Full Abstraction for Resource Calculus with Tests[*]

Antonio Bucciarelli[1], Alberto Carraro[1,3], Thomas Ehrhard[1], and Giulio Manzonetto[2]

1 **Laboratoire PPS, CNRS, Université Paris-Diderot, Paris, France**
{antonio.bucciarelli,alberto.carraro,thomas.ehrhard}@pps.jussieu.fr
2 **Intelligent Systems, Radboud University, Nijmegen, The Netherlands**
g.manzonetto@cs.ru.nl
3 **Department of Computer Science, Ca'Foscari University, Venice, Italy**

Abstract

We study the semantics of a resource sensitive extension of the λ-calculus in a canonical reflexive object of a category of sets and relations, a relational version of the original Scott $\mathcal{D}_\infty$ model of the pure λ-calculus. This calculus is related to Boudol's resource calculus and is derived from Ehrhard and Regnier's differential extension of Linear Logic and of the λ-calculus. We extend it with new constructions, to be understood as implementing a very simple exception mechanism, and with a "must" parallel composition. These new operations allow to associate a context of this calculus with any point of the model and to prove full abstraction for the finite sub-calculus where ordinary λ-calculus application is not allowed. The result is then extended to the full calculus by means of a Taylor Expansion formula.

1998 ACM Subject Classification F.4.1 Lambda calculus and related systems

Keywords and phrases resource lambda calculus, relational semantics, full abstraction, differential linear logic

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

1 Introduction

In concurrent calculi like CCS [11], guarded processes are resources which can be used only once by other processes. This fundamental linearity of resources leads naturally to non-determinism, since several agents (senders and receivers) can interact on the same channel. In general, various synchronization scenarios are possible, giving rise to different behaviours. On the other hand in the λ-calculus, a function (receiver) can duplicate its argument (sender) arbitrarily. Thanks to this asymmetry, the λ-calculus enjoys a strong determinism (Church-Rosser), but it lacks any form of control on resource handling.

Resource Lambda Calculi. Resource λ-calculi stem from an attempt to combine the functionality of the λ-calculus and the resource sensitivity of process calculi. Boudol has been the first to design a resource conscious functional programming language, the *resource λ-calculus*, extending the usual one along two directions [2]: a function is not necessarily applied to a single argument but can also be applied to a multiset of arguments called *resources*; a resource can be either linear (it must be used exactly once) or reusable (it can be used *ad libitum*). In this context, the evaluation of a function applied to a multiset of resources gives rise to several possible choices, corresponding to the different possibilities of

[*] This work was partially supported by NWO Project 612.000.936 CALMOC (CAtegorical and ALgebraic Models of Computation) and ANR Project ANR-07-BLAN-0324 CHOCO.

Conference title on which this volume is based on.
Editors: Billy Editor, Bill Editors; pp. 1–15
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

distributing the resources in the multiset among the occurrences of the formal parameter. From the viewpoint of concurrent programming, this was a natural step to take since one of the main features of this programming setting is the consumption of resources which cannot be copied. Milner's π-calculus [12] features this phenomenon in great generality, and Boudol's calculus keeps track of it in a functional setting.

Together with Regnier, Ehrhard observed that this idea of resource consumption can be understood as resulting from a *differential* extension of λ-calculus (and of Linear Logic) [6]. Instead of considering two kinds of resources, they defined two kinds of applications: the *ordinary* application and a *linear* one. In a simply typed setting, linear application of a term $M : A \to B$ to a multiset made of n terms $N_1, \dots, N_n : A$, combined with ordinary application to a term $N : A$, corresponds to computing $M^{(n)}(N)(N_1, \dots, N_n) : B$, where $M^{(n)}$ is the n-th derivative of M, which is of type $A \to (A^n \to B)$ and associates a symmetric n-linear map with any element of A. The symmetry of this multilinear map corresponds to the Schwarz Lemma of differential calculus and is implemented in the resource λ-calculus by the use of multisets for representing linear applications.

The main difference between the resource λ-calculus and the differential λ-calculus is that the first is lazy and is endowed with an explicit substitution mechanism. Therefore, Boudol's calculus is not an extension of the ordinary λ-calculus. Also, the resource λ-calculus is rather affine than linear, since depletable resources cannot be duplicated but can be erased. Another difference lies in the respective origins of these calculi: the resource λ-calculus originates from syntactical considerations related to the theory of concurrent processes, while the differential one arises from denotational models of linear logic where the existence of differential operations has been observed. These models are based on the well known relational model of Linear Logic and the interpretation of the new differential constructions is as natural and simple as the interpretation of the ordinary LL constructions.

Two main syntaxes have been proposed for the differential λ-calculus: Ehrhard and Regnier's original one [6], simplified by Vaux in [16], and Tranquilli's *resource calculus* of [15] whose syntax is close to Boudol's one. These calculi share a common semantical backbone as well as similar connections with differential Linear Logic and proof nets. We adopt roughly Tranquilli's syntax and call our calculus *$\partial\lambda$-calculus.*

Full Abstraction. A natural open problem when a new calculus is introduced is to characterize when two programs are operationally equivalent, namely when one can be replaced by the other in every context without noticing any difference with respect to a given observational equivalence. In this paper we prove a full abstraction result (a semantical characterization of operational equivalence) for the $\partial\lambda$-calculus in the spirit of [3]. As in that paper, we extend the language with a convergence testing mechanism. Implicitly, this extension already appears in [5], in a differential LL setting: it corresponds to the 0-ary tensor and par cells. To implement the corresponding extension of the λ-calculus, we introduce two sorts of expressions: the *terms* (variable, application, abstraction, "throw" $\bar{\tau}(P)$ where P is a test) and the *tests* (empty test, parallel composition of tests and "catch" $\tau(M)$ where M is a term). Parallel composition allows to combine tests in such a way that the combination succeeds if and only if each test succeeds. Outcomes of tests (convergence or divergence) are the only observations allowed in our calculus, and the corresponding contextual equivalence and preorder on terms constitute our main object of study.

This extended $\partial\lambda$-calculus, that we call *$\partial\lambda$-calculus with tests*, has a natural denotational interpretation in a model of the pure λ-calculus introduced by Bucciarelli, Ehrhard and Manzonetto in [4], which is indeed a denotational model of the differential pure nets of [5] as one can check easily. This model is a reflexive object $\mathcal{D}$ in the Kleisli category of the LL model of sets and relations where $!X$ is the set of all finite multisets over X. An element of $\mathcal{D}$ can be described as a finite tree which alternates two kinds of layers: *multiplicative layers* where subtrees are indexed by natural numbers and *exponential layers* where subtrees are organized as non-empty multisets. To be more precise, $\parr - ?$ (negative) pairs of layers alternate with $\otimes - !$ (positive) pairs, respecting a strict polarity discipline very much in the spirit of Ludics [9]. The empty positive multiplicative tree corresponds to the empty tensor cell and the negative one to the empty par cell. The corresponding constructions τ, $\bar{\tau}$ are therefore quite easy to interpret.

We use this logical interpretation to turn the elements of $\mathcal{D}$ into $\partial\lambda$-calculus terms with tests. More precisely, with each element α of $\mathcal{D}$, we associate a test $\alpha^+(\!|\cdot|\!)$ with a hole $(\!|\cdot|\!)$ for a term, and we show that α belongs to the interpretation of a (closed) term M iff the test $\alpha^+(\!|M|\!)$ converges. From this fact, we derive a full abstraction result for the fragment of the $\partial\lambda$-calculus with tests in which all ordinary applications are trivial, that we call $\partial_0\lambda$-calculus with tests. To extend this result to the $\partial\lambda$-calculus with tests, we use the Taylor formula introduced in [6] which allows to turn any ordinary application into a sum of infinitely many linear applications of all possible arities. One exploits then the fact that the Taylor formula holds in the model, as well as a simulation lemma which relates the head reduction of a term with the head reduction of its Taylor expansion.

Contributions. The definability of the elements of $\mathcal{D}$ in the $\partial\lambda$-calculus with tests is the main conceptual contribution of this paper: it shows that, in the $\partial\lambda$-calculus with tests, the standard syntax versus semantics dichotomy is essentially meaningless. We also consider the use of the Taylor expansion to reduce the full abstraction problem to its $\partial_0\lambda$ version as an original and promising reduction technique. Notice that the tests added to the calculus are needed to develop this new methodology, although we conjecture they do not add discriminating power to the calculus (contrary to [3]).

Notations and basic definitions. We denote by $\mathbf{N}$ the set of natural numbers and by $\mathbf{1}$ an arbitrary singleton set. We write $\mathfrak{S}_k$ for *the set of all permutations of* $\{1, \dots, k\}$.

Let S be a set. We write $\mathcal{P}(S)$ (resp. $\mathcal{P}_\mathrm{f}(S)$) for the set of all (resp. finite) subsets of S. A *multiset* a over S is defined as an unordered list $a = [\alpha_1, \alpha_2, \dots]$ with repetitions such that $\alpha_i \in S$ for all indices i. A multiset a is called *finite* if it is a finite list, we denote by $\#a$ its cardinality. We write $\mathcal{M}_\mathrm{f}(S)$ for the set of all finite multisets over S. Given two multisets a, b we denote their *union* by $a \uplus b$. Given two finite sequences of multisets $\vec{a}, \vec{b}$ of the same length n we define $\vec{a} \uplus \vec{b} = (a_1 \uplus b_1, \dots, a_n \uplus b_n)$.

An operator $F(-)$ is *extended by linearity* by setting $F(\Sigma_i x_i) = \Sigma_i F(x_i)$.

2 The $\partial_0\lambda$-Calculus with Tests

We now introduce the $\partial_0\lambda$-*calculus with tests* which is the promotion-free fragment of the $\partial\lambda$-calculus with tests presented in Section 5. The $\partial_0\lambda$-calculus with tests has four syntactic categories: *terms* that are in functional position, *bags* that are in argument position and represent multisets of linear resources, *tests* that are "corked" multisets of terms having only two possible outcomes and *finite formal sums* representing all possible results of a computation. Formally, we have the following grammar:

$$
\begin{array}{ll}
(\Lambda^{\bar{\tau}})\ M, N, L, H ::= x \mid \lambda x.M \mid MP \mid \bar{\tau}(Q) & \text{terms} \\
(\Lambda^{b})\ P ::= [L_1, \dots, L_k] & \text{bags} \\
(\Lambda^{\tau})\ Q, R ::= \tau[L_1, \dots, L_k] & \text{tests} \\
(\Lambda^{e})\ A, B ::= M \mid P \mid Q & \text{expressions}
\end{array}
$$

Tests are multisets of terms, the “τ” being a tag for distinguishing them from bags.

Throughout the paper, we will enforce the distinction between bags and tests by using systematically the following notational conventions.

- For bags, we use the usual multiset notation: $[]$ is the empty bag and $P \uplus P'$ is the union.
- For tests, ε is the empty multiset and $Q|R$ is the multiset union of Q and R. In other words, $\varepsilon = \tau[]$ and $\tau[L_1, \dots, L_k] \mid \tau[L_{k+1}, \dots, L_n] = \tau[L_1, \dots, L_n]$.

Terms are the real protagonists of the $\partial_0\lambda$-calculus with tests. The term $\lambda x.M$ represents the λ-*abstraction* and MP the *application* of a term M to a bag P of linear resources. Thus, in $(\lambda x.M)P$, each resource in P is available exactly once for $\lambda x.M$ and if the number of occurrences of x in M “disagrees” with the cardinality of P then the result is 0 (see later, when sums are introduced). We set $\mathbf{I} := \lambda x.x$, where ‘:=’ denotes definitional equality.

Tests are expressions which can produce two results: either *success*, represented by ε, or *failure*, represented by 0. The test $Q|R$ represents the (must-)parallel composition of Q and R (i.e., $Q|R$ succeeds if both Q and R succeed). The composition is parallel in the sense that the order of evaluation is inessential.

The operator $\bar{\tau}(\cdot)$ allows to build a term out of a test: intuitively, the term $\bar{\tau}(Q)$ may be thought of as Q preceded by an infinite sequence of dummy λ-abstractions. Dually, the “cork construction” $\tau[L_1, \dots, L_k]$ may be thought of as an operator applying to all its arguments an infinite sequence of empty bags. This suggests that it is sound to reduce $\tau[\bar{\tau}(Q)]$ to Q.

Hence the term $\bar{\tau}(Q)$ raises an exception encapsulating Q and the test $\tau[L_1, \dots, L_k]$ catches the exception possibly raised by any of the L_i’s and replaces L_i by the multiset of terms encapsulated in that exception. The context of the exception is thrown away by the dummy abstractions of $\bar{\tau}$ and the dummy applications of τ. A test needs to catch an exception in order to succeed; for instance, $\tau[M]$ fails as soon as M is a $\bar{\tau}$-free, closed term.

We will write $\|_{i=1}^{n} R_i$ for $R_1|\cdots|R_n$; obviously we have $\|_{i=1}^{0} R_i = \varepsilon$ and $\|_{i=1}^{1} R_i = R_1$.

Expressions are either terms, bags or tests.

Sums. Let $\mathbf{2}$ be the semiring $\{0, 1\}$ with $1 + 1 = 1$ and multiplication defined in the obvious way. For any set A, we write $\mathbf{2}\langle A\rangle$ for the free $\mathbf{2}$-module generated by A, so that $\mathbf{2}\langle A\rangle \cong \mathcal{P}_{\mathrm{f}}(A)$ with addition corresponding to union, and scalar multiplication defined in the obvious way. However we prefer to keep the algebraic notations for elements of $\mathbf{2}\langle A\rangle$, hence set unions will be denoted by $+$ and the empty set by 0. This amounts to say that $\mathbf{2}\langle\Lambda^{\bar{\tau}}\rangle$ (resp. $\mathbf{2}\langle\Lambda^{\tau}\rangle$, $\mathbf{2}\langle\Lambda^{b}\rangle$) is the set of finite formal sums of terms (resp. tests, bags) with an idempotent sum. We also set $\mathbf{2}\langle\Lambda^{e}\rangle = \mathbf{2}\langle\Lambda^{\tau}\rangle \cup \mathbf{2}\langle\Lambda^{\bar{\tau}}\rangle \cup \mathbf{2}\langle\Lambda^{b}\rangle$. This is an abuse of notation as $\mathbf{2}\langle\Lambda^{e}\rangle$ here does not denote the $\mathbf{2}$-module generated over $\Lambda^{\tau} \cup \Lambda^{\bar{\tau}} \cup \Lambda^{b}$, but rather the union of the three $\mathbf{2}$-modules; this means that sums should be taken only in the same sort.

Typical metavariables to denote sums are: $\mathbb{M}, \mathbb{N}, \mathbb{L}, \mathbb{H} \in \mathbf{2}\langle\Lambda^{\bar{\tau}}\rangle$, $\mathbb{P} \in \mathbf{2}\langle\Lambda^{b}\rangle$, $\mathbb{Q}, \mathbb{R} \in \mathbf{2}\langle\Lambda^{\tau}\rangle$, $\mathbb{A}, \mathbb{B} \in \mathbf{2}\langle\Lambda^{e}\rangle$. The α-*equivalence* relation and the set $\mathrm{FV}(\mathbb{A})$ of *free variables of* $\mathbb{A}$ are defined as usual, like in the ordinary λ-calculus [1]. We write $\deg_x(\mathbb{A})$ for the number of free occurrences of x in $\mathbb{A}$. Hereafter, (sums of) expressions are considered up to α-equivalence.

2.1 Two Kinds of Substitutions

Notice that the grammar for terms and tests does not include any sums, so they may arise only on the “surface”. However, as syntactic sugar – and *not* as actual syntax – we extend all the constructors to sums by multilinearity, setting for instance $(\Sigma_i M_i)(\Sigma_j P_j) := \Sigma_{i,j} M_i P_j$, in such a way that the following equations hold:

$$\begin{array}{llll} \lambda x.(\Sigma_i M_i) = \Sigma_i \lambda x.M_i & \mathbb{M}(\Sigma_i P) = \Sigma_i \mathbb{M} P_i & (\Sigma_i M_i)\mathbb{P} = \Sigma_i M_i \mathbb{P} & \tau[\Sigma_i M_i] = \Sigma_i \tau[M_i] \\ (\Sigma_i R_i) \mid \mathbb{Q} = \Sigma_i\ R_i \mid \mathbb{Q} & [\Sigma_i L_i] = \Sigma_i [L_i] & (\Sigma_i P_i) \uplus \mathbb{P} = \Sigma_i P_i \uplus \mathbb{P} & \bar{\tau}(\Sigma_i R_i) = \Sigma_i \bar{\tau}(R_i) \end{array}$$

As an example of this *extended (meta-)syntax*, we may write $(x_1 + x_2)[y_1 + y_2]$ for $x_1[y_1] + x_1[y_2] + x_2[y_1] + x_2[y_2]$. This kind of meta-syntactic notation is discussed thoroughly in [8].

Observe that in the particular case of empty sums, we get $\lambda x.0 := 0$, $M0 := 0$, $0P := 0$, $\tau[0] := 0$, $\bar\tau(0) := 0$, $R|0 := 0$, $[0] := 0$, $0 \uplus P := 0$. Thus 0 annihilates any term, bag or test.

We now introduce two kinds of substitutions: the usual λ-calculus substitution and a linear one, which is proper to differential and resource calculi (see [2, 6, 15]).

Let $A \in \Lambda^e$ and $N \in \Lambda^{\bar\tau}$. The *(capture-free) substitution of N for x in A*, denoted by $A\{N/x\}$, is defined as usual. Accordingly, $A\{\mathbb{N}/x\}$ denotes a term of the extended syntax. Finally, we extend this operation to sums as in $\mathbb{A}\{\mathbb{N}/x\}$ by linearity in $\mathbb{A}$.

The *linear (capture-free) substitution of N for x in A*, denoted by $A\langle N/x\rangle$, is defined as follows (in this definition we strongly use the extended syntax):

$$y\langle N/x\rangle = \begin{cases} N & \text{if } y = x,\\ 0 & \text{otherwise,}\end{cases} \qquad \begin{aligned}[]
[L_1,\dots,L_k]\langle N/x\rangle &= \Sigma_{i=1}^{k}[L_1,\dots,L_i\langle N/x\rangle\dots,L_k],\\
\tau[L_1,\dots,L_k]\langle N/x\rangle &= \Sigma_{i=1}^{k}\tau[L_1,\dots,L_i\langle N/x\rangle,\dots,L_k],
\end{aligned}$$

$$(MP)\langle N/x\rangle = M\langle N/x\rangle P + M(P\langle N/x\rangle), \qquad \bar\tau(Q)\langle N/x\rangle = \bar\tau(Q\langle N/x\rangle),$$

$$(\lambda y.M)\langle N/x\rangle = \lambda y.M\langle N/x\rangle, \qquad \text{(in the abstraction case we assume wlog } x \neq y).$$

Roughly speaking, linear substitution replaces the resource to *exactly one* linear free occurrence of x. If there is no occurrence of x then the result is 0. In presence of multiple occurrences, all possible choices are made and the result is the sum of them. For example, we have $(y[x][x])\langle \mathbf{I}/x\rangle = y[\mathbf{I}][x] + y[x][\mathbf{I}]$.

An example of regular substitution is $(x[x])\{(z_1+z_2)/x\} = z_1[z_1]+z_1[z_2]+z_2[z_1]+z_2[z_2]$.

Turning to the extension of linear substitution to sums: the term $A\langle\mathbb{N}/x\rangle$ belongs to the extended syntax, and we extend it to sums as in $\mathbb{A}\langle\mathbb{N}/x\rangle$ by linearity in $\mathbb{A}$, as we did for usual substitution.

Observe that $\mathbb{A}\langle\mathbb{N}/x\rangle$ is linear in $\mathbb{A}$ and in $\mathbb{N}$, whereas $\mathbb{A}\{\mathbb{N}/x\}$ is linear in $\mathbb{A}$ but not in $\mathbb{N}$.

Linear substitutions commute in the sense expressed by the next lemma.

▶ **Lemma 1** (Schwarz Lemma, cf. [6]). *For $\mathbb{A} \in \mathbf{2}\langle\Lambda^e\rangle$, $\mathbb{M},\mathbb{N} \in \mathbf{2}\langle\Lambda^{\bar\tau}\rangle$ and $y \notin \mathrm{FV}(\mathbb{M})\cup\mathrm{FV}(\mathbb{N})$ we have $\mathbb{A}\langle\mathbb{M}/y\rangle\langle\mathbb{N}/x\rangle = \mathbb{A}\langle\mathbb{N}/x\rangle\langle\mathbb{M}/y\rangle + \mathbb{A}\langle\mathbb{M}\langle\mathbb{N}/x\rangle/y\rangle$. In particular, if $x \notin \mathrm{FV}(\mathbb{M})$ then the two substitutions commute.*

Given a bag $P = [L_1,\dots,L_k]$ such that $x \notin \mathrm{FV}(P)$ it makes sense to define $\mathbb{A}\langle P/x\rangle := \mathbb{A}\langle L_1/x\rangle\cdots\langle L_k/x\rangle$, because this expression does not depend on the enumeration $L_1,\dots,L_k$. In particular, $\mathbb{A}\langle []/x\rangle = \mathbb{A}$. Given bags $P_1,\dots,P_n$ we set $\mathbb{A}\langle\vec{P}/\vec{x}\rangle := \mathbb{A}\langle P_1/x_1\rangle\cdots\langle P_n/x_n\rangle$.

2.2 The Operational Semantics

We are going to introduce the reduction rules defining the operational semantics of the $\partial_0\lambda$-calculus with tests and show that it enjoys Church-Rosser and strong normalization, even in the untyped version of the calculus.

▶ **Definition 2.** The reduction semantics of the $\partial_0\lambda$-calculus with tests is generated by the following rules (in the abstraction case we suppose wlog that $x \notin \mathrm{FV}(P)$):

$$(\lambda x.M)P \to_\beta M\langle P/x\rangle\{0/x\}, \qquad \bar\tau(Q)P \to_{\bar\tau} \begin{cases}\bar\tau(Q) & \text{if } P = [],\\ 0 & \text{otherwise,}\end{cases}$$

$$\tau[\lambda x.M]|R \to_\tau \tau[M\{0/x\}]|R, \qquad \tau[\bar\tau(Q)]|R \to_\gamma Q|R.$$

Notice that the reduction preserves the sort of an expression in the sense that terms rewrite to (sums of) terms and tests to (sums of) tests. Also remark that, if M has k free occurrences of x (represented by $x^1,\dots,x^k$) then we have $M\langle L_1/x\rangle\cdots\langle L_k/x\rangle\{0/x\} = \Sigma_{\sigma\in\mathfrak{S}_k} M\{L_{\sigma(1)}/x^1,\dots,L_{\sigma(k)}/x^k\}$; it is equal to 0 otherwise (namely, when $\deg_x(M) \neq k$).

We denote by $\to \subseteq \mathbf{2}\langle\Lambda^e\rangle \times \mathbf{2}\langle\Lambda^e\rangle$ the contextual closure of $\to_\beta \cup \to_{\bar\tau} \cup \to_\tau \cup \to_\gamma$. In particular, parallel composition is treated asynchronously, thus $R \to \mathbb{R}$ entails $Q|R \to Q|\mathbb{R}$ (which is equal to $\mathbb{R}|Q$). This means, for instance, that if $L \to \bar\tau(Q)$, then $\tau[L, \vec{N}] \to \tau[\bar\tau(Q), \vec{N}] \to Q \mid \tau[\vec{N}]$. We write $\twoheadrightarrow$ for the transitive and reflexive closure of $\to$.

▶ **Definition 3.** An expression A is *in normal form* (*nf*, for short) if there is no $\mathbb{B}$ such that $A \to \mathbb{B}$. A sum of expressions $\mathbb{A}$ *is in nf* if $\mathbb{A} \neq 0$ and all its summands are in nf.

It is easy to check that a term $M \in \Lambda^{\bar\tau}$ is in normal-form if either $M = \lambda\vec{x}.yP_1 \cdots P_n$ or $M = \lambda\vec{x}.\bar\tau(\|_{i=1}^n \tau[y_i P_1^i \cdots P_{k_i}^i])$ where $n \geq 0$, $k_i \geq 0$ and each P_j^i is a bag of terms in nf.

▶ **Theorem 4.** *The $\partial_0\lambda$-calculus with tests is strongly normalizing and Church-Rosser.*

Proof. The fact that there are no infinite reduction chains is trivial, since every reduction step decreases the size of an expression (which is straightforward to define). For the Church-Rosser property just check local confluence and conclude by Newman's lemma. ◀

▶ **Lemma 5.** *For any closed term M, either $\tau[M] \twoheadrightarrow \varepsilon$ or $\tau[M] \twoheadrightarrow 0$.*

Proof. As $\partial_0\lambda$-calculus with tests is strongly normalizing, we have that $M \twoheadrightarrow \Sigma_{i=1}^k M_i$, where each M_i is a closed nf. If $k = 0$ then $\tau[M] \twoheadrightarrow 0$ since $\tau[0] = 0$. Otherwise for each M_i there are two possibilities:

- $M_i = \lambda\vec{x}.x_j P_1 \cdots P_n$ with $x_j \in \vec{x}$ and $n \geq 0$. Then $\tau[M_i] \twoheadrightarrow \tau[(x_j P_1 \cdots P_n)\{0/\vec{x}\}] = \tau[0] = 0$.
- $M_i = \lambda\vec{x}.\bar\tau(\|_{j=1}^n \tau[x_j P_1^j \cdots P_{k_j}^j])$ with $n \geq 0$ and $x_j \in \vec{x}$. If $n = 0$ then we have $\|_{j=1}^n \tau[x_j P_1^j \cdots P_{k_j}^j] = \varepsilon$ and $\tau[\lambda\vec{x}.\bar\tau(\varepsilon)] \twoheadrightarrow \tau[\bar\tau(\varepsilon)] \to \varepsilon$. If $n > 0$, then we have $\tau[M_i] \twoheadrightarrow \tau[\bar\tau(\|_{j=1}^n \tau[0P_1^j\{0/\vec{x}\} \cdots P_{k_j}^j\{0/\vec{x}\}])] = 0$.

We conclude since $\tau[M] \twoheadrightarrow \Sigma_{i=1}^k \tau[M_i]$, and this latter expression reduces to a finite (possibly empty) sum of ε's, which is thus equal either to 0 or to ε. ◀

▶ **Corollary 6.** *If R is a closed test then either $R \twoheadrightarrow \varepsilon$ or $R \twoheadrightarrow 0$.*

Contexts. A *test-context* $C(\!|\cdot|\!)$ is a test having one occurrence of a *hole*, denoted by $(\!|\cdot|\!)$, appearing in term-position. The set of test-contexts is denoted by $\Lambda^\tau_{(\!|\cdot|\!)}$. Given $M \in \Lambda^{\bar\tau}$ we indicate by $C(\!|M|\!)$ the test resulting by blindly replacing M for the hole (allowing capture of free variables) in $C(\!|\cdot|\!)$. We say that $C(\!|\cdot|\!)$ is *closed* if it contains no free variable; it is *closing* M if $C(\!|M|\!)$ is closed. We say that a test Q *converges*, and we write $Q\downarrow$, if $Q \twoheadrightarrow \varepsilon$.

▶ **Definition 7.** The *operational pre-order* $\sqsubseteq_{\mathcal{O}}$ is defined by:

$$M \sqsubseteq_{\mathcal{O}} N \Leftrightarrow \forall C(\!|\cdot|\!) \in \Lambda^\tau_{(\!|\cdot|\!)} \text{ closing } M, N \ (C(\!|M|\!)\downarrow \Rightarrow C(\!|N|\!)\downarrow).$$

We set $M \approx_{\mathcal{O}} N$ if and only if $M \sqsubseteq_{\mathcal{O}} N$ and $N \sqsubseteq_{\mathcal{O}} M$.

The choice of test-convergence as the basic observation in our calculus is very natural. Indeed, tests provide a canonical notion of observation since – by design – they either converge (to ε) or diverge.

3 A Relational Semantics

This section is devoted to build a relational model $\mathcal{D}$ of $\partial_0\lambda$-calculus with tests, that has been first introduced in [4] as a model of the ordinary λ-calculus. We first give a sketchy presentation of the Cartesian closed category **MRel** where $\mathcal{D}$ lives.

The objects of $\mathbf{MRel}$ are all the sets. A morphism from S to T is a relation from $\mathcal{M}_{\mathrm{f}}(S)$ to T, in other words, $\mathbf{MRel}(S,T) = \mathcal{P}(\mathcal{M}_{\mathrm{f}}(S) \times T)$. The identity of S is the relation $\mathrm{Id}_S = \{([\alpha],\alpha) : \alpha \in S\}$. The composition of $s : S \to T$ and $t : T \to U$ is defined by:

$$t \circ s = \{(m,c) : \exists (m_1,\beta_1),...,(m_k,\beta_k) \in s \text{ such that } m = \uplus_{i=1}^{k} m_i \text{ and } ([\beta_1,...,\beta_k],c) \in t\}.$$

The categorical product $S \,\&\, T$ of two sets S and T is their disjoint union. The terminal object is the empty set $\emptyset$. The exponential object internalizing $\mathbf{MRel}(S,T)$ is $\mathcal{M}_{\mathrm{f}}(S) \times T$.

An infinite sequence $\alpha = (a_1, a_2, \dots)$ of multisets is *quasi-finite* if $a_i = []$ holds for all but a finite number of indices i. If S is a set, we denote by $\mathcal{M}_{\mathrm{f}}(S)^{(\omega)}$ the set of all quasi-finite $\mathbf{N}$-indexed sequences of multisets over S.

We build a family of sets $(D_n)_{n\in\mathbf{N}}$ as follows: $D_0 = \emptyset$, $D_{n+1} = \mathcal{M}_{\mathrm{f}}(D_n)^{(\omega)}$. Since the operation $S \mapsto \mathcal{M}_{\mathrm{f}}(S)^{(\omega)}$ is monotonic with respect to inclusion and $D_0 \subseteq D_1$, we have $D_n \subseteq D_{n+1}$ for all $n \in \mathbf{N}$. Finally, we set $\mathcal{D} = \bigcup_{n\in\mathbf{N}} D_n$.

To define an isomorphism between $\mathcal{D}$ and $\mathcal{M}_{\mathrm{f}}(\mathcal{D}) \times \mathcal{D}$ just remark that every element $\alpha = (a_1, a_2, a_3, \dots) \in \mathcal{D}$ stands for the pair $(a_1, (a_2, a_3, \dots))$ and *vice versa*. Hence $\mathcal{D} \cong [\mathcal{D} \Rightarrow \mathcal{D}]$ (we have a canonical bijection between these two sets, and therefore an isomorphism in $\mathbf{MRel}$). Given $\alpha = (a_1, a_2, a_3, \dots) \in \mathcal{D}$ and $a \in \mathcal{M}_{\mathrm{f}}(\mathcal{D})$, we write $a :: \alpha$ for the element $(a, a_1, a_2, a_3, \dots) \in \mathcal{D}$. We set $* = ([], [], \dots, [], \dots) \in \mathcal{D}$. Remark that $[] :: * = *$.

3.1 Interpreting the $\partial_0\lambda$-calculus with tests

For all terms M, bags P, tests Q and repetition-free sequences $\vec{x}, \vec{y}, \vec{z}$ respectively containing the free variables of M, P, Q, we define by mutual induction the interpretations $[\![M]\!]_{\vec{x}} : \mathcal{D}^n \to \mathcal{D}$, $[\![P]\!]_{\vec{y}} : \mathcal{D}^m \to \mathcal{M}_{\mathrm{f}}(\mathcal{D})$ and $[\![Q]\!]_{\vec{z}} : \mathcal{D}^k \to \mathbf{1}$ (n, m, k are the lengths of $\vec{x}, \vec{y}, \vec{z}$) as follows[1]:

- $[\![x_i]\!]_{\vec{x}} = \{(([], \dots, [], [\alpha], [], \dots, []), \alpha) : \alpha \in \mathcal{D}\}$, where $[\alpha]$ stands in i-th position,
- $[\![\lambda y.M]\!]_{\vec{x}} = \{(\vec{a}, b :: \alpha) : ((\vec{a}, b), \alpha) \in [\![M]\!]_{\vec{x},y}\}$, where we suppose wlog that $y \notin \vec{x}$,
- $[\![MP]\!]_{\vec{x}} = \{(\vec{a}_0 \uplus \vec{a}_1, \alpha) : \exists b \in \mathcal{M}_{\mathrm{f}}(\mathcal{D})\ (\vec{a}_0, b :: \alpha) \in [\![M]\!]_{\vec{x}},\ (\vec{a}_1, b) \in [\![P]\!]_{\vec{x}}\}$,
- $[\![[L_1, \dots, L_k]]\!]_{\vec{x}} = \{(\uplus_{i=1}^{k} \vec{a}_i, [\beta_1, \dots, \beta_k]) : (\vec{a}_i, \beta_i) \in [\![L_i]\!]_{\vec{x}},\ 1 \le i \le k\}$,
- $[\![\bar{\tau}(Q)]\!]_{\vec{x}} = \{(\vec{a}, *) : \vec{a} \in [\![Q]\!]_{\vec{x}}\}$,
- $[\![\tau[M]]\!]_{\vec{x}} = \{\vec{a} : (\vec{a}, *) \in [\![M]\!]_{\vec{x}}\}$,
- $[\![Q|R]\!]_{\vec{x}} = \{\vec{a}_0 \uplus \vec{a}_1 : \vec{a}_0 \in [\![Q]\!]_{\vec{x}}, \vec{a}_1 \in [\![R]\!]_{\vec{x}}\}$,
- $[\![\varepsilon]\!]_{\vec{x}} = \{([], \dots, [])\}$.

The interpretation is then extended to sums by setting $[\![\Sigma_{i=1}^{k} A_i]\!]_{\vec{x}} = \cup_{i=1}^{k} [\![A_i]\!]_{\vec{x}}$. Note that $[\![[]]\!]_{\vec{x}} = \{([], \dots, [])\} \in \mathcal{M}_{\mathrm{f}}(\mathcal{D})^{n+1}$. Since every test R is of the form $\tau[L_1, \dots, L_k]$ we might define its interpretation directly as $[\![R]\!]_{\vec{x}} = \{\uplus_{i=1}^{k} \vec{a}_i : (\vec{a}_i, *) \in [\![L_i]\!]_{\vec{x}}, 1 \le i \le k\}$.

Hereafter, whenever we write $[\![A]\!]_{\vec{x}}$ we suppose that $\vec{x}$ is a repetition-free list of variables of length n containing $\mathrm{FV}(A)$. Moreover, we will sometimes silently use the fact $[\![M]\!]_{\vec{x},y} = \{((\vec{a}, []), \alpha) : (\vec{a}, \alpha) \in [\![M]\!]_{\vec{x}}\}$ whenever $y \notin \vec{x}$.

Clearly the interpretation is monotonic, i.e., for any test context $C(\!|\cdot|\!)$ with free variables $\vec{y}$, if $[\![M]\!]_{\vec{x}} \subseteq [\![N]\!]_{\vec{x}}$ then $[\![C(\!|M|\!)]\!]_{\vec{x},\vec{y}} \subseteq [\![C(\!|N|\!)]\!]_{\vec{x},\vec{y}}$.

The following substitution lemmas are needed for proving the invariance of the interpretation under reduction. The proofs are lengthy but not difficult, and are omitted.

[1] Since $\mathcal{M}_{\mathrm{f}}(S \,\&\, T) \cong \mathcal{M}_{\mathrm{f}}(S) \times \mathcal{M}_{\mathrm{f}}(T)$ we have, up to isomorphism, $[\![M]\!]_{\vec{x}} \subseteq \mathcal{M}_{\mathrm{f}}(\mathcal{D})^n \times \mathcal{D}$, $[\![P]\!]_{\vec{y}} \subseteq \mathcal{M}_{\mathrm{f}}(\mathcal{D})^{m+1}$ and $[\![Q]\!]_{\vec{z}} \subseteq \mathcal{M}_{\mathrm{f}}(\mathcal{D})^k \times \mathbf{1} \cong \mathcal{M}_{\mathrm{f}}(\mathcal{D})^k$.

▶ **Lemma 8** (Linear Substitution Lemma). *Let $M \in \Lambda^{\bar{\tau}}, Q \in \Lambda^{\tau}$ and $P = [L_1, \dots, L_k] \in \Lambda^b$ such that $\deg_y(M) = \deg_y(Q) = k$. We have:*

(i) $(\vec{a}, \alpha) \in [\![M\langle P/y\rangle]\!]_{\vec{x}}$ *iff there exist* $(\vec{a}_i, \beta_i) \in [\![L_i]\!]_{\vec{x}}$ *(for* $1 \leq i \leq k$*) and* $\vec{a}_0 \in \mathcal{M}_{\mathrm{f}}(\mathcal{D})^n$ *such that* $((\vec{a}_0, [\beta_1, \dots, \beta_k]), \alpha) \in [\![M]\!]_{\vec{x},y}$ *and* $\uplus_{i=0}^{k} \vec{a}_i = \vec{a}$.

(ii) $\vec{a} \in [\![Q\langle P/y\rangle]\!]_{\vec{x}}$ *iff there exist* $(\vec{a}_i, \beta_i) \in [\![L_i]\!]_{\vec{x}}$ *(for* $1 \leq i \leq k$*) and* $\vec{a}_0 \in \mathcal{M}_{\mathrm{f}}(\mathcal{D})^n$ *such that* $(\vec{a}_0, [\beta_1, \dots, \beta_k]) \in [\![Q]\!]_{\vec{x},y}$ *and* $\uplus_{i=0}^{k} \vec{a}_i = \vec{a}$.

▶ **Lemma 9** (Regular Substitution Lemma). *Let $M \in \Lambda^{\bar{\tau}}$, $Q \in \Lambda^{\tau}$ and $\mathbb{N} \in \mathbf{2}\langle\Lambda^{\bar{\tau}}\rangle$. We have:*

(i) $(\vec{a}, \alpha) \in [\![M\{\mathbb{N}/y\}]\!]_{\vec{x}}$ *iff* $\exists k \in \mathbf{N}$, $\exists \beta_1, \dots, \beta_k \in \mathcal{D}$, $\exists \vec{a}_0, \dots, \vec{a}_k \in \mathcal{M}_{\mathrm{f}}(\mathcal{D})^n$ *such that* $(\vec{a}_i, \beta_i) \in [\![\mathbb{N}]\!]_{\vec{x}}$ *(for* $1 \leq i \leq k$*),* $((\vec{a}_0, [\beta_1, \dots, \beta_k]), \alpha) \in [\![M]\!]_{\vec{x},y}$ *and* $\vec{a} = \uplus_{j=0}^{k} \vec{a}_j$*,*

(ii) $\vec{a} \in [\![Q\{\mathbb{N}/y\}]\!]_{\vec{x}}$ *iff* $\exists k \in \mathbf{N}$, $\exists \beta_1, \dots, \beta_k \in \mathcal{D}$, $\exists \vec{a}_0, \dots, \vec{a}_k \in \mathcal{M}_{\mathrm{f}}(\mathcal{D})^n$ *such that* $(\vec{a}_i, \beta_i) \in [\![\mathbb{N}]\!]_{\vec{x}}$ *(for* $1 \leq i \leq k$*) and* $(\vec{a}_0, [\beta_1, \dots, \beta_k]) \in [\![Q]\!]_{\vec{x},y}$ *and* $\vec{a} = \uplus_{j=0}^{k} \vec{a}_j$*.*

The substitution lemmas above generalize straightforwardly to sums. Although Lemma 9 is stated in full generality, for the $\partial_0\lambda$-calculus with tests it is only useful for $\mathbb{N} = 0$. However, this formulation will be needed in Section 5 for the $\partial\lambda$-calculus with tests.

▶ **Theorem 10.** *$\mathcal{D}$ is a model of the $\partial_0\lambda$-calculus with tests, i.e., if $\mathbb{A} \twoheadrightarrow \mathbb{B}$ then $[\![\mathbb{A}]\!]_{\vec{x}} = [\![\mathbb{B}]\!]_{\vec{x}}$.*

Proof. It is easy to check that the interpretation is contextual. The fact that the semantics is invariant under reduction follows from Lemmas 8 and 9. ◀

4 First Full Abstraction Results

A model is *equationally fully abstract* (*FA*, for short) if the equivalence induced on terms by their interpretations is exactly $\approx_{\mathcal{O}}$; it is *inequationally FA* if the induced preorder is $\sqsubseteq_{\mathcal{O}}$. Every inequationally FA model is also FA. In this section we prove that $\mathcal{D}$ is inequationally FA for the $\partial_0\lambda$-calculus (Thm. 19), i.e., that $[\![M]\!]_{\vec{x}} \subseteq [\![N]\!]_{\vec{x}}$ iff $M \sqsubseteq_{\mathcal{O}} N$.

4.1 Building Separating Test-Contexts

In this section we are going to associate a test-context $\alpha^{+}(\!|\cdot|\!)$ with each element $\alpha \in \mathcal{D}$, the idea being that – for every closed term M – we have $\alpha \in [\![M]\!]$ iff $\alpha^{+}(\!|M|\!)$ converges.

▶ **Definition 11.** Let $\alpha \in \mathcal{D}$. The *rank of* α, written $\mathrm{rk}(\alpha)$, is the least $n \in \mathbf{N}$ such that $\alpha \in D_{n+1}$; the *length of* α, written $\ell(\alpha)$, is 0 if $\alpha = *$, and it is the unique r such that $\alpha = a_1 :: \cdots :: a_r :: *$ with $a_r \neq []$, otherwise.

Note that if $\alpha = a_1 :: \cdots :: a_r :: *$ then for all $1 \leq i \leq r$ and for all $\alpha' \in a_i$ we have $\mathrm{rk}(\alpha) > \mathrm{rk}(\alpha')$. Hence $\mathrm{rk}(\alpha) = 0$ entails $\alpha = *$ and the following definition is well-founded.

▶ **Definition 12.** For $\alpha \in \mathcal{D}$ of the form $\alpha = [\alpha_1^1, \dots, \alpha_{k_1}^1] :: \cdots :: [\alpha_1^r, \dots, \alpha_{k_r}^r] :: *$ with $\ell(\alpha) = r$, define by mutual induction a closed term α^{-} and a test-context $\alpha^{+}(\!|\cdot|\!)$ as follows:

- $\alpha^{-} = \lambda x_1 \dots x_r.\bar{\tau}(\|_{i=1}^{r} ((\alpha_1^i)^{+}(\!|x_i|\!) | \cdots | (\alpha_{k_i}^i)^{+}(\!|x_i|\!)))$,
- $\alpha^{+}(\!|\cdot|\!) = \tau[(\!|\cdot|\!)[(\alpha_1^1)^{-}, \dots, (\alpha_{k_1}^1)^{-}] \cdots [(\alpha_1^r)^{-}, \dots, (\alpha_{k_r}^r)^{-}]]$.

Given $a = [\alpha_1, \dots, \alpha_k]$ we set $a^{-} = [\alpha_1^{-}, \dots, \alpha_k^{-}]$.

For instance, we have $*^{-} = \bar{\tau}(\varepsilon)$ (as the empty parallel composition is equal to ε) and $*^{+}(\!|\cdot|\!) = \tau[(\!|\cdot|\!)]$.

The next lemma, along with its corollaries, shows the interplay between the elements of $\mathcal{D}$ and the terms/tests of Definition 12. It provides the main motivation for our extension of the $\partial\lambda$-calculus.

▶ **Lemma 13.** *Let $\alpha \in \mathcal{D}$. Then:*
(i) $\llbracket \alpha^- \rrbracket = \{\alpha\}$,
(ii) $\llbracket \alpha^+ \llparenthesis x \rrparenthesis \rrbracket_x = \{[\alpha]\}$.

Proof. The points (i) and (ii) are proved simultaneously by induction on $\mathrm{rk}(\alpha)$. We write IH(i) and IH(ii) for the induction hypotheses concerning (i) and (ii), respectively.

If $\mathrm{rk}(\alpha) = 0$ then $\alpha = *$, hence $\llbracket *^- \rrbracket = \llbracket \bar{\tau}(\varepsilon) \rrbracket = \{*\}$ and $\llbracket *^+ \llparenthesis x \rrparenthesis \rrbracket_x = \llbracket \tau[x] \rrbracket_x = \{[*]\}$.

If $\mathrm{rk}(\alpha) > 0$ and $\ell(\alpha) = r$, then we have $\alpha = a_1 :: \cdots :: a_r :: *$ with $a_i = [\alpha^i_1, \dots, \alpha^i_{k_i}]$ for $1 \le i \le r$.

We prove (i). Remember that by definition $\llbracket \alpha^- \rrbracket = \llbracket \lambda y_1 \dots y_r . \bar{\tau}(\|_{i=1}^r \|_{j=1}^{k_i} (\alpha^i_j)^+ \llparenthesis y_i \rrparenthesis) \rrbracket$. So we have $\beta \in \llbracket \alpha^- \rrbracket$ iff $\beta = b_1 :: \cdots :: b_r :: *$ and for all $1 \le i \le r$, $1 \le j \le k_i$ there is $\vec{d}^{\,i}_j \in \llbracket (\alpha^i_j)^+ \llparenthesis y_i \rrparenthesis \rrbracket_{\vec{y}}$ such that $\vec{b} = \biguplus_{i=1}^r \biguplus_{j=1}^{k_i} \vec{d}^{\,i}_j$. By IH(ii) we have $\vec{d}^{\,i}_j \in \llbracket (\alpha^i_j)^+ \llparenthesis y_i \rrparenthesis \rrbracket_{\vec{y}}$ iff $\vec{d}^{\,i}_j = (\vec{[]}, [\alpha^i_j], \vec{[]})$ where $[\alpha^i_j]$ appears in i-th position. Therefore $\biguplus_{j=1}^{k_i} \vec{d}^{\,i}_j = (\vec{[]}, a_i, \vec{[]})$ and $b_i = a_i$ for every index i. Thus $\beta = \alpha$.

We prove (ii). By def. $\llbracket \alpha^+ \llparenthesis x \rrparenthesis \rrbracket_x = \llbracket \tau[x a_1^- \cdots a_r^-] \rrbracket_x$. So we have $c \in \llbracket \alpha^+ \llparenthesis x \rrparenthesis \rrbracket_x$ iff there are $b_i = [\beta^i_i, \dots, \beta^i_{k_i}]$, $c_0, c^i_1, \dots, c^i_{k_i} \in \mathcal{M}_{\mathrm{f}}(\mathcal{D})$ (for $1 \le i \le r$) such that $(c_0, b_1 :: \cdots :: b_r :: *) \in \llbracket x \rrbracket_x$, $(c^i_j, \beta^i_j) \in \llbracket (\alpha^i_j)^- \rrbracket_x$ (for all $1 \le i \le r$ and $1 \le j \le k_i$) and $c = c_0 \uplus (\biguplus_{i=1}^r \biguplus_{j=1}^{k_i} c^i_j)$. As, by IH(i), $\llbracket (\alpha^i_{j_i})^- \rrbracket_x = \{([], \alpha^i_j)\}$ we get $c^i_j = []$ and $\beta^i_j = \alpha^i_j$. Thus $c = c_0$, $\alpha = b_1 :: \cdots :: b_r :: *$ and from this it follows that $(c, \alpha) \in \llbracket x \rrbracket_x$. We conclude that $c = [\alpha]$. ◀

▶ **Corollary 14.** $\llbracket \alpha^+ \llparenthesis M \rrparenthesis \rrbracket_{\vec{x}} = \{\vec{c} : (\vec{c}, \alpha) \in \llbracket M \rrbracket_{\vec{x}}\}$.

Proof. By Lemma 13(ii) we have $\llbracket \alpha^+ \llparenthesis y \rrparenthesis \rrbracket_{\vec{x},y} = \{([], \dots, [], [\alpha])\}$. As $\alpha^+ \llparenthesis \cdot \rrparenthesis$ does not have outer λ-abstractions we have $\alpha^+ \llparenthesis M \rrparenthesis = \alpha^+ \llparenthesis y \rrparenthesis \langle [M]/y \rangle$. We then apply Lemma 8 to conclude. ◀

▶ **Corollary 15.** *All finite subsets of $\mathcal{D}$ are definable.*

Proof. By Lemma 13(i), for every finite set $u = \{\alpha_1, \dots, \alpha_k\}$ we have $\llbracket \alpha_1^- + \cdots + \alpha_k^- \rrbracket = u$. ◀

Lemma 13 reveals the behaviour of a test-context $\alpha^+ \llparenthesis \cdot \rrparenthesis$ when applied to a term β^-.

▶ **Corollary 16.** *Let $\alpha, \beta \in \mathcal{D}$. If $\alpha = \beta$ then $\alpha^+ \llparenthesis \beta^- \rrparenthesis \twoheadrightarrow \varepsilon$, otherwise $\alpha^+ \llparenthesis \beta^- \rrparenthesis \twoheadrightarrow 0$.*

Proof. By Lemma 13, $\llbracket \alpha^+ \llparenthesis \beta^- \rrparenthesis \rrbracket = \{()\} \subseteq \mathcal{M}_{\mathrm{f}}(\mathcal{D})^0$ if $\alpha = \beta$, $\emptyset$ otherwise. By Corollary 6, we know that $\alpha^+ \llparenthesis \beta^- \rrparenthesis$ reduces either to ε or to 0. The result follows by soundness (Thm. 10). ◀

4.2 (In)equational Full Abstraction

We now show that the operational preorder $\sqsubseteq_{\mathcal{O}}$ (Def. 7) coincides with the inclusion of interpretations in $\mathcal{D}$. The proof of this full abstraction result needs some preliminary lemmas.

▶ **Lemma 17.** *Let $Q \in \Lambda^\tau$, $\mathrm{FV}(Q) \subseteq \vec{x}$ and $\vec{a} \in \mathcal{M}_{\mathrm{f}}(\mathcal{D})^n$. Then $\vec{a} \in \llbracket Q \rrbracket_{\vec{x}} \Leftrightarrow \llbracket Q \langle \vec{a}^-/\vec{x} \rangle \rrbracket \neq \emptyset$ and $\deg_{x_i}(Q) = \# a_i$.*

Proof. By applying n times (one for each variable in $\vec{x}$) Lemma 8 and Corollary 14. ◀

The ensuing lemma is the key argument for proving that $\mathcal{D}$ is inequationally fully abstract.

▶ **Lemma 18.** *Let $M \in \Lambda^{\bar{\tau}}$, $\vec{x} \supseteq \mathrm{FV}(M)$, $\alpha \in \mathcal{D}$, $\vec{a} \in \mathcal{M}_{\mathrm{f}}(\mathcal{D})$. The following are equivalent:*
(i) $(\vec{a}, \alpha) \in \llbracket M \rrbracket_{\vec{x}}$,
(ii) $\alpha^+ \llparenthesis M \langle \vec{a}^-/\vec{x} \rangle \rrparenthesis \downarrow$.

Proof. We have the following chain of equivalences:

$(\vec{a}, \alpha) \in \llbracket M \rrbracket_{\vec{x}} \Leftrightarrow \vec{a} \in \llbracket \alpha^+ \llparenthesis M \rrparenthesis \rrbracket_{\vec{x}}$, by Corollary 14,

- $\Leftrightarrow [\![\alpha^+ (\!| M\langle \vec{a}^- / \vec{x} \rangle |\!)]\!] \neq \emptyset$ and $\deg_{x_i}(M) = \# a_i$, by Lemma 17, using $(\alpha^+ (\!| M |\!))\langle \vec{a}^- / \vec{x} \rangle = \alpha^+ (\!| M\langle \vec{a}^- / \vec{x} \rangle |\!)$,
- $\Leftrightarrow \alpha^+ (\!| M\langle \vec{a}^- / \vec{x} \rangle |\!) \twoheadrightarrow \varepsilon$, by Corollary 6, i.e. the fact that closed tests can only reduce to either ε or 0, and Theorem 10, i.e. the soundness of the model. ◀

▶ **Theorem 19.** *$\mathcal{D}$ is inequationally fully abstract for the $\partial_0\lambda$-calculus with tests:*

$$[\![M]\!]_{\vec{x}} \subseteq [\![N]\!]_{\vec{x}} \Leftrightarrow M \sqsubseteq_{\mathcal{O}} N$$

Proof. ($\Rightarrow$) Assume that $[\![M]\!]_{\vec{x}} \subseteq [\![N]\!]_{\vec{x}}$, and let $C(\!|\cdot|\!)$ be a context closing both M and N and such that $C(\!|M|\!) \twoheadrightarrow \varepsilon$. By Thm. 10, $[\![C(\!|M|\!)]\!] = [\![\varepsilon]\!] = \{()\}$. By monotonicity of the interpretation we get $[\![C(\!|M|\!)]\!] \subseteq [\![C(\!|N|\!)]\!]$, thus $[\![C(\!|N|\!)]\!] \neq \emptyset$. By Cor. 6 this entails $C(\!|N|\!)\Downarrow$.

($\Leftarrow$) Suppose, by the way of contradiction, that $M \sqsubseteq_{\mathcal{O}} N$ holds but there is an element $(\vec{a}, \alpha) \in [\![M]\!]_{\vec{x}} - [\![N]\!]_{\vec{x}}$. Then the test-context $C(\!|\cdot|\!) = \alpha^+ (\!| (\lambda \vec{x}.(\!|\cdot|\!))\vec{a}^- |\!)$ is such that $C(\!|M|\!) \twoheadrightarrow \alpha^+ (\!| M\langle \vec{a}^- / \vec{x} \rangle |\!) \twoheadrightarrow \varepsilon$ and $C(\!|N|\!) \not\twoheadrightarrow \varepsilon$ by Lemma 18. This leads to a contradiction. ◀

The rest of the paper is devoted to extend the above result to the $\partial\lambda$-calculus with tests. The main ingredients will be the Taylor expansion and the head-reduction introduced in Subsections 6.1 and 5.1, respectively.

5 The $\partial\lambda$-Calculus with Tests

The *$\partial\lambda$-calculus with tests* is an extension of the $\partial_0\lambda$-calculus with tests with a promotion available on resources. In this calculus a resource can be linear (it must be used exactly once) or not (it can be used *ad libitum*) and in the latter case it is decorated with a "!" superscript.

Syntax. The grammar generating the terms, the tests and the expressions of the $\partial\lambda$-calculus with tests, is the same as the one for the $\partial_0\lambda$-calculus with tests (in particular tests are still plain multisets of linear resources), except for the rule concerning bags which becomes:

$$P ::= [L_1, \dots, L_k, \mathbb{N}^!] \qquad\qquad \text{bags}$$

where $\mathbb{N}$ is a finite sum of terms of this new syntax. We write $\Lambda_!^{\bar{\tau}}$ for the set of terms generated by this new grammar, $\Lambda_!^{\tau}$ for the set of tests, $\Lambda_!^{b}$ for the set of bags, $\Lambda_!^{e}$ for the set of expressions. From now on bags are no more plain multisets of terms: they are compound objects, consisting of a multiset of terms $[L_1, \dots, L_k]$ and a sum of terms $\mathbb{N}$, denoted as $[L_1, \dots, L_k, \mathbb{N}^!]$. We shall deal with them as if they were multisets, defining union by $[L_1, \dots, L_k, \mathbb{N}^!] \uplus [L_{k+1}, \dots, L_n, \mathbb{M}^!] := [L_1, \dots, L_n, (\mathbb{N} + \mathbb{M})^!]$. This operation is commutative, associative and has $[0^!]$ as neutral element.

The $\partial_0\lambda$-calculus with tests is the sub-calculus of $\partial\lambda$-calculus with tests in which all bags have shape $[L_1, \dots, L_k, 0^!]$, and this identification is compatible with the reduction rules.

As in the $\partial_0\lambda$-calculus with tests, we extend this syntax by multilinearity to sums of expressions with the only exception that the bag $[\vec{L}, (\mathbb{N} + \mathbb{M})^!]$ *is not* required to be equal to $[\vec{L}, \mathbb{N}^!] + [\vec{L}, \mathbb{M}^!]$. The intuition is that in the first expression $\mathbb{N} + \mathbb{M}$ can be used several times and each time one can choose either $\mathbb{N}$ or $\mathbb{M}$, whereas in the second expression one has to choose once and for all one of the summands, and then use it as many times as needed.

Substitutions. Linear substitution is denoted and defined as in the $\partial_0\lambda$-calculus with tests, except of course for bags, where we set:

$$[L_1, .., L_k, \mathbb{N}^!]\langle N/x \rangle = \Sigma_{i=1}^{k} [L_1, .., L_i\langle N/x \rangle, .., L_k, \mathbb{N}^!] + \ [L_1, .., L_k, \mathbb{N}\langle N/x \rangle, \mathbb{N}^!]\,.$$

For example, $(x[x^!])\langle y/x\rangle\langle z/x\rangle = y[z,x^!] + z[y,x^!] + x[y,z,x^!]$. Remark that in the !-free case, that is when $\mathbb{N} = 0$, the above definitions and notations agree with those introduced in Subsection 2.1, because in that case we have $[L_1,\dots,L_k,\mathbb{N}\langle N/x\rangle,\mathbb{N}^!] = 0$, since $0\langle N/x\rangle = 0$.

We also define the regular substitution $A\{\mathbb{N}/x\}$ for the $\partial\lambda$-calculus with tests, by simply replacing each occurrence of x in the expression A with $\mathbb{N}$: in that way we get an expression of the extended syntax, since $\mathbb{N}$ is a sum in general. E.g., $x[x^!]\{(y+z)/x\} = y[y^!,z^!] + z[y^!,z^!]$.

Both substitutions are then generalized to sums: linear substitution is extended to $\mathbb{A}\langle\mathbb{N}/x\rangle$ by bilinearity in $\mathbb{A}$ and $\mathbb{N}$, while ordinary substitution to $\mathbb{A}\{\mathbb{N}/x\}$ by linearity in $\mathbb{A}$.

A Schwarz lemma, analogous to Lemma 1, holds for the $\partial\lambda$-calculus with tests. Hence, given a sum of expressions $\mathbb{A}$ and a bag $P = [L_1,\dots,L_k]$ with $x \notin \mathrm{FV}(P)$, it still makes sense to set $\mathbb{A}\langle P/x\rangle := \mathbb{A}\langle L_1/x\rangle\cdots\langle L_k/x\rangle$ because this expression does not depend on the enumeration of $L_1,\dots,L_k$. In particular we have $\mathbb{A}\langle[]/x\rangle = \mathbb{A}$.

Operational semantics. The reduction rules of $\partial\lambda$-calculus extend those of the $\partial_0\lambda$-calculus with tests in the sense that they are equivalent on !-free expressions.

The rules (τ) and (γ) are exactly the same, while the β-reduction and $\bar\tau$-reduction are rephrased as follows:

- $(\lambda x.M)[L_1,..,L_k,\mathbb{N}^!] \to_\beta M\langle[L_1,..,L_k]/x\rangle\{\mathbb{N}/x\}$, where wlog $x \notin \mathrm{FV}([L_1,...,L_k])$,
- $\bar\tau(Q)[L_1,\dots,L_k,\mathbb{N}^!] \to_{\bar\tau} \begin{cases} \bar\tau(Q) & \text{if } k=0,\\ 0 & \text{otherwise.}\end{cases}$

The $\partial\lambda$-calculus with tests is still Church-Rosser (just adapt the proof in [14]), while it is no more strongly normalizing. For instance the term $\Omega = (\lambda x.x[x^!])[(\lambda x.x[x^!])^!]$ has an infinite reduction chain, just like the paradigmatic homonymous unsolvable λ-term. Indeed, the usual λ-calculus can be embedded into the $\partial\lambda$-calculus with tests by translating every application MN into $M[N^!]$.

In this framework a *test-context* $C\llparenthesis\cdot\rrparenthesis$ is a test of the $\partial\lambda$-calculus with tests having a single occurrence of its *hole*, appearing in term-position. The set of test-contexts is denoted by $\Lambda^{\tau!}_{\llparenthesis\cdot\rrparenthesis}$.

A test Q *converges*, notation $Q\!\downarrow$, if there exists a sum $\mathbb{Q}$ such that $Q \twoheadrightarrow \varepsilon + \mathbb{Q}$.

▶ **Definition 20.** The *operational pre-order* $\sqsubseteq^!_{\mathcal{O}}$ on the $\partial\lambda$-calculus with tests is defined by:

$$M \sqsubseteq^!_{\mathcal{O}} N \Leftrightarrow \forall C\llparenthesis\cdot\rrparenthesis \in \Lambda^{\tau!}_{\llparenthesis\cdot\rrparenthesis} \text{ closing } M,N\ (C\llparenthesis M\rrparenthesis\!\downarrow \Rightarrow C\llparenthesis N\rrparenthesis\!\downarrow).$$

We then set $M \approx^!_{\mathcal{O}} N$ if and only if $M \sqsubseteq^!_{\mathcal{O}} N$ and $N \sqsubseteq^!_{\mathcal{O}} M$.

Relational semantics. The $\partial\lambda$-calculus with tests can be interpreted into $\mathcal{D}$ by extending the interpretation of the $\partial_0\lambda$-calculus with tests as follows:

$$\begin{aligned}\llbracket[L_1,\dots,L_k,\mathbb{N}^!]\rrbracket_{\vec{x}} = \{(\uplus_{r=1}^{k+m}\vec{a}_r,[\beta_1,\dots,\beta_{k+m}]) : {}& (\vec{a}_j,\beta_j) \in \llbracket L_j\rrbracket_{\vec{x}}, 1 \le j \le k \text{ and}\\ & (\vec{a}_i,\beta_i) \in \llbracket\mathbb{N}\rrbracket_{\vec{x}},\ k < i \le k+m\}.\end{aligned}$$

It is easy to check that both Lemma 8 and Lemma 9 generalize to this context. From these lemmas it ensues that $\mathcal{D}$ is also a model of the $\partial\lambda$-calculus with tests.

▶ **Theorem 21.** *$\mathcal{D}$ is a model of $\partial\lambda$-calculus with tests.*

5.1 Head Reduction

We now provide a notion of *head-reduction* for the $\partial\lambda$-calculus with tests. Intuitively, the head-reduction is obtained by reducing a head-redex, that is a redex occurring in head-position in an expression A. The interest of introducing this reduction strategy is that it "behaves well" with respect to the Taylor expansion in the sense of Proposition 31.

We start by defining the notion of redex.

$$x^\circ = \{x\}, \qquad (\lambda x.M)^\circ = \{\lambda x.M' : M' \in M^\circ\}, \qquad (MP)^\circ = \{M'P' : M' \in M^\circ,\ P' \in P^\circ\},$$
$$(\bar\tau(Q))^\circ = \{\bar\tau(Q') : Q' \in Q^\circ\}, \quad (\tau[M_1,\dots,M_k])^\circ = \{\tau[M'_1,\dots,M'_k] : M'_i \in M_i^\circ, \text{ for } 1 \le i \le k\},$$
$$[L_1,\dots,L_k,\mathbb{N}^!]^\circ = \{[L'_1,\dots,L'_k] \uplus P : L'_i \in L_i^\circ, \text{ for } 1 \le i \le k,\ P \in \mathcal{M}_{\mathrm{f}}(\mathbb{N}^\circ)\},$$
$$(\Sigma_{i=1}^k A_i)^\circ = \cup_{i=1}^k A_i^\circ.$$

Figure 1 The *Taylor expansion* $\mathbb{A}^\circ$ of $\mathbb{A} \in \mathbf{2}\langle\Lambda^e_!\rangle$.

▶ **Definition 22.** A *term-redex* is any term of the form $(\lambda x.M)P$ or $\bar\tau(Q)P$. A *test-redex* is any test of the form $\tau[\lambda x.M]$, $\tau[\bar\tau(Q)]$.

Among term- and test-redexes we distinguish those redexes that are in "head" position.

▶ **Definition 23.** A *head-redex* is:

- either a term-redex H in terms of shape $\lambda\vec{y}.H\vec{P}$,
- or a term-redex H in tests of shape $\tau[H\vec{P}]|Q$,
- or a test-redex R in tests of shape $R|Q$.

▶ **Definition 24.** We say that $A \to \mathbb{B}$ is a step of *head-reduction* if $\mathbb{B}$ is obtained from A by contracting a head-redex. If $A \to \mathbb{B}$ is a step of head-reduction then also $A + \mathbb{A} \to \mathbb{B} + \mathbb{A}$ is.

One-step head-reduction is denoted by $\to_h$, while $\twoheadrightarrow_h$ indicates its reflexive and transitive closure. Notice that, unlike in ordinary λ-calculus, an expression A may have more than one head-redex, hence there may be more than one head-reduction steps starting from A.

The head-reduction induces a notion of head-normal form on (finite sums of) expressions.

▶ **Definition 25.** An expression A *is in head-normal form* (*hnf*, for short) if there is no $\mathbb{B}$ such that $A \to_h \mathbb{B}$; a sum $\mathbb{A}$ *is in hnf* if $\mathbb{A} \neq 0$ and each summand is in hnf.

This notion of hnf differs from that given by Pagani and Ronchi della Rocca in [13]. We keep this name since their definition captures the notion of "outer-nf" rather than that of hnf.

It is easy to check that a term M is in hnf iff $M := \lambda\vec{x}.y\vec{P}$ or $M := \lambda\vec{x}.\bar\tau(Q)$; a test R is in hnf iff $R := \varepsilon$, $R := \tau[x\vec{P}]$ or $R := Q_1|Q_2$ for some tests Q_1, Q_2 in hnf.

The following two lemmas concern reduction properties of !*-free closed* tests.

▶ **Lemma 26.** *Let $R \in \Lambda^\tau$. If R is closed and $R \neq \varepsilon$ then it has a head-redex (hence, $R \to_h \mathbb{R}'$ for some $\mathbb{R}'$).*

Proof. By induction on R. It suffices to consider the case $R = \tau[M]$. We then proceed by cases on the structure of M (which must be closed). If $M = \lambda x.N$ then R head-reduces using (τ). If M is an application then it must be written either as $M = (\lambda y.N)P_1 \cdots P_k$ or as $M = \bar\tau[Q]P_1 \cdots P_k$ (in both cases $k \geq 1$) and hence R head-reduces using either (β) or $(\bar\tau)$, respectively. If $M = \bar\tau(Q)$ then R head-reduces using (γ). ◀

▶ **Lemma 27.** *If $R \in \Lambda^\tau$ is closed then $R \twoheadrightarrow \varepsilon$ iff $R \twoheadrightarrow_h \varepsilon$.*

Proof. $(\Rightarrow)$ Suppose, by contradiction, that $R \twoheadrightarrow \varepsilon$ but $R \not\twoheadrightarrow_h \varepsilon$. By confluence (Thm. 4), we cannot have $R \twoheadrightarrow_h 0$. Thus, since $R \in \Lambda^\tau$ is strongly normalizing, the only way to have $R \not\twoheadrightarrow_h \varepsilon$ is that $R \twoheadrightarrow_h \mathbb{R}$ where $\mathbb{R} \neq \varepsilon$ is in hnf. This is impossible by Lemma 26.

$(\Leftarrow)$ Trivial since $\twoheadrightarrow_h \subseteq \twoheadrightarrow$. ◀

One should be careful when trying to extend the above result to terms $M \in \Lambda^{\bar\tau}$. For instance, it is false that $M \twoheadrightarrow 0$ iff $M \twoheadrightarrow_h 0$. Indeed $M := \lambda x.x[\mathbf{I}[]]$ is in hnf but $M \to \lambda x.x[0] := 0$.

6 Full Abstraction via Taylor Expansion

In this section we are going to define the Taylor expansion of terms and tests of the $\partial\lambda$-calculus with tests. We will then use this expansion, combined with head-reduction, to generalize the full abstraction results obtained in Subsection 4.2 to the framework of $\partial\lambda$-calculus with tests.

6.1 Taylor Expansion

The *(full) Taylor expansion* was first introduced in [6, 7], in the context of λ-calculus. The Taylor expansion M° of an ordinary λ-term M gives an infinite formal linear combination of terms of the $\partial_0\lambda$-calculus. In the case of ordinary application it looks like:

$$(MN)^\circ = \sum_{n=0}^{\infty} \frac{1}{n!} M[\underbrace{N, \dots, N}_{n \text{ times}}]$$

in accordance with the intended meaning and the denotational semantics of application in the resource calculus. In the syntax of differential λ-calculus [6] the above formula looks like $\sum_{n=0}^{\infty} \frac{1}{n!} M^{(n)}(0)(N, \dots, N)$, hence the connection with analytical Taylor expansion is clear.

Following [10], we extend the definition of Taylor expansion from ordinary λ-terms to the expressions of the $\partial\lambda$-calculus with tests. Since the sum is idempotent, the coefficients disappear and our Taylor expansion corresponds to the *support* of the actual Taylor expansion.

As the set $\mathbf{2}\langle\Lambda^e\rangle_\infty$ of possibly infinite formal sums of expressions is isomorphic to $\mathcal{P}(\Lambda^e)$, in the following we may use sets instead of sums.

▶ **Definition 28.** Let $\mathbb{A} \in \mathbf{2}\langle\Lambda^e_!\rangle$. The *(full) Taylor expansion of* $\mathbb{A}$ is the set $\mathbb{A}^\circ \subseteq \Lambda^e$ which is defined (by structural induction on $\mathbb{A}$) in Figure 1.

As previously announced, the Taylor expansion of an expression A can be infinite. For example, we have that $(\lambda x.x[x^!])^\circ = \{\lambda x.x[x^n] : n \in \mathbf{N}\}$.

To lighten the notations, we adopt for infinite sets of expressions the same abbreviations as introduced for finite sums in Subsection 2.1 (including those for substitutions). For instance, if $X, Y \subseteq \Lambda^{\bar\tau}$ then $\lambda x.X$ denotes the set $\{\lambda x.M' : M' \in X\}$ and $X\langle Y/x\rangle = \cup_{M\in X, N\in Y} M\langle N/x\rangle$.

In [10] it is proved that **MRel** is a differential Cartesian closed category that "models the Taylor expansion". This property entails that Taylor expansion preserves the meaning of an expression in $\mathcal{D}$, as expressed in the next theorem.

▶ **Theorem 29.** *$[\![\mathbb{A}]\!]_{\vec{x}} = \cup_{A\in\mathbb{A}^\circ} [\![A]\!]_{\vec{x}}$, for all $\mathbb{A} \in \mathbf{2}\langle\Lambda^e_!\rangle$.*

Proof. By adapting the proof in [10] of the analogous result for the differential λ-calculus. ◀

We now need the following lemma stating the commutation of Taylor expansion with respect to ordinary and linear substitutions. The proof is lengthy but not difficult and is omitted. For the sake of readability, in the next statements we use sums and unions interchangeably.

▶ **Lemma 30.** *Let $A \in \Lambda^e_!$, $N \in \Lambda^{\bar\tau}_!$ and $\mathbb{N} \in \mathbf{2}\langle\Lambda^{\bar\tau}_!\rangle$. Then, for $x \notin \mathrm{FV}(N) \cup \mathrm{FV}(\mathbb{N})$:*

(i) $(A\langle N/x\rangle)^\circ = A^\circ\langle N^\circ/x\rangle$,

(ii) $(A\{\mathbb{N}/x\})^\circ = \bigcup_{P\in\mathcal{M}_\mathrm{f}(\mathbb{N}^\circ)} A^\circ\langle P/x\rangle\{0/x\}$.

The next proposition is devoted to show how Taylor expansion interacts with head-reduction. To ease the formulation of the next proposition we assimilate $\mathbf{2}\langle\Lambda^e_!\rangle$ to $\mathcal{P}_\mathrm{f}(\Lambda^e_!)$.

▶ **Proposition 31.** *Let $A \in \Lambda^e_!$ and let $A' \in A^\circ$ be such that $A' \to_h \mathbb{B}'$, for some $\mathbb{B}'$. Then there exists $\mathbb{B}$ such that $A \to_h \mathbb{B}$ and $\mathbb{B}' \subseteq \mathbb{B}^\circ$.*

Proof. The idea is that the syntactic tree of A has the same structure as that of A' and we can define a surjective mapping of the redexes of A' into those of A.

We only treat the case $A' = \lambda\vec{x}.H'P'_1 \cdots P'_p$ where $H' = (\lambda y.M')P'$ is a head-redex. From $A' \in A^\circ$ we get $A = \lambda\vec{x}.HP_1 \cdots P_p$ for some H such that $H' \in H^\circ$. Hence, supposing wlog $P' = [\vec{L}', \vec{N}']$, we have that $H = (\lambda y.M)[\vec{L}, \mathbb{N}^!]$ where $M' \in M^\circ$, the lengths of $\vec{L}'$ and $\vec{L}$ coincide, $L'_i \in L^\circ_i$ for all i and $[\vec{N}'] \in \mathcal{M}_{\mathrm{f}}(\mathbb{N}^\circ)$. We now know that $H' \to_h M'\langle[\vec{L}']/y\rangle\langle[\vec{N}']/y\rangle\{0/y\}$ and $H \to_h M\langle[\vec{L}]/y\rangle\{\mathbb{N}/y\}$. By Lemma 30, $(M\langle[\vec{L}]/y\rangle\{\mathbb{N}/y\})^\circ = \cup_{P\in\mathcal{M}_{\mathrm{f}}(\mathbb{N}^\circ)} M^\circ\langle[\vec{L}^\circ]/y\rangle\langle P/y\rangle\{0/y\} \supseteq M\langle P'/y\rangle\{0/y\}$.

We can conclude that $\lambda\vec{x}.M'\langle P'/y\rangle\{0/y\}P'_1 \cdots P'_p \subseteq (\lambda\vec{x}.M\langle[\vec{L}]/y\rangle\{\mathbb{N}/y\}P_1 \cdots P_p)^\circ$. ◀

Note that the above proposition is false for regular β-reduction. E.g., take $A := x[(\mathbf{I}[y])^!]$ and $A' := x[\mathbf{I}[y], \mathbf{I}[y]] \in A^\circ$, then $A' \to_\beta x[y, \mathbf{I}[y]]$ and $A \to_\beta x[y^!]$ but $x[y, \mathbf{I}[y]] \notin (x[y^!])^\circ$.

▶ **Corollary 32.** *Let $R \in \Lambda^\tau_!$ be closed. If there is an $R' \in R^\circ$ such that $R' \twoheadrightarrow \varepsilon$, then $R{\downarrow}$.*

Proof. Suppose that there exists $R' \in R^\circ$ such that $R' \twoheadrightarrow \varepsilon$. By Lemma 27 there is a head-reduction chain of the form $R' \to_h \mathbb{R}'_1 \to_h \cdots \to_h \mathbb{R}'_n = \varepsilon$. By iterated application of a corollary[2] of Prop. 31 there are tests $\mathbb{R}_i$ (for $i = 1, \ldots, n$) such that $R \to_h \mathbb{R}_1 \to_h \cdots \to_h \mathbb{R}_n$ with $\mathbb{R}'_i \subseteq \mathbb{R}^\circ_i$. We conclude since $\varepsilon \in \mathbb{R}^\circ_n$ is only possible when $\varepsilon \in \mathbb{R}_n$. ◀

6.2 Full Abstraction for the $\partial\lambda$-Calculus with Tests

We now prove that the relational model $\mathcal{D}$ is fully abstract for the $\partial\lambda$-calculus with tests.

▶ **Lemma 33.** *Given $A \in \Lambda^e_!$ and $M \in \Lambda^{\bar{\tau}}_!$ we have:*

(i) $(\alpha^+(\!|M|\!))^\circ = \alpha^+(\!|M^\circ|\!)$*, for all* $\alpha \in \mathcal{D}$*,*
(ii) $(A\langle a^-/x\rangle)^\circ = A^\circ\langle a^-/x\rangle$*, for all* $a \in \mathcal{M}_{\mathrm{f}}(\mathcal{D})$*.*

Proof. As $\alpha^+(\!|\cdot|\!)$ and a^- are !-free, and $(\cdot)^\circ$ behaves like the identity on !-free expressions. ◀

▶ **Proposition 34.** *Let $M \in \Lambda^{\bar{\tau}}_!$, $\vec{x} \supseteq \mathrm{FV}(M)$, $\alpha \in \mathcal{D}$ and $\vec{a} \in \mathcal{M}_{\mathrm{f}}(\mathcal{D})$. Then the following statements are equivalent:*

(i) $(\vec{a}, \alpha) \in [\![M]\!]_{\vec{x}}$,
(ii) $\alpha^+(\!|M\langle\vec{a}^-/\vec{x}\rangle|\!){\downarrow}$.

Proof. (i ⇒ ii) Suppose $(\vec{a}, \alpha) \in [\![M]\!]_{\vec{x}}$, then by Thm. 29 there is an $M' \in M^\circ$ such that $(\vec{a}, \alpha) \in [\![M']\!]_{\vec{x}}$. Applying Lemma 18 we know that $\alpha^+(\!|M'\langle\vec{a}^-/\vec{x}\rangle|\!) \twoheadrightarrow \varepsilon$. Now, since $\alpha^+(\!|M'\langle\vec{a}^-/\vec{x}\rangle|\!) \in (\alpha^+(\!|M\langle\vec{a}^-/\vec{x}\rangle|\!))^\circ$ (by Lemma 33), we can apply Corollary 32 and get $\alpha^+(\!|M\langle\vec{a}^-/\vec{x}\rangle|\!){\downarrow}$.

(ii ⇒ i) Suppose that $\alpha^+(\!|M\langle\vec{a}^-/\vec{x}\rangle|\!) \twoheadrightarrow \varepsilon + \mathbb{Q}$, for some $\mathbb{Q}$; then $[\![\alpha^+(\!|M\langle\vec{a}^-/\vec{x}\rangle|\!)]\!]_{\vec{x}} \neq \emptyset$. Hence, by Theorem 29, there is a closed test $R \in (\alpha^+(\!|M\langle\vec{a}^-/\vec{x}\rangle|\!))^\circ$ such that $[\![R]\!] \neq \emptyset$. By Lemma 33 $R = \alpha^+(\!|M'\langle\vec{a}^-/\vec{x}\rangle|\!)$ for some $M' \in M^\circ$ and since its interpretation is non-empty we have $R \twoheadrightarrow \varepsilon$. By applying Lemma 18 we get $(\vec{a}, \alpha) \in [\![M']\!]_{\vec{x}} \subseteq [\![M]\!]_{\vec{x}}$ (by Theorem 29). ◀

▶ **Theorem 35.** *$\mathcal{D}$ is inequationally fully abstract for the $\partial\lambda$-calculus with tests:*

$$[\![M]\!]_{\vec{x}} \subseteq [\![N]\!]_{\vec{x}} \Leftrightarrow M \sqsubseteq^!_{\mathcal{O}} N.$$

[2] If $\mathbb{A}' \subseteq \mathbb{A}^\circ$ and $\mathbb{A}' \to_h \mathbb{B}'$ then there exists $\mathbb{B}$ such that $\mathbb{A} \to_h \mathbb{B}$ and $\mathbb{B}' \subseteq \mathbb{B}^\circ$.

Proof. ($\Rightarrow$) Suppose that $[\![M]\!]_{\vec{x}} \subseteq [\![N]\!]_{\vec{x}}$ and there is a test-context $C(\!|\cdot|\!)$ (closing M, N) such that $C(\!|M|\!)\!\downarrow$. Since $C(\!|M|\!) \twoheadrightarrow \varepsilon + \mathbb{Q}$, for some $\mathbb{Q}$, we have $[\![C(\!|M|\!)]\!] \neq \emptyset$. Thus, by monotonicity of the interpretation we get $[\![C(\!|M|\!)]\!] \subseteq [\![C(\!|N|\!)]\!] = [\![(C(\!|N|\!))^\circ]\!] \neq \emptyset$. By Corollary 6 there is $R \in (C(\!|N|\!))^\circ$ such that $R \twoheadrightarrow \varepsilon$ and we conclude that $C(\!|N|\!)\!\downarrow$ by applying Proposition 34.

($\Leftarrow$) Suppose by contradiction that $M \sqsubseteq^!_{\mathcal{O}} N$, but there is an $(\vec{a}, \alpha) \in [\![M]\!]_{\vec{x}} - [\![N]\!]_{\vec{x}}$. By Prop. 34 $\alpha^+(\!|M\langle\vec{a}^-/\vec{x}\rangle|\!)\!\downarrow$ and since $M \sqsubseteq^!_{\mathcal{O}} N$ we have $\alpha^+(\!|N\langle\vec{a}^-/\vec{x}\rangle|\!)\!\downarrow$. Again, by Prop. 34 $(\vec{a}, \alpha) \in [\![N]\!]_{\vec{x}}$. Contradiction. ◀

Further Work. We proved that $\mathcal{D}$ is a fully abstract model of the $\partial\lambda$-calculus and of the $\partial_0\lambda$-calculus with tests. We strongly conjecture that it also (in)equationally fully abstract for the corresponding calculi *without tests*. A possible approach to obtain these results might be to define a "test-expansion" translating every test-context $C(\!|\cdot|\!)$ sending $M \in \Lambda_!^{\bar{\tau}}$ to $\varepsilon + \mathbb{R}$ into a term-context $C'(\!|\cdot|\!)$ sending M to $\mathbf{I} + \mathbb{N}$. This generalization is non trivial and is kept for future work. Another open problem is to find a fully abstract model of these calculi where $+$ is treated as *must* non-determinism (a sum converges if all its summands converge).

References

1 H. P. Barendregt. *The Lambda Calculus: Its Syntax and Semantics.* North-Holland, Amsterdam, 1984.

2 G. Boudol. The lambda-calculus with multiplicities (abstract). In Eike Best, editor, *CONCUR*, volume 715 of *Lecture Notes in Computer Science*, pages 1–6. Springer, 1993.

3 G. Boudol, P.-L. Curien, and C. Lavatelli. A semantics for lambda calculi with resources. *Math. Struct. in Comp. Sci.*, 9(4):437–482, 1999.

4 A. Bucciarelli, T. Ehrhard, and G. Manzonetto. Not enough points is enough. In *CSL'07*, volume 4646 of *LNCS*, pages 298–312. Springer, 2007.

5 T. Ehrhard and O. Laurent. Interpreting a finitary pi-calculus in differential interaction nets. *Inf. Comput.*, 208(6):606–633, 2010.

6 T. Ehrhard and L. Regnier. The differential lambda-calculus. *Theor. Comput. Sci.*, 309(1-3):1–41, 2003.

7 T. Ehrhard and L. Regnier. Böhm trees, Krivine's machine and the Taylor expansion of lambda-terms. In *CiE*, volume 3988 of *LNCS*, pages 186–197. Springer, 2006.

8 T. Ehrhard and L. Regnier. Uniformity and the Taylor expansion of ordinary lambda-terms. *Theor. Comput. Sci.*, 403(2-3):347–372, 2008.

9 J.-Y. Girard. From foundations to ludics. *Bulletin of Symbolic Logic*, 9(2):131–168, 2003.

10 G. Manzonetto. What is a categorical model of the differential and the resource lambda calculi? June 2010. Submitted to Mathematical Structures in Computer Science.

11 R. Milner. *A Calculus of Communicating Systems.* Springer-Verlag, 1982.

12 R. Milner. The polyadic π-calculus: a tutorial. *Logic and algebra of specification*, pages 203–246, 1993.

13 M. Pagani and S. Ronchi Della Rocca. Solvability in resource lambda-calculus. In *Proc. of FOSSACS'10*, volume 6014 of *LNCS*, pages 358–373. Springer, 2010.

14 M. Pagani and P. Tranquilli. Parallel reduction in resource lambda-calculus. In *APLAS'09*, volume 5904 of *LNCS*, pages 226–242. Springer, 2009.

15 P. Tranquilli. Intuitionistic differential nets and lambda-calculus. *Theor. Comp. Sci.* To appear.

16 L. Vaux. The differential $\lambda\mu$-calculus. *Theor. Comput. Sci.*, 379(1-2):166–209, 2007.