perspective the following rules would be sound:

$$\frac{M : \sigma \qquad N : \tau}{M \oplus N : \sigma \vee \tau} \quad \text{and} \quad \frac{M : \sigma \qquad N : \tau}{M \parallel N : \sigma \wedge \tau}$$

where the exchange between conjunctive/disjunctive operators and disjunctive/conjunctive types is the effect of the usual duality.

It seems that this could provide even a framework in which the denotational semantics of the calculus would be assessed, along the lines of filter model technique introduced in [10], giving us the right place where questions about the semantical analysis of non determinism and parallelism could be profitably investigated and possibly answered.

## Acknowledgements

We are grateful to Corrado Böhm, Mariangiola Dezani-Ciancaglini, Rocco De Nicola, Gianfranco Mascari and Eugenio Moggi for helpful discussions and suggestions about

the topics of this paper.

## References


[1] S. Abramsky, "On Semantic Foundations for Applicative Multiprogramming" *LNCS* 154, 1983.

[2] S. Abramsky, C.H.L. Ong, *Full Abstraction in the Lazy Lambda Calculus*, Research Rep., Dept. of Comp., Imperial College 1989.

[3] J. Goguen, J. Thatcher, E. Wagner, J. Wrigth, "Initial algebraic semantics and continuous algebras", *J. ACM* 24, 1977.

[4] K.R. Apt, G.D. Plotkin, "Countable Non Determinism and Random Assignment", *J. ACM* 33, 1986.

[5] E.A. Ashcroft, M.C.B. Hennessy, "A mathematical Semantics for a Non Deterministic Typed Lambda Calculus", *TCS* 11, 1980.

[6] E. Astesiano, G. Costa, "Non Determinism and Fully Abstract Models", *R.A.I.R.O. Theor. Inf.* 14, 1980.

[7] J.C.M. Baeten, W.P. Weijland, *Process Algebra*, Cambridge University Press 1990.

[8] F. Barbanera, M. Dezani "Intersection and Union Types" Proc. of TACS'91, *LNCS* 526, 1991.

[9] H.P. Barendregt, *The Lambda-Calculus: Its Syntax and Semantics*, North-Holland, 1984.

[10] H. Barendregt, M. Coppo, M. Dezani "A Filter Lambda Model and the Completeness of Type Assignment" *JSL* 48, 1983.

[11] G. Berry, "Stable models of typed lambda-calculi", *LNCS* 62, 1978.

[12] G. Berry, G. Boudol "The Chemical Avstract Machine" in Proc. POPL '90, *ACM* 1990.

[13] C. Böhm, "Alcune proprietà delle forme $\beta$-$\eta$-normali nel $\lambda$-$K$-calcolo", Pubblicazioni dell' I.A.C. n. 696, Roma 1968.

[14] G. Boudol, "Towards a Lambda Calculus for Concurrent and Communicating Systems", in Proc. TAPSOFT '89, *LNCS* 351, 1989.

[15] G. Boudol, "A Lambda-Calculus for Parallel Functions", Report of INRIA, 1990.

[16] G. Boudol, "A Lambda Calculus for (Strict) Parallel Functions", Report of INRIA, 1990.

[17] M. Coppo, M. Dezani-Ciancaglini, S. Ronchi della Rocca, "(Semi)-separability of finite sets of terms in Scott's $D_\infty$-models of the $\lambda$-calculus", *LNCS* 62, 1978.

[18] U. de' Liguoro, "Non-deterministic untyped $\lambda$-calculus. A study about explicit non determinism in higher-order functional calculi", PhD Thesis December 1991 (available as internal report SI - 91/01, Dipartimento di Matematica, Università degli Studi di Roma "La Sapienza").

[19] R. De Nicola, M.C.B. Hennessy, "Testing Equivalences for Processes", *TCS* 34, 1983.

[20] I. Guessarian, "Algebraic Semantics", *LNCS* 99, 1981.

[21] C.A. Gunter, D. Scott, "Semantic Domains" in *Handbook of Theoretical Computer Science*, J. van Leeuwen (ed.), Elsevier Publ. Co., 1990.

[22] M.C.B. Hennessy, *Algebraic Theory of Processes*, MIT Press 1988.

[23] M.C.B. Hennessy, "The Semantics of Call-by-value and Call-by-name in a non deterministic Environment", *SIAM J. Comput.* 9, 1980.

[24] M.C.B. Hennessy, "Powerdomains and non deterministic recursive definitions", *LNCS* 137, 1982.

[25] M.C.B. Hennessy, G.D. Plotkin, "Full Abstraction for a Simple Parallel Programming Language", *LNCS* 74, 1979.

[26] J.R. Hindley, G. Longo, "Lambda Calculus Models and Extensionality", *Z. Math. Logik Grundlag. Math.* 26, 1980.

[27] C.A.R. Hoare, *Communicating Sequential Processes*, Prantice-Hall, 1985.

[28] M. Hyland, "A Syntactic Characterization of the Equality in some Models for the Lambda Calculus", *J. of the London Math. Soc.* 12, 1976.

[29] R. Jagadeesan, P. Panangaden, "A Domain-theoretic Model for a Higher-order Process Calculus", *LNCS* 443, 1990.

[30] J.W. Klop, "Combinatory reduction systems", *PhD thesis*, Mathematisch Centrum, Amsterdam 1980.

[31] J. Lambeek, P.J. Scott, *Introduction to Higher Order Categorical Logic*, Cambridge University Press, 1986.

[32] J.J. Levy, "An algebraic interpretation of the $\lambda\beta K$-calculus and a labelled $\lambda$-calculus", *LNCS* 37, 1975.

[33] J. Meseguer, "Conditional rewriting logic as a unified model of concurrency" *TCS* 96, 1992.

[34] R. Milner, "Fully Abstract Models for Typed $\lambda$-calculi", *TCS* 4, 1977.

[35] R. Milner, J.G. Parrow, D.J. Walker, "A Calculus of Mobile Processes, Parts I and II", Report of ECS-LFCS-89-85 and 86, Edinburgh Un., 1989.

[36] R. Milner, *Communication and Concurrency*, Prentice Hall, 1989.

[37] R. Milner, "Functions as Processes", *LNCS* 443, 1990.

[38] E. Moggi, "Notions of Computation and Monads", *Inf. Comp.* 93, 1991.

[39] J.H. Morris, *Lambda Calculus Models of Programming Languages*, Dissertation, M.I.T. 1968.

[40] C.-H.L. Ong, "Concurrent Lambda Calculus and a General Precongruence Theorem for Applicative Bisimulations", Draft, 1992.

[41] P. Panangaden, J.R. Russell "A Category-theoretic semantics for Unbounded Indeterminacy", *LNCS* 442, 1989.

[42] G.D. Plotkin, "Call-by-name, Call-by-value and the $\lambda$-calculus", *TCS* 1, 1975.

[43] G.D. Plotkin, "A Powerdomain Construction", *SIAM J. of Computing* 5, 1976.

[44] G.D. Plotkin, "LCF considered as a programming language", *TCS* 5, 1977.

[45] G.D. Plotkin, M. Smyth, "The category theoretic solution of recursive domain equations", *SIAM J. of Computing* 11, 1982.

[46] V. Pratt, "Event Spaces and their Linear Logic", *draft*, 1991.

[47] D. Sangiorgi, "The Lazy Lambda Calculus in a Concurrency Scenario", *LICS 92*, 1992.

[48] K. Sharma, *Syntactic Aspects of the Non-deterministic Lambda Calculus*, Master's thesis, Washington State University, September 1984. Available as internal report CS-84-127 of the Comp. Sci Dept.

[49] M.B. Smyth, "Power Domains", *J. Comp. Sys. Sci.* 16, 1978.

[50] B. Thomsen, "A Calculus of Higher-Order Communicating Systems", *ACM* 143, 1989.

[51] C.P. Wadsworth, " The relation between computational and denotational properties for Scott's $D_\infty$-models of the lambda-calculus", *SIAM J. of Computing* 5, 1976.