



HAL
open science

The weak algebraic λ -calculus

Zacharie Moughanim, Lionel Vaux Auclair

► **To cite this version:**

Zacharie Moughanim, Lionel Vaux Auclair. The weak algebraic λ -calculus. 36es Journées Francophones des Langages Applicatifs (JFLA 2025), Jan 2025, Roiffé, France. hal-04859524

HAL Id: hal-04859524

<https://hal.science/hal-04859524v1>

Submitted on 30 Dec 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Copyright

The weak algebraic λ -calculus

Zacharie Moughanim¹ and Lionel Vaux Auclair²

¹ENS Rennes, France

²Aix-Marseille Université, CNRS, I2M, France

We introduce a variant of the algebraic λ -calculus, which extends the ordinary λ -calculus with the possibility of forming linear combinations of terms, with coefficients taken in a fixed semiring. The only difference from the original version is that we drop the identity $0M = \mathbf{0}$, i.e., multiplication by the scalar 0 does not yield the empty sum in general: terms thus form a weak module instead of a module. We then study how this minor relaxing of axioms yields a simpler and more robust rewriting theory. In particular, we show that the obtained reduction is still confluent; however, contrasting with the original version, the induced equivalence is conservative over β -equivalence, without any assumption on the semiring. We moreover characterize normalization properties *via* a system of intersection type systems, and provide a normalization strategy, again without assumption on scalars. Finally, we show that the Taylor expansion of algebraic λ -terms can be adapted to this setting.

1 Introduction

Quantitative semantics and linear combinations of λ -terms. In 2003, Ehrhard and Regnier introduced the differential λ -calculus [ER03], an extension of the λ -calculus where, in addition to ordinary functional application, one can form the derivative of a term in the direction of another term. The reduction rules give a syntactic account of *quantitative semantics*, an umbrella term for particular denotational models of the typed λ -calculus, and of linear logic, in which the interpretation of types is based on a notion of topological vector space, and λ -terms are interpreted as particular analytic maps, in a more or less standard sense—the idea predates the invention of linear logic [Gir88], and the instance closest to standard notions of is due to Ehrhard [Ehr05].

The syntax of the calculus also involves finite sums of terms, or even linear combinations of terms, with coefficients taken in a fixed field or, more generally, a semiring with units (AKA rig) \mathbb{R} : sums are needed because they are generated¹ by differentiation itself; and it is natural to allow for linear combinations, precisely because the calculus mimics quantitative semantics. Terms are moreover subject to identities reflecting the pointwise definition of sums of functions: the application $(M + N)P$ of the sum of terms $M + N$ to the argument P is identified with the sum of applications $MP + NP$; and the nullary version of this identity states that $\mathbf{0}P = \mathbf{0}$, where $\mathbf{0}$ is the empty sum of terms. In presence of linear combinations, we also enforce $(aM)P = a(MP)$ for any scalar $a \in \mathbb{R}$.

¹Think, e.g., of the product rule, which yields a sum; or of the derivative of a constant, which is zero. On the other hand, arbitrary finite sums can be seen as a syntactic artefact of computational non-determinism. It is in fact possible to restrict the shape of sums, and to reconcile differentiation with determinism, but this requires a much more sophisticated analysis: this is the object of *coherent differentiation*, developed by Ehrhard in recent years [Ehr23b, Ehr23a, EW23]. This framework is already very satisfactory from a semantic point of view, but the problem of designing a well-behaved rewriting theory for it is still open.

These identities are all straightforwardly validated by the semantics, which ensures that the typed version of the calculus is consistent. Still, a number of peculiarities must be taken into account as soon as one is interested in operational properties. First, if one chooses to quotient terms up to the above identities, should we consider that $\mathbf{0}M = \mathbf{0}$ is a normal form even when M is β -reducible? And, in case we have coefficients, the same question arises for $0M = \mathbf{0}$ (multiplication by the scalar zero being identified with the empty sum). Worse: considering $M = \frac{1}{2}M + \frac{1}{2}M$, should we allow to reduce only half of M ?

In the absence of typing (or if one adds fixed-points combinators *à la* PCF), the equational theory becomes even wilder. Given any term M , one can construct a term ∞_M which reduces to, or is equivalent to, $M + \infty_M$: consider for instance $Y\lambda x.(M + x)$ where Y is any fixed-point combinator. Intuitively, ∞_M produces an infinite amount of M . Then, in case -1 is allowed as a coefficient (with $-1 + 1 = 0$ in \mathbf{R}), consistency is broken: $\mathbf{0} = \infty_M + (-1)\infty_M$ is β -equivalent to $M + \infty_M + (-1)\infty_M = M$ for any term M !

It is evident that all the subtle (or catastrophic) phenomena we have just outlined arise from the treatment of sums and scalars only: differentiation plays no particular role here. Sums or linear combinations of terms have moreover been used to represent various kinds of superpositions of computational behaviours: e.g., non-deterministic choice (this was the intuition behind Girard's original quantitative semantics [Gir88]), finite and discrete probability distributions [Lev19] or quantum superposition [vT04, AD17]. Higher-order rewriting in presence of linear combinations of terms is thus an interesting subject of study in itself, and it is useful to develop a general theory of it, to investigate the possible issues that can arise in the abstract, and to design strategies to circumvent them.

The algebraic λ -calculus. The second author has introduced the *algebraic λ -calculus* precisely as an extension of the pure, untyped λ -calculus with finite linear combinations of terms, up to the previously discussed identities, plus similar identities ensuring the linearity of λ -abstraction [Vau09].² One can then consider a notion of *canonical form*, obtained by pulling sums outside of syntactic constructs: this can be done by orienting linearity identities, e.g., $(M + N)P \rightsquigarrow MP + NP$. As in Ehrhard and Regnier's differential λ -calculus, one then distinguishes a class of so-called *simple terms*, which are those that can be written canonically without sum at top level (although sums can still appear in arguments of applications): these form a basis of the \mathbf{R} -module of terms.

On simple terms, β -reduction is defined as usual: $(\lambda x.M)N$ reduces to the substitution $M[N/x]$ (which is not simple in general: it suffices to consider the case of $M = x$ and N an arbitrary term). To ensure the existence of normal forms despite the identities $\mathbf{0}M = \mathbf{0} = 0M$, the reduction is extended from simple terms to terms as follows: M reduces to N whenever we can write $M = as + T$ and $N = aS' + T$, with s a simple term reducing to S' and $a \neq 0$. With this definition, reduction is shown to be confluent by standard means, adapting the Tait–Martin-Löf technique.

Still, if \mathbf{R} contains rational coefficients (or more generally $a, b \neq 0$ such that $a + b = 1$) this leaves the possibility of reducing only half of a simple term M in $\frac{1}{2}M + \frac{1}{2}M$. One could think of imposing that no amount of s remains in T (formally: s is not in the support set of T) to allow for the reduction of $as + T$ to $aS' + T$, but this hampers the contextuality of reduction, which is essential in the proof of confluence: in the presence of fixed-points, this actually breaks confluence [Vau09].

In case the semiring \mathbf{R} admits an opposite for 1, the above argument stands and β -equivalence becomes trivial. But if \mathbf{R} is *positive* (i.e., $a + b = 0$ only if $a = b = 0$) then normal forms do exist and confluence ensures the consistency of β -equivalence. Moreover, under the

²This reflects an implicit bias towards call-by-name evaluation, which simplifies the study of normalization: we will adopt the same bias in the following. It is also possible to renounce the linearity of abstraction, and consider a bilinear application instead, as was done by Arrighi and Dowek [AD17]: this is more in line with call-by-value evaluation. Those approaches are somehow dual to each other [ADP⁺14], but the problems and solutions we discuss in the rest of the paper are essentially the same, whether one adheres to any of the two disciplines, or to none (i.e., if one only admits the left-linearity of application).

same assumption, Kerinec and the second author have established that this extension of β -equivalence is conservative: two ordinary λ -terms are equivalent in the algebraic λ -calculus iff they are β -equal in the ordinary λ -calculus [KV23].

Dealing with negative coefficients. In order to lift the positivity assumption, while retaining good operational properties (at least confluence and consistency), it is necessary to tame the interplay between negative coefficients and fixed-points.

One possibility is to control how identities on linear combinations are applied by treating them as rewriting steps: Arrighi and Dowek restrict those algebraic rewriting steps and apply them on closed normal forms only [AD17], thus ensuring confluence. The price to pay is that the study of the rewriting system is much more complicated, and that we obtain a module of terms only by restricting to closed normal forms.

Typing is another possible course: one can consider a system of simple types, or even variants of system F [Ehr10, ADV17], to ensure strong normalizability, and thus forbid fixed-points; then one gets consistency by proving local confluence, or *via* denotational arguments. One must be careful, though, and stick to a Church-style typing; otherwise, the identity $M = M + N - N$ would allow to rewrite a typeable term to an untypeable one.

It is also possible to “freeze” computations on coefficients temporarily. The semiring of formal polynomials (without coefficients, or with positive integer coefficients only) is not only positive, but also *finitely splitting* (meaning that each element can be written only in finitely many ways as a sum of non zero elements): this ensures that, e.g., typed terms are strongly normalizable [Vau09]. Then one can consider the following weak normalization scheme: given a term M , fix a polynomial for each scalar coefficient of M , together with a global valuation of indeterminates sending each polynomial to the associated scalar; then reduce the version with polynomial coefficients to a normal form (if any); and finally replace indeterminates with their values. It is easy to check that the (canonical form of the) obtained term has no redex, although it might not be normal, in presence of negative coefficients: it is called a *pre-normal form* of M . Alberti showed that, in case M admits a normalizing polynomial version, the obtained pre-normal form is unique [Alb14].

As an alternative approach, the Taylor expansion of λ -terms provides an untyped form of quantitative semantics. The Taylor expansion of a λ -term M is obtained by applying the usual Taylor formula to each application in M recursively, differentiation being performed in the differential λ -calculus. Taylor expansion thus yields a (possibly infinitely supported) vector of resource terms: the latter are the terms of the *resource calculus*, which is the fragment of the differential λ -calculus where the only form of application is iterated differentiation at zero. The dynamics of this fragment is very regular, with each term having a unique normal form, without typing condition. Ehrhard and Regnier’s seminal results [ER06, ER08] ensure that Taylor expansion commutes with normalization: formally the normal form of the Taylor expansion of a term is the Taylor expansion of its Böhm tree. The second author has shown that, although the Taylor expansion of algebraic λ -terms is not normalizable in general, the class of terms whose Taylor expansion does normalize is well behaved: it contains both untyped ordinary λ -terms and typed algebraic λ -terms; and if one restricts β -equivalence to that class, it is confluent and conservative w.r.t. Böhm-tree equivalence (in particular it is consistent).

The weak algebraic λ -calculus. Compared to the lines of work we have just outlined, the solution we consider in the present paper is surprisingly simple: we just drop one identity on linear combinations: $0M = \mathbf{0}$. Without this equation, terms form a weak module instead of a module: we call the corresponding system the *weak algebraic λ -calculus*. Valiron has already observed [Val13] that removing this single axiom avoids the inconsistency induced by the interplay between fixed-points and negative coefficients: he used that trick to study the categorical semantics of a typed version of the calculus, with a typed fixed-point operator.

Here we study how this very same trick can be leveraged in an untyped setting, and how it allows to vastly simplify the rewriting theory.

In Section 2, we introduce the syntax of the calculus: we follow the exact same construction as for the algebraic λ -calculus, save for the omitted identity. In particular we make explicit the notion of canonical form induced by the revised treatment of linear combinations. In Section 3, we study the dynamics of the calculus. Again, the reduction is similar to that of the algebraic λ -calculus, except that we do not need to require that $a \neq 0$ in a reduction from $as + T$ to $aS' + T$, as this step is no longer “silent”. We establish the confluence of reduction, as well as the conservativity of the induced equivalence w.r.t. β -equivalence in the ordinary λ -calculus.

We study normalization properties in Section 4. We distinguish between may- and must-normalizability and characterize each property *via* an intersection type system: a variant of $D\Omega$ [Kri93] for must-normalizability, inspired by the system introduced by Pagani, Tasson and the second author [PTV16] for the *non-deterministic* λ -calculus; and a non-idempotent intersection type system [BKV17] for may-normalizability, inspired by de Carvalho’s system R [dC18], which accounts for the multiset-based relational semantics (a kind of quantitative semantics with boolean coefficients).

Finally, we refine to our weak setting the argument showing that Taylor expansion is compatible with reduction in the algebraic λ -calculus. For this, we consider a weak notion of resource vectors (distinguishing between resource terms that occur with a 0 coefficient, and those that do not occur in the support set) and we adapt Taylor expansion accordingly. We obtain a result of commutation between (must-)normalization and Taylor expansion, even in the presence of negative coefficients. We also deduce a conservativity result, which is limited to normalizable ordinary λ -terms. Our results raise a number of questions for future investigations, that we discuss in the concluding Section 6.

2 Weak algebraic λ -terms

We begin by defining the terms of the weak algebraic λ -calculus: we define a grammar of *raw terms* and then define an equivalence relation \triangleq on these terms. Raw terms up to this equivalence are then the terms of the weak algebraic λ -calculus. The resulting set is endowed with a weak R -module structure.

Let us first recall that a *semiring* is a set R equipped with two monoid structures $(R, +, 0)$ and $(R, \cdot, 1)$, such that $+$ is commutative, \cdot is distributive over $+$ and 0 is absorbing for \cdot . We generally denote multiplication by juxtaposition: $ab = a \cdot b$. We say R is *commutative* when \cdot is. And we say it is *positive* when it is zero-sum-free: for all $a, b \in R$, $a + b = 0$ implies $a = b = 0$.

A *weak R -module* is then a commutative monoid $(M, +, \mathbf{0})$ equipped with a scalar multiplication $R \times M \rightarrow M$, again denoted by juxtaposition, and such that, for all $a, b \in R$, $s, t \in M$:

$$a(s + t) = as + at \quad (a + b)s = as + bs \quad (ab)s = a(bs) \quad 1s = s .$$

An *R -module* is a weak R -module such that, moreover, $0s = \mathbf{0}$.

Definition 1 (Raw terms). We write L_R for the set of *raw terms*, defined by the grammar:

$$L_R \ni M, N, \dots ::= x \mid \lambda x.M \mid MN \mid M + N \mid aM \mid \mathbf{0}$$

and considered up to α -equivalence only.

We say a relation r on L_R is *contextual* if it is reflexive and, assuming $M r N$ and $P r Q$:

$$aM r aN \quad M + P r N + Q \quad \mathbf{0} r \mathbf{0} \quad \lambda x.M r \lambda x.N \quad MP r NQ .$$

Definition 2. We call *weak algebraic equality* the least contextual relation $\stackrel{\Delta}{\equiv}$ satisfying:

$$\begin{array}{ll}
\mathbf{0} + M \stackrel{\Delta}{\equiv} M & a\mathbf{0} \stackrel{\Delta}{\equiv} \mathbf{0} \\
M + N \stackrel{\Delta}{\equiv} N + M & \lambda x.\mathbf{0} \stackrel{\Delta}{\equiv} \mathbf{0} \\
(M + N) + P \stackrel{\Delta}{\equiv} M + (N + P) & \lambda x.aM \stackrel{\Delta}{\equiv} a\lambda x.M \\
a(M + N) \stackrel{\Delta}{\equiv} aM + aN & \lambda x.(M + N) \stackrel{\Delta}{\equiv} \lambda x.M + \lambda x.N \\
aM + bM \stackrel{\Delta}{\equiv} (a + b)M & \mathbf{0}M \stackrel{\Delta}{\equiv} \mathbf{0} \\
a(bM) \stackrel{\Delta}{\equiv} (ab)M & (aM)N \stackrel{\Delta}{\equiv} a(MN) \\
1M \stackrel{\Delta}{\equiv} M & (M + N)L \stackrel{\Delta}{\equiv} ML + NL.
\end{array}$$

Beware that, in the equation $\mathbf{0}M \stackrel{\Delta}{\equiv} \mathbf{0}$, the LHS is the application of $\mathbf{0}$ to M , not to be confused with the scalar multiplication $0M$. If $M \in \mathbf{L}_R$, we denote by \underline{M} its $\stackrel{\Delta}{\equiv}$ -class. We naturally extend this notation to sets of terms: if $E \subseteq \mathbf{L}_R$, we write $\underline{E} = \{\underline{M} \mid M \in E\}$.

We call *weak algebraic λ -terms* (or just *terms*, for short) the $\stackrel{\Delta}{\equiv}$ -classes of raw terms. We will often use syntactic operations on terms instead of raw terms : one can easily check that all syntactic constructors, as well as the usual capture avoiding substitution, are well defined on terms.

Now, as for the algebraic λ -calculus [Vau09], we separate commutativity and associativity (AC) from the other equalities to define a canonical writing of a $\stackrel{\Delta}{\equiv}$ -class as a normal form in a rewriting system given by the above equalities, oriented from left to right. We will later describe a big-step evaluator to compute such a normal form.

Definition 3 (Permutative terms). We denote by \equiv the least contextual relation such that $M + N \equiv N + M$ and $(M + N) + P \equiv M + (N + P)$. We call *permutative terms* the \equiv -classes, and denote by Λ_R the quotient set \mathbf{L}_R/\equiv .

So permutative terms are raw terms up to AC. We now describe canonical representatives of $\stackrel{\Delta}{\equiv}$ -classes among them.

Definition 4 (Base terms, canonical terms). We define the set of *base terms* $\mathbf{B}_R \subseteq \Lambda_R$ and the set of *canonical terms* $\mathbf{C}_R \subseteq \Lambda_R$ by mutual induction as follows:

$$\mathbf{B}_R \ni s, t, \dots ::= x \mid \lambda x.s \mid sT \quad \text{and} \quad \mathbf{C}_R \ni S, T, \dots ::= \sum a_i s_i$$

where the s_i 's are pairwise distinct base terms and $a_1, \dots, a_n \in \mathbf{R}$.

Note that we do not require the a_i 's to be non-zero.

Definition 5 (Canonization). Let $M = \sum_{i=1}^n a_i s_i \in \Lambda_R$. Let $I_{(s)}^M = \{i \in \{1, \dots, n\} \mid s_i = s\}$, we set $M_{(s)} = \sum_{i \in I_{(s)}^M} a_i$. Then, let $(t_i)_{i \in \{1, \dots, p\}}$ be a tuple of pairwise distinct elements such that $\{t_1, \dots, t_p\} = \{s_1, \dots, s_n\}$. Finally, we define $\text{cansum}(M) = \sum_{i=1}^p M_{(t_i)} t_i$.

We define $\text{can} : \Lambda_R \rightarrow \mathbf{C}_R$ by induction on permutative terms. Let $\text{can}(M) = \sum_{i=1}^n a_i s_i$ and $\text{can}(N) = \sum_{i=n+1}^{n+p} a_i s_i = T$:

$$\begin{array}{lll}
\text{can}(x) = 1x & \text{can}(\lambda x.M) = \sum_{i=1}^n a_i \lambda x.s_i & \text{can}(MN) = \sum_{i=1}^n a_i (s_i T) \\
\text{can}(\mathbf{0}) = \mathbf{0} & \text{can}(aM) = \sum_{i=1}^n (aa_i) s_i & \text{can}(M + N) = \text{cansum} \left(\sum_{i=1}^{n+p} a_i s_i \right)
\end{array}$$

Again, note that, in the definition of $\text{cansum}(M)$, we do not remove those summands s_i such that $M_{s_i} = 0$. Like for the algebraic λ -calculus, the following characterization of follows easily from the definitions:

Proposition 1. For all $M, N \in \Lambda_R$, we have $\text{can}(\text{can}(M)) = \text{can}(M)$. Moreover, $\text{can}(M) = \text{can}(N)$ if and only if $M \stackrel{\Delta}{=} N$.

In other words, each term M admits a canonical $\stackrel{\Delta}{=}$ -representative $\text{can}(M)$, which is unique up to \equiv . We then call *simple terms* the $\stackrel{\Delta}{=}$ -classes of base terms: it follows that each term is a linear combination (possibly using coefficient 0) of simple terms. In the algebraic λ -calculus, the support of M is defined as the set of simple terms with a nonzero coefficient in M , but the definition must of course be different here. It is actually simpler, because it can be described straightforwardly by induction:

Definition 6 (Support of a term). We define $\text{Supp} : \mathbf{L}_R \rightarrow \mathcal{P}(\mathbf{B}_R)$ by induction on M :

$$\begin{aligned} \text{Supp}(x) &= \{x\} \\ \text{Supp}(\lambda x.M) &= \{\lambda x.s \mid s \in \text{Supp}(M)\} \\ \text{Supp}(MN) &= \{sN \mid s \in \text{Supp}(M)\} \\ \text{Supp}(M + N) &= \text{Supp}(M) \cup \text{Supp}(N) \\ \text{Supp}(aM) &= \text{Supp}(M). \end{aligned}$$

It is a routine exercise to check that the support is stable under $\stackrel{\Delta}{=}$ so that we can set $\text{Supp}(\underline{M}) = \text{Supp}(M)$ for any $M \in \Lambda_R$. For any set $\mathcal{X} \subseteq \mathbf{L}_R$, we write $\mathbf{R}\langle \mathcal{X} \rangle$ for the weak \mathbf{R} -submodule of the module of terms, generated by \mathcal{X} : in other words, $\mathbf{R}\langle \mathcal{X} \rangle = \{\underline{T} \in \mathbf{L}_R \mid \text{Supp}(T) \subseteq \mathcal{X}\}$. A straightforward consequence of Proposition 1 is that $\mathbf{L}_R = \mathbf{C}_R = \mathbf{R}\langle \mathbf{B}_R \rangle$: simple terms generate the weak module of terms.

3 Reductions

In this section, we introduce the extension of β -reduction in the weak algebraic λ -calculus. A main difference with the algebraic λ -calculus arises here: we allow terms to be reduced under a zero coefficient. We take advantage of the fact that, in general, $\underline{0M} \neq \underline{0}$, so that this does not make the reduction trivially reflexive: we will discuss normalization properties in the next section. Here we prove confluence following the usual Tait–Martin–Löf technique. We moreover establish a conservativity result: for any reduction sequence between two ordinary λ -terms in our calculus, there is β -reduction sequence between the same terms.

3.1 Definitions

We will in fact consider three notions of reduction: *one-step reduction* $\stackrel{\square}{\rightarrow}$; *parallel reduction* $\stackrel{\boxplus}{\rightarrow}$; and *canonical reduction* $\stackrel{\widehat{\rightarrow}}{\rightarrow}$. Parallel reduction will be useful to establish the confluence of one-step reduction, while canonical reduction will serve as an auxiliary notion when we discuss (may)-normalizability. We introduce each notion of reduction by defining an increasing sequence of relations for $k \in \mathbb{N}$. And each one will involve a particular way to extend reduction from simple terms to terms:

Definition 7. Let $r \subseteq \mathbf{B}_R \times \mathbf{C}_R$, we define three extensions of r from terms to terms.

- If $\underline{s} r \underline{S}, a \in \mathbf{R}, T \in \mathbf{C}_R$, then $\underline{as + T} \stackrel{\square}{r} \underline{aS + T}$;
- If $\underline{s}_i r \underline{T}_i$ for $i \in \{1, \dots, n\}$ and $(a_i) \in \mathbf{R}^n$, then $\sum a_i s_i \bar{r} \sum a_i T_i$;
- If $\underline{s} r \underline{S}, a \in \mathbf{R}, T \in \mathbf{C}_R$ and $\underline{s} \notin \text{Supp}(T)$, then $\underline{as + T} \hat{r} \underline{aS + T}$.

Definition 8 (One-step reduction). Let $\rightarrow_0 = \emptyset$. We set:

- $\underline{\lambda x.s} \rightarrow_{k+1} \underline{\lambda x.S}$ if $\underline{s} \rightarrow_k \underline{S}$;
- $\underline{sT} \rightarrow_{k+1} \underline{S'T'}$ if $\underline{s} \rightarrow_k \underline{S'}$;
- $\underline{sT} \rightarrow_{k+1} \underline{sT'}$ if $\underline{T} \stackrel{\square}{\rightarrow}_k \underline{T'}$;
- $\underline{(\lambda x.s)T} \rightarrow_{k+1} \underline{s[T/x]}$.

Then let $\rightarrow = \bigcup_{k \in \mathbb{N}} \rightarrow_k$: *one-step reduction* is then $\overset{\square}{\rightarrow} = \bigcup_{k \in \mathbb{N}} \overset{\square}{\rightarrow}_k$.

Definition 9 (Parallel reduction). Let $\overset{\square}{\rightarrow}_0 = \{(\underline{s}, \underline{s}) \mid s \in \mathbf{B}_R\}$:

- $\underline{\lambda x.s} \overset{\square}{\rightarrow}_{k+1} \underline{\lambda x.S}$ if $\underline{s} \overset{\square}{\rightarrow}_k \underline{S}$;
- $\underline{sT} \overset{\square}{\rightarrow}_{k+1} \underline{S'T'}$ if $\underline{s} \overset{\square}{\rightarrow}_k \underline{S'}$ and $\underline{T} \overset{\square}{\rightarrow}_k \underline{T'}$;
- $\underline{(\lambda x.s)T} \overset{\square}{\rightarrow}_{k+1} \underline{S'[T'/x]}$ if $\underline{s} \overset{\square}{\rightarrow}_k \underline{S'}$ and $\underline{T} \overset{\square}{\rightarrow}_k \underline{T'}$.

Then let $\overset{\square}{\rightarrow} = \bigcup_{k \in \mathbb{N}} \overset{\square}{\rightarrow}_k$: *parallel reduction* is then $\overline{\overset{\square}{\rightarrow}} = \bigcup_{k \in \mathbb{N}} \overline{\overset{\square}{\rightarrow}_k}$.

It should be clear that $\overline{\overset{\square}{\rightarrow}}$ is contextual. The essential property of parallel reduction is its compatibility with substitution:

Lemma 1. *Let $\underline{M}, \underline{M'}, \underline{N}, \underline{N'} \in \underline{\mathbf{C}}_R$. If $\underline{M} \overline{\overset{\square}{\rightarrow}} \underline{M'}$ and $\underline{N} \overline{\overset{\square}{\rightarrow}} \underline{N'}$, then $\underline{M}[N/x] \overline{\overset{\square}{\rightarrow}} \underline{M'}[N'/x]$.*

Definition 10 (Canonical reduction). Let $\rightarrow_0 = \emptyset$:

- $\underline{\lambda x.s} \rightarrow_{k+1} \underline{\lambda x.S}$ if $\underline{s} \rightarrow_k \underline{S}$;
- $\underline{sT} \rightarrow_{k+1} \underline{S'T}$ if $\underline{s} \rightarrow_k \underline{S'}$;
- $\underline{sT} \rightarrow_{k+1} \underline{sT'}$ if $\underline{T} \widehat{\rightarrow}_k \underline{T'}$;
- $\underline{(\lambda x.s)T} \rightarrow_{k+1} \underline{s[T/x]}$.

Then, let $\rightarrow = \bigcup_{k \in \mathbb{N}} \rightarrow_k$, the canonical reduction is then $\widehat{\rightarrow} = \bigcup_{k \in \mathbb{N}} \widehat{\rightarrow}_k$.

Canonical reduction differs from one-step reduction only in the more restrictive way reduction is extended to sums: when reducing a simple term of the support, one must reduce it in full, and not leave some amount of it (possibly with weight 0) in the rest of the sum. This notion is intuitively better for normalization purposes, but it is not locally confluent: the counter-example given in the ordinary case [Vau09] is also valid in the weak case.

3.2 Confluence

We now prove confluence of $\overset{\square}{\rightarrow}$, using the usual Tait-Martin-Löf technique.

Lemma 2. *The following inclusions hold: $\overset{\square}{\rightarrow} \subset \overline{\overset{\square}{\rightarrow}} \subset \overset{\square}{\rightarrow}^*$.*

Definition 11 (Full parallel reduct). We define $F : \underline{\mathbf{C}}_R \rightarrow \underline{\mathbf{C}}_R$ by induction:

$$\begin{aligned} F(\underline{x}) &= \underline{x} & F(\underline{\lambda x.s}) &= \underline{\lambda x.F(s)} & F(\underline{(\lambda x.s)T}) &= F(\underline{s})[F(\underline{T})/x] \\ F(\underline{sT}) &= F(\underline{s})F(\underline{T}) \text{ if } \underline{s} \text{ is not an abstraction} & F\left(\sum a_i s_i\right) &= \sum a_i F(s_i) . \end{aligned}$$

Lemma 3. *We have $\underline{M} \overline{\overset{\square}{\rightarrow}} F(\underline{M})$ for any $\underline{M} \in \underline{\Lambda}_R$. If $\underline{M} \overline{\overset{\square}{\rightarrow}} \underline{N}$ then $F(\underline{M}) \overline{\overset{\square}{\rightarrow}} F(\underline{N})$. And if $\underline{M} \overset{\square}{\rightarrow}^k \underline{N}$ then $\underline{N} \overline{\overset{\square}{\rightarrow}}^* F^k(\underline{M})$.*

Theorem 1. *$\overline{\overset{\square}{\rightarrow}}$ is strongly confluent, hence $\overset{\square}{\rightarrow}$ enjoys the Church-Rosser property.*

3.3 Conservativity

In this section, we adapt a proof of conservativity for the algebraic λ -calculus [KV23] to our setting. We denote by Λ the set of pure λ -terms, by \rightarrow_Λ its reduction and by \simeq_R (resp. \simeq_Λ) the symmetric, reflexive and transitive closure of $\overset{\square}{\rightarrow}$ (resp. \rightarrow_Λ). We define two mutually inductive relations $\vdash \subseteq \Lambda \times \underline{\mathbf{B}}_R$ and $\Vdash \subseteq \Lambda \times \underline{\mathbf{C}}_R$. The goal is to prove the relations defined in Figure 1 are, in a sense, compatible with \rightarrow_Λ and $\overset{\square}{\rightarrow}$ in such a way that we can transform a \rightarrow_Λ -reduction sequence into a $\overset{\square}{\rightarrow}$ -reduction sequence. The main results of this section are Theorem 2 and Corollary 1. We omit most proofs which are straightforwardly adapted from the original ones.

Lemma 4. *Let $\underline{M} \in \Lambda$, $\underline{N} \in \underline{\mathbf{C}}_R$, $\underline{M} \Vdash \underline{N}$ if and only if for every $\underline{s} \in \text{Supp}(\underline{N})$, $\underline{M} \vdash \underline{s}$.*

$$\begin{array}{c}
\frac{M \rightarrow_{\Lambda}^* x}{M \vdash x} \text{ (v)} \quad \frac{M \rightarrow_{\Lambda}^* \lambda x.N \quad N \vdash \underline{N}'}{M \vdash \underline{\lambda x.N'}} \text{ (\lambda)} \\
\frac{M \rightarrow_{\Lambda}^* NP \quad N \vdash \underline{N}' \quad P \Vdash \underline{P}'}{M \vdash \underline{N'P'}} \text{ (@)} \quad \frac{M \vdash \underline{s_1} \quad \dots \quad M \vdash \underline{s_n}}{M \Vdash \underline{\sum a_i s_i}} \text{ (+)}
\end{array}$$

Figure 1. Rules for the mashup relation

$$\begin{array}{c}
\frac{M \vdash \underline{N}}{M \Vdash \underline{N}} \text{ (s)} \quad \frac{M \rightarrow_{\Lambda}^* \lambda x.N \quad N \Vdash \underline{N}'}{M \Vdash \underline{\lambda x.N'}} \text{ (\lambda')} \\
\frac{M \rightarrow_{\Lambda}^* NP \quad N \Vdash \underline{N}' \quad P \Vdash \underline{P}'}{M \Vdash \underline{N'P'}} \text{ (@')} \quad \frac{M \Vdash \underline{S_1} \quad \dots \quad M \Vdash \underline{S_n}}{M \Vdash \underline{\sum a_i S_i}} \text{ (+')}
\end{array}$$

Figure 2. Admissible rules for the mashup relation

Lemma 5. *The rules of Figure 2 are admissible.*

Lemma 6 (Reflexivity of \vdash and \Vdash). *For all $M \in \Lambda$, $M \vdash \underline{M}$ and $M \Vdash \underline{M}$.*

Lemma 7 (Conservativity of \Vdash). *Let $M, M' \in \Lambda$. If $M \Vdash \underline{M}'$, then $M \rightarrow_{\Lambda}^* M'$.*

Lemma 8 (Left compatibility). *Let $M, M' \in \Lambda$ and $\underline{N} \in \underline{\mathcal{C}}_{\mathcal{R}}$. If $M \rightarrow_{\Lambda}^* M'$ and $M' \vdash \underline{N}$ (resp. $M' \Vdash \underline{N}$), then $M \vdash \underline{N}$ (resp. $M \Vdash \underline{N}$).*

Lemma 9. *Let $M, P \in \Lambda$, $\underline{N}, \underline{Q} \in \underline{\mathcal{C}}_{\mathcal{R}}$. If $M \Vdash \underline{N}$ and $P \Vdash \underline{Q}$, then $M[P/x] \Vdash \underline{N[Q/x]}$.*

Lemma 10 (Right compatibility). *Let $M \in \Lambda$, $\underline{M}', \underline{N} \in \underline{\mathcal{C}}_{\mathcal{R}}$. If $M \Vdash \underline{M}'$ and $\underline{M}' \xrightarrow{\square} \underline{N}$, then $M \Vdash \underline{N}$.*

Proof. The proof is essentially the same as the original one: it differs only in the treatment of sums. We thus detail the case of $\underline{as} + \underline{T} \xrightarrow{\square} \underline{aS'} + \underline{T}$ only. Let $T = \sum_{i=1}^n a_i s_i$, since $\text{Supp}(M') = \{s, s_1, \dots, s_n\}$ by definition 6, we get from lemma 4 that $M \vdash s$ and $M \vdash s_i$ for each s_i . By induction hypothesis, $M \Vdash \underline{S'}$ and we conclude by applying the rule (+'). \square

Remark 1. The sum case of previous proof is the only argument where we need the positivity of the semiring in the algebraic λ -calculus [KV23]. Notice that we do not need this hypothesis in our setting, thanks to our definition of the support.

Theorem 2. *Let $M, N \in \Lambda$, $\underline{M} \xrightarrow{\square} \underline{N}$ if and only if $M \rightarrow_{\Lambda}^* N$.*

Corollary 1. *Let $M, N \in \Lambda$, $M \simeq_{\mathcal{R}} N$ if and only if $M \simeq_{\Lambda} N$.*

4 Normalizability

A substantial advantage of the weak algebraic λ -calculus is the fact that weak algebraic equality does not break typing in presence of negative coefficients, contrary to the non-weak version. In this section, we illustrate this improvement by characterizing normalizability properties as typability in suitable intersection type systems. First observe that, in a λ -calculus with sums, two distinct modes of normalization naturally appear: *must* and *may*.

Definition 12 (Must-normalizability). We say a term is in *must-normal form* if it does not reduce under $\xrightarrow{\square}$. We say a term is *must-normalizable* if it reduces to a must-normal form.

$$\begin{array}{c}
\frac{}{\Gamma, x : \alpha \vdash x : \alpha} \text{ (v)} \quad \frac{\Gamma, x : \alpha \vdash M : \beta}{\Gamma \vdash \lambda x.M : \alpha \rightarrow \beta} \text{ (\lambda)} \quad \frac{\Gamma \vdash M : \alpha \rightarrow \beta \quad \Gamma \vdash N : \alpha}{\Gamma \vdash MN : \beta} \text{ (@)} \\
\frac{\Gamma \vdash M : \alpha \quad \Gamma \vdash N : \alpha}{\Gamma \vdash M + N : \alpha} \text{ (+)} \quad \frac{\Gamma \vdash M : \alpha}{\Gamma \vdash aM : \alpha} \text{ (Scal)} \quad \frac{\Gamma \vdash M : \alpha \quad \Gamma \vdash M : \beta}{\Gamma \vdash M : \alpha \cap \beta} \text{ (\cap)} \\
\frac{}{\Gamma \vdash M : \Omega} \text{ (\Omega)} \quad \frac{\Gamma \vdash M : \alpha \quad \alpha \preceq \alpha'}{\Gamma \vdash M : \alpha'} \text{ (sub)} \quad \frac{}{\Gamma \vdash \mathbf{0} : \alpha} \text{ (z)}
\end{array}$$

(a) Typing rules

$$\frac{}{\alpha \preceq \alpha} \quad \frac{}{\alpha_1 \cap \alpha_2 \preceq \alpha_i} \quad \frac{\alpha' \preceq \alpha \quad \beta \preceq \beta'}{\alpha \rightarrow \beta \preceq \alpha' \rightarrow \beta'} \quad \frac{}{(\alpha \rightarrow \beta) \cap (\alpha \rightarrow \gamma) \preceq \alpha \rightarrow (\beta \cap \gamma)}$$

(b) Subtyping rules

Figure 3. The intersection type system $\mathcal{D}\Omega_+$ for the weak algebraic λ -calculus [PTV16]**Definition 13** (May-normalizability). We inductively define *may-normal forms* as follows:

- \underline{x} is in may-normal form;
- if $\underline{s} \in \underline{\mathbb{B}}_{\mathbb{R}}$ is in may-normal form, then $\underline{\lambda x.s}$ is too;
- if $\underline{s} \in \underline{\mathbb{B}}_{\mathbb{R}}$ and $\underline{T} \in \underline{\mathbb{C}}_{\mathbb{R}}$ are in may-normal form and \underline{s} is not an abstraction, then \underline{sT} is in may-normal form;
- if $\underline{s} \in \underline{\mathbb{B}}_{\mathbb{R}}$ is in may-normal form, then for any $T \in \mathbb{C}_{\mathbb{R}}$, $\underline{as} + \underline{T}$ is too.

We say a term is *may-normalizable* whenever it reduces to a term in may-normal form.

So, if a sum $\sum a_i s_i$ is must-normal, *each* s_i is also must-normal, meaning that each sum inside s_i is must-normal, inductively (see Proposition 3 below). On the other hand, for $\sum a_i s_i$ to be may-normal, it is sufficient that *at least one* s_i is may-normal.

Also observe that that it would not make sense to consider strong normalizability for one-step reduction. Indeed, recall that, in the algebraic λ -calculus, when \mathbb{R} contains all positive rational numbers, the only strongly normalizable terms are normal terms. In our current setting, we have a similar issue *whatever the semiring*:

Proposition 2. *Every term that reduces has an infinite reduction sequence for $\xrightarrow{\Omega}$.*

Proof. Let \underline{M} be a non-normal term, let \underline{N} be a term to which \underline{M} reduces. Then, $\underline{M} = \underline{M} + \underline{0M} \xrightarrow{\Omega} \underline{M} + \underline{0N} = \underline{M} + \underline{0M} + \underline{0N} \xrightarrow{\Omega} \underline{M} + \underline{0N} + \underline{0N} = \underline{M} + \underline{0N}$. \square

Thus type systems can only ensure (weak) normalizability for one-step reduction — but canonical reduction avoids precisely that drawback.

4.1 Must-normalizability

We now follow Krivine's proof [Kri93] to characterize normalizability as typability in the intersection type system $\mathcal{D}\Omega_+$ of Figure 3.

Definition 14. The intersection types of system $\mathcal{D}\Omega_+$ are given by the following grammar:

$$\alpha, \beta, \dots ::= o \mid \Omega \mid \alpha \rightarrow \beta \mid \alpha \cap \beta$$

From now on, we write $\Pi \triangleright \Gamma \vdash M : \alpha$ whenever Π is a proof-tree witnessing $\Gamma \vdash M : \alpha$.**Proposition 3.** *$\underline{N} \in \underline{\mathbb{C}}_{\mathbb{R}}$ is must-normal if and only if each $\underline{s} \in \text{Supp}(\underline{N})$ is must-normal. A simple must-normal term \underline{s} is of the form $\underline{\lambda x_1, \dots, x_n. y N_1 \dots N_k}$ with each \underline{N}_i must-normal.*

Proof. For the first statement, both implications are immediate, thanks to the definition of the support. The second is proved by a straightforward induction on $\text{can}(s)$. \square

4.1.1 Compatibility of typing with the module structure

Lemma 11. *The following rules are admissible:*

$$\frac{\Gamma, x : \alpha \vdash M : \beta \quad \gamma \preceq \alpha}{\Gamma, x : \gamma \vdash M : \beta} \text{ (sub')} \quad \frac{\Gamma \vdash M : \alpha \quad \alpha \preceq^* \beta}{\Gamma \vdash M : \beta} \text{ (sub}^*\text{)} \quad \frac{\alpha' \preceq^* \alpha \quad \beta \preceq^* \beta'}{\alpha \rightarrow \beta \preceq^* \alpha' \rightarrow \beta'}$$

Proof. For the rule (sub'), by induction on Π such that $\Pi \triangleright \Gamma, x : \gamma \vdash M : \beta$, replacing each introduction of $x : \alpha$ by an introduction of $x : \gamma$, and an application of (sub) with $\gamma \preceq \alpha$. For the second rule, one just iterates rule (sub). For the third rule, we have $\alpha' \preceq^k \alpha$ and $\beta \preceq^l \beta'$, we assume w.l.o.g. $k \leq l$. One iterates the corresponding rule k times, and then, iterates $l - k$ times the same rule but with a premise $\alpha \preceq \alpha$. \square

Lemma 12. *If $\Pi \triangleright \Gamma \vdash M : \alpha$, then there is a tree Φ such that $\Phi \triangleright \Gamma \vdash M : \alpha$ and for each occurrence of (sub) in Φ , the first subtree contains only rules (sub), (\cap) and (v).*

Proof. The proof is tedious but mostly straightforward by induction on Π : only the case of (sub) needs some care. \square

Lemma 13. $\Gamma \vdash \sum_{i=1}^n a_i s_i : \alpha$ is derivable iff $\Gamma \vdash s_i : \alpha$ is derivable for all $i \in \{1, \dots, n\}$.

Proof. For the direct implication, by induction on $n \in \mathbb{N}$; after applying Lemma 12, the last rule must be (+). For the reverse implication one iterates the rules (Scal) and (+). \square

Proposition 4. *Let $M, N \in \Lambda_R$. If $M \triangleq N$, $\Gamma \vdash M : \alpha$ is derivable iff $\Gamma \vdash N : \alpha$ is derivable.*

Proof. We actually prove this proposition in the case $N = \text{can}(M)$. For the direct implication, by induction on M , one just replaces each premises with their equivalent with their canonical form by I.H. For the reverse implication, by induction on M , after applying Lemma 12. \square

In the remaining of this section, we apply Krivine's proof to characterize normalizability as typability in system $\mathcal{D}\Omega_+$, and as a normalizability property for a particular reduction that defines, almost, a reduction strategy.

4.1.2 Subject expansion

Lemma 14. *Let Γ, Δ be typing contexts, α a type and $M \in L_R$. If $\Gamma \vdash M : \alpha$ is derivable and if for all variables x , $\Delta(x) \preceq^* \Gamma(x)$ is derivable, then $\Delta \vdash M : \alpha$ is derivable.*

Proof. One simply iterates the rule (sub') from Lemma 11. \square

The following crucial lemma holds thanks to the specificity of system $\mathcal{D}\Omega_+$ (Figure 3). For instance, in a simple type system, when x does not appear in M , we may be able to type $M[N/x]$ even if N is not typable. Here, we can give it type Ω with rule (Ω).

Lemma 15 (Reverse substitution lemma). *If $\Gamma \vdash M[N/x] : \beta$ is derivable, then there is a type α such that $\Gamma, x : \alpha \vdash M : \beta$ and $\Gamma \vdash N : \alpha$ are derivable.*

Proof. By induction on $M \in L_R$, using Lemma 14. We also use rule (Ω) in the case of a variable. \square

Lemma 16 (Subject expansion). *Let $\underline{M}, \underline{N} \in \underline{C}_R$. If $\underline{M} \xrightarrow{\square} \underline{N}$ and $\Gamma \vdash \underline{N} : \alpha$ is derivable, then $\Gamma \vdash \underline{M} : \alpha$ is derivable.*

Proof. By induction on the reduction, in the case of a fired redex, use Lemma 15. \square

Lemma 17. *Every must-normal term is typable in system $\mathcal{D}\Omega_+$ with no occurrence of Ω , neither in the context nor in the resulting type.*

Proof. By induction on must-normal terms, we show $\underline{M} \in \mathbb{C}_R$ is typable with a type of the form $v_1 \rightarrow \dots \rightarrow v_K \rightarrow \alpha$ in some context Δ . By Proposition 3, $\underline{M} = \sum_{i=1}^N a_i s_i$ with $\text{can}(s_i) = \lambda x_1, \dots, x_{k_i}. y_i M_1^i \dots M_{n_i}^i$, with M_j^i must-normal. Let $K \geq \max\{k_1, \dots, k_N\}$.

By induction hypothesis, each M_j^i are typable with type δ_j^i , in context Δ_j^i , with no occurrence of Ω in neither of them.

We define contexts with distinct holes \mathfrak{U}_i for $i \in \mathbb{N}$:

$$\mathcal{U}, \mathcal{V}, \dots ::= o \mid \Omega \mid \mathcal{U} \rightarrow \mathcal{V} \mid \mathcal{U} \cap \mathcal{V} \mid \mathfrak{U}_i$$

We now assign contexts to each x_l and for each term of the support. Let $i \in \{1, \dots, N\}, l \in \{1, \dots, K\}$, we set $\mathcal{U}_l^{(i)} = \begin{cases} \delta_1^i \rightarrow \dots \rightarrow \delta_{n_i}^i \rightarrow \mathfrak{U}_{k_i+1} \rightarrow \dots \rightarrow \mathfrak{U}_K \rightarrow \alpha & \text{otherwise} \\ \Omega & \text{if } l > k_i \text{ or } y_i \neq x_l \end{cases}$

Now, by downwards induction on $l \in \{1, \dots, K\}$, we define types τ_l such that $\forall i \in \{1, \dots, N\}$, if $\mathcal{U}_l^{(i)} \neq \Omega$, $\tau_l \preceq^* \delta_1^i \rightarrow \dots \rightarrow \delta_{n_i}^i \rightarrow \tau_{k_i+1} \rightarrow \dots \rightarrow \tau_K \rightarrow \alpha$ is derivable.

In the case $l = K$, all $\mathcal{U}_l^{(i)}$ are actual types: indeed, either $k_i < l$ and $\mathcal{U}_l^{(i)} = \Omega$, either $k_i = l = K$ and $\mathcal{U}_l^{(i)} = \delta_1^i \rightarrow \dots \rightarrow \delta_{n_i}^i \rightarrow \alpha$. Let $I_K = \{i \in \{1, \dots, N\} \mid \mathcal{U}_K^{(i)} \neq \Omega\}$, we set $\tau_K = \bigcap_{i \in I_K} \mathcal{U}_K^{(i)}$ and we easily check it satisfies the hypothesis.

Let's assume the result holds for some $l+1 \leq K$, we extend the result to l . First, let $i \in \{1, \dots, n\}$, we assign actual types $\tilde{\mathcal{U}}_l^{(i)}$ to each $\mathcal{U}_l^{(i)}$. If $l > k_i$, then $\mathcal{U}_l^{(i)} = \Omega$ and we set $\tilde{\mathcal{U}}_l^{(i)} = \Omega$. If $l \leq k_i$, then $\mathcal{U}_l^{(i)} = \delta_1^i \rightarrow \dots \rightarrow \delta_{n_i}^i \rightarrow \mathfrak{U}_{k_i+1} \rightarrow \dots \rightarrow \mathfrak{U}_K \rightarrow \alpha$. Remark that for $k \in \{k_i+1, \dots, K\}$, τ_k are defined by I.H. since $l \leq k_i$. Hence, we can set $\tilde{\mathcal{U}}_l^{(i)} = \delta_1^i \rightarrow \dots \rightarrow \delta_{n_i}^i \rightarrow \tau_{k_i+1} \rightarrow \dots \rightarrow \tau_K \rightarrow \alpha$. Let $I_l = \{i \in \{1, \dots, N\} \mid \tilde{\mathcal{U}}_l^{(i)} \neq \Omega\}$, we set $\tau_l = \bigcap_{i \in I_l} \tilde{\mathcal{U}}_l^{(i)}$ and easily check it satisfies the hypothesis.

Types τ_l are intended to type bound head variables. However, these can also appear as free in the M_j^i . From now on, whenever y is not defined in Δ_j^i , we set $\Delta_j^i(y) = \Omega$. For all $l \in \{1, \dots, K\}$, let $J_l^i = \{j \in \{1, \dots, n_i\} \mid \Delta_j^i(x_l) \neq \Omega\}$, we set $v_l = \tau_l \cap \bigcap_{i=1}^N \bigcap_{j \in J_l^i} \Delta_j^i(x_l)$.

Now that we have our type, we can set our typing context: we need the typing context of each M_j^i , and on the other hand, we also need a typing context for each free head variable. For each $i \in \{1, \dots, N\}$ and $y \in \text{fv}(\underline{M})$, we define γ_y^i a type for y , which will later be used in typing $\text{can}(s_i)$. Let $\mathcal{J}_i^{l,y} = \{j \in \{1, \dots, n_i\} \mid \Delta_j^i(y) \neq \Omega\}$.

Recall y_i is the head variable of $\text{can}(s_i)$. We set:

$$\gamma_y^i = \begin{cases} (\delta_1^i \rightarrow \dots \rightarrow \delta_{n_i}^i \rightarrow v_{k_i+1} \rightarrow \dots \rightarrow v_K \rightarrow \alpha) \cap \bigcap_{j \in \mathcal{J}_i^{l,y}} \Delta_j^i(y) & \text{if } y = y_i \\ \bigcap_{j \in \mathcal{J}_i^{l,y}} \Delta_j^i(y) \end{cases}$$

Finally, let $I_T = \{i \in \{1, \dots, N\} \mid \gamma_{y_i}^i \neq \Omega\}$, we set $\Gamma = \{y : \bigcap_{i \in I_T} \gamma_y^i \mid y \in \text{fv}(\underline{M})\}$.

We then prove that $\Gamma \vdash \text{can}(s_i) : v_1 \rightarrow \dots \rightarrow v_K \rightarrow \alpha$ is derivable for each i . To lighten our proof tree, we set $\Xi_i = \Gamma \cup \{x_1 : v_1, \dots, x_{k_i} : v_{k_i}\}$.

First, if y_i is free, then:

$$\frac{\frac{\Xi_i \vdash y_i : \bigcap_{i=1}^N \gamma_{y_i}^i \quad \bigcap_{i=1}^N \gamma_{y_i}^i \preceq^* \delta_1^i \rightarrow \dots \rightarrow v_K \rightarrow \alpha}{\Xi_i \vdash y_i : \delta_1^i \rightarrow \dots \rightarrow \delta_{n_i}^i \rightarrow v_{k_i+1} \rightarrow \dots \rightarrow v_K \rightarrow \alpha} \quad \frac{\text{I.H. and Lemma 14}}{\Xi_i \vdash M_1^i : \delta_1^i} \quad \frac{\text{I.H. and Lemma 14}}{\dots} \quad \frac{\text{I.H. and Lemma 14}}{\Xi_i \vdash M_{n_i}^i : \delta_{n_i}^i}}{\Xi_i \vdash y_i M_1^i \dots M_{n_i}^i : v_{k_i+1} \rightarrow \dots \rightarrow v_K \rightarrow \alpha}$$

$$\frac{\vdots}{\Gamma \vdash \lambda x_1, \dots, x_{k_i}. y_i M_1^i \dots M_{n_i}^i : v_1 \rightarrow \dots \rightarrow v_K \rightarrow \alpha}$$

Then, if y_i is bound, there is a $l_0 \in \{1, \dots, k_i\}$ such that $y_i = x_{l_0}$, then:

$$\begin{array}{c}
\vdots \\
\frac{\frac{\Xi_i \vdash x_{l_0} : v_{l_0}}{\Xi_i \vdash x_{l_0} : \delta_1^i \rightarrow \dots \rightarrow \delta_{n_i}^i \rightarrow v_{k_i+1} \rightarrow \dots \rightarrow v_K \rightarrow \alpha} \quad \frac{v_{l_0} \preceq^* \delta_1^i \rightarrow \dots \rightarrow v_{k_i} \rightarrow \alpha}{\Xi_i \vdash M_1^i : \delta_1^i} \quad \frac{I.H. \text{ and Lemma 14}}{\dots} \quad \frac{I.H. \text{ and Lemma 14}}{\Xi_i \vdash M_{n_i}^i : \delta_{n_i}^i}}{\Xi_i \vdash x_{l_0} M_1^i \dots M_{n_i}^i : v_{k_i+1} \rightarrow \dots \rightarrow v_K \rightarrow \alpha} \\
\vdots \\
\frac{\Gamma \vdash \lambda x_1, \dots, x_{k_i}. x_{l_0} M_1^i \dots M_{n_i}^i : v_1 \rightarrow \dots \rightarrow v_K \rightarrow \alpha}{\Gamma \vdash \lambda x_1, \dots, x_{k_i}. x_{l_0} M_1^i \dots M_{n_i}^i : v_1 \rightarrow \dots \rightarrow v_K \rightarrow \alpha}
\end{array}$$

And we conclude by Lemma 13 and Proposition 4. \square

Corollary 2. *Normalizable terms are typable in system $\mathcal{D}\Omega_+$ with no occurrence of Ω , neither in the context nor in the resulting type.*

4.1.3 Subject reduction

We define a new reduction which extends the leftmost outermost reduction of the pure λ -calculus. These kind of reductions are normalization strategies in numerous λ -calculi.

Definition 15 (Leftmost outermost reduction). Let $\rightarrow_{\text{lm}k} = \emptyset$:

- $\lambda x.s \rightarrow_{\text{lm}k+1} \lambda x.S$ if $s \rightarrow_{\text{lm}k} S$;
- $sT \rightarrow_{\text{lm}k+1} ST$ if s is not an abstraction and $s \rightarrow_{\text{lm}k} S$;
- $sT \rightarrow_{\text{lm}k+1} sT'$ if s is not an abstraction, is must-normal, and $T \widehat{\rightarrow}_{\text{lm}k} T'$;
- $(\lambda x.s)T \rightarrow_{\text{lm}k+1} s[T/x]$.

Then, let $\rightarrow_{\text{lm}} = \bigcup_{k \in \mathbb{N}} \rightarrow_{\text{lm}k}$, the leftmost outermost reduction is then $\widehat{\rightarrow}_{\text{lm}} = \bigcup_{k \in \mathbb{N}} \widehat{\rightarrow}_{\text{lm}k}$.

We'll only show typable terms are normalizable for this particular reduction, for two reasons: first, we can avoid the issue of Proposition 2, and since normalizable terms are typable, we will still be able to prove the characterization of Theorem 3. We will moreover obtain that $\widehat{\rightarrow}_{\text{lm}}$ induces a normalization strategy along the way.

Definition 16 (Leftmost normalizing term). Let \mathbb{N}_R^{lm} be the set of strongly normalizing base terms for $\widehat{\rightarrow}_{\text{lm}}$.

Proposition 5. *The set of all strongly normalizing terms for $\widehat{\rightarrow}_{\text{lm}}$ is then $R\langle \mathbb{N}_R^{\text{lm}} \rangle$.*

Proof. It is clear that a term with a non-normalizing term in the support is not strongly normalizing. For the reverse inclusion, consider the multiset of maximal length of derivation for each term of the support, and observe it decreases along leftmost outermost reduction. \square

Definition 17 (\mathcal{N} -saturated set of terms). Let $\mathcal{N} \subseteq \mathbb{B}_R$. We say $\mathcal{X} \subseteq \mathbb{B}_R$ is \mathcal{N} -saturated when, for all $\underline{s} \in \mathcal{N}$, $\underline{T}_0, \dots, \underline{T}_n \in R\langle \mathcal{N} \rangle$, $s[T_0/x]T_1 \dots T_n \in R\langle \mathcal{X} \rangle$ entails $\underline{s}T_0 \dots T_n \in \mathcal{X}$.

Proposition 6. \mathbb{N}_R^{lm} is \mathbb{N}_R^{lm} -saturated.

Proof. Straightforward from the definition of $\widehat{\rightarrow}_{\text{lm}}$. \square

Definition 18. Let $\mathcal{X}, \mathcal{Y} \subseteq \mathbb{B}_R$, we set $\mathcal{X} \rightarrow \mathcal{Y} = \{ \underline{s} \in \mathbb{B}_R \mid \forall \underline{T} \in R\langle \mathcal{X} \rangle, sT \in \mathcal{Y} \}$.

Proposition 7. *Let $\mathcal{X}, \mathcal{X}', \mathcal{Y}, \mathcal{Y}' \subseteq \mathbb{B}_R$. If $\mathcal{X}' \subseteq \mathcal{X}$ and $\mathcal{Y} \subseteq \mathcal{Y}'$, then, $\mathcal{X} \rightarrow \mathcal{Y} \subseteq \mathcal{X}' \rightarrow \mathcal{Y}'$.*

Proof. Straightforward from definition 18. \square

Definition 19. Let $\mathbb{E}_R^{\text{lm}} = \{ xT_1 \dots T_n \mid xT_1 \dots T_n \in \mathbb{N}_R^{\text{lm}} \}$ be the set of *neutral terms*.

Proposition 8. *Let \mathcal{X}, \mathcal{Y} such that $\mathbb{E}_R^{\text{lm}} \subseteq \mathcal{X}, \mathcal{Y} \subseteq \mathbb{N}_R^{\text{lm}}$:*

- *If \mathcal{Y} is \mathbb{N}_R^{lm} -saturated, then $\mathbb{E}_R^{\text{lm}} \subseteq \mathcal{X} \rightarrow \mathcal{Y} \subseteq \mathbb{N}_R^{\text{lm}}$ and it is \mathbb{N}_R^{lm} -saturated;*
- *If \mathcal{X} and \mathcal{Y} are \mathbb{N}_R^{lm} -saturated, then $\mathbb{E}_R^{\text{lm}} \subseteq \mathcal{X} \cap \mathcal{Y} \subseteq \mathbb{N}_R^{\text{lm}}$ and it is \mathbb{N}_R^{lm} -saturated.*

Proof. For the first assertion, the fact that $\mathcal{X} \rightarrow \mathcal{Y}$ is saturated is straightforward from the definitions. From Proposition 7, $\underline{N}_R^{\text{lm}} \rightarrow \underline{E}_R^{\text{lm}} \subseteq \mathcal{X} \rightarrow \mathcal{Y} \subseteq \underline{E}_R^{\text{lm}} \rightarrow \underline{N}_R^{\text{lm}}$. Conclude by noticing that $\underline{E}_R^{\text{lm}} \subseteq \underline{N}_R^{\text{lm}} \rightarrow \underline{E}_R^{\text{lm}}$ and $\underline{E}_R^{\text{lm}} \rightarrow \underline{N}_R^{\text{lm}} \subseteq \underline{N}_R^{\text{lm}}$. The second statement is straightforward. \square

Definition 20 (Type interpretation). We inductively define type interpretation:

- $o^* = \underline{N}_R^{\text{lm}}$;
- $(\alpha \rightarrow \beta)^* = \alpha^* \rightarrow \beta^*$;
- $(\alpha \cap \beta)^* = \alpha^* \cap \beta^*$.

Lemma 18. *If α contains no occurrence of Ω , $\underline{E}_R^{\text{lm}} \subseteq \alpha^* \subseteq \underline{N}_R^{\text{lm}}$ and α^* is $\underline{N}_R^{\text{lm}}$ -saturated.*

Proof. Straightforward induction on types, using Proposition 8. \square

Proposition 9. *The following statements hold:*

- $R\langle \mathcal{X} \cap \mathcal{Y} \rangle = R\langle \mathcal{X} \rangle \cap R\langle \mathcal{Y} \rangle$;
- *If $\mathcal{X} \subseteq \mathcal{Y} \subseteq \underline{B}_R$, then $R\langle \mathcal{X} \rangle \subseteq R\langle \mathcal{Y} \rangle$;*
- *If $\alpha \preceq \beta$ is derivable, then $\alpha^* \subseteq \beta^*$.*

Proof. The first two statements are straightforward. For the last one, we prove each subtyping rule is correct w.r.t. inclusion of type interpretation, using Proposition 7. \square

Lemma 19. *If $x_1 : \alpha_1, \dots, x_n : \alpha_n \vdash M : \beta$ is derivable, and if Ω does not occur in any of these types, then for any $\underline{T}_1 \in R\langle \alpha_1^* \rangle, \dots, \underline{T}_n \in R\langle \alpha_n^* \rangle$, $\underline{M}[\underline{T}_1, \dots, \underline{T}_n/x_1, \dots, x_n] \in R\langle \beta^* \rangle$.*

Proof. By induction on the canonical form of M , using Proposition 9 and Lemma 18. \square

Corollary 3. *For any $M \in \Lambda_R$ typable in system $\mathcal{D}\Omega_+$ with no occurrence of Ω in the context and its type, \underline{M} is strongly normalizing for $\widehat{\rightarrow}_{\text{lm}}$, and is thus must-normalizable.*

Proof. Notice variables are neutral terms and apply Lemma 19 and then Lemma 18. \square

Theorem 3. *The reduction $\widehat{\rightarrow}_{\text{lm}}$ is a strategy for must-normalization in the following sense: if a term is must-normalizable, it is strongly must-normalizing for this reduction.*

Proof. Let $M \in \Lambda_R$ and consider the following statements:

1. M is normalizable;
2. M is strongly normalizing for $\widehat{\rightarrow}_{\text{lm}}$;
3. There is Γ, α such that $\Gamma \vdash M : \alpha$ is derivable, with no occurrence of Ω in Γ and α .

Clearly, (2) \Rightarrow (1). Corollary 3 states (3) \Rightarrow (2) and Theorem 2 states (1) \Rightarrow (3). \square

Notice however that $\widehat{\rightarrow}_{\text{lm}}$ is not a strategy; the choice of the term of the support to reduce remains. This can be bypassed, for instance considering a sequence of simple terms where each simple term appears infinitely often, we can define a strategy by replacing the arbitrary choice by reading the sequence and picking the first simple term of the sequence appearing in the support. We can then define *the* reduction sequence of a term under this strategy.

4.1.4 Confluence of the canonical reduction

We saw that $\widehat{\rightarrow}$ is not locally confluent. Fortunately, thanks to the results of this section, we can still state a weaker confluence property, since $\widehat{\rightarrow}_{\text{lm}} \subset \widehat{\rightarrow}$.

Lemma 20. *$\widehat{\rightarrow}$ is confluent on must-normalizable terms.*

Proof. Let $\underline{M}, \underline{N}, \underline{N}' \in \underline{C}_R$, $\underline{M} \widehat{\rightarrow}^* \underline{N}$, $\underline{M} \widehat{\rightarrow}^* \underline{N}'$ and \underline{M} is normalizable. By confluence of $\widehat{\rightarrow}$, \underline{N} and \underline{N}' are both normalizable. Thus, there is a $\widehat{\rightarrow}_{\text{lm}}$ -normalization reduction sequence from \underline{N} and \underline{N}' by Theorem 3. Finally, their normal forms are equal by confluence of $\widehat{\rightarrow}$. \square

$$\begin{array}{c}
\frac{}{x : [\alpha] \vdash x : \alpha} \text{ (v)} \quad \frac{\Gamma, x : \bar{\alpha} \vdash M : \beta}{\Gamma \vdash \lambda x.M : \bar{\alpha} \rightarrow \beta} \text{ (\lambda)} \quad \frac{\Gamma_1 \vdash M : \bar{\alpha} \rightarrow \beta \quad \Gamma_2 \vdash_! N : \bar{\alpha}}{\Gamma_1 + \Gamma_2 \vdash MN : \beta} \text{ (@)} \\
\frac{\Gamma \vdash M : \alpha}{\Gamma \vdash M + N : \alpha} \text{ (+)} \quad \frac{\Gamma \vdash M : \alpha}{\Gamma \vdash aM : \alpha} \text{ (Scal)} \quad \frac{\Gamma_1 \vdash M : \alpha_1 \quad \dots \quad \Gamma_n \vdash M : \alpha_n}{\Gamma_1 + \dots + \Gamma_n \vdash_! M : [\alpha_1, \dots, \alpha_n]} \text{ (!)}
\end{array}$$

Figure 4. Algebraic \mathcal{W}_+ type system

4.2 May-normalizability

Now we characterize *may-normalizability* (Definition 13) as typability in system \mathcal{W}_+ (Figure 4). Here, typing a sum boils down to typing one of its terms (Rule (+)): this allows us to use a non-idempotent intersection type system. Indeed, our system is inspired by de Carvalho's system R [dC18], which is itself grounded in the relational model of the λ -calculus, interpreting sum as union. A strength of non-idempotent intersection type systems is that the proof of normalization can be done simply by induction on the size of typing derivations [BKV17]: this applies in our setting.

Observe that rules with several premises *sum* their contexts in the following sense: if $\Gamma_1(x) = [\alpha_1, \dots, \alpha_n]$ and $\Gamma_2(x) = [\beta_1, \dots, \beta_k]$, then $(\Gamma_1 + \Gamma_2)(x) = [\alpha_1, \dots, \alpha_n, \beta_1, \dots, \beta_k]$.

Definition 21. Types of system \mathcal{W}_+ are given by the following mutually inductive grammars, where $[\alpha_1, \dots, \alpha_n]$ is a finite multiset standing for $\alpha_1 \cap \dots \cap \alpha_n$:

$$\alpha, \beta, \dots ::= o \mid \bar{\alpha} \rightarrow \beta \quad \bar{\alpha}, \dots ::= [\alpha_1, \dots, \alpha_n]$$

Notice we allow intersections only on the left-hand sides of arrows: the conclusion of rule (!) is *not* a typing judgement: it uses the symbol $\vdash_!$ instead of \vdash . We say M is typable if there are Γ and α such that $\Gamma \vdash M : \alpha$ is derivable. In particular, no term has type \square .

4.2.1 Compatibility of typing with the module structure

Lemma 21. $\Gamma \vdash \sum_{i=1}^n a_i s_i : \alpha$ is derivable iff $\Gamma \vdash s_j : \alpha$ is derivable for a $j \in \{1, \dots, n\}$.

Proof. By induction on $n \in \mathbb{N}$. \square

Proposition 10. Let $M, N \in \Lambda_R$. If $M \stackrel{\Delta}{=} N$, $\Gamma \vdash M : \alpha$ is derivable iff $\Gamma \vdash N : \alpha$ is derivable.

Proof. We actually prove this proposition in the case $N = \text{can}(M)$, by induction on M . \square

4.2.2 Subject expansion

Lemma 22. Let $M, N \in \Lambda_R$. If $\Gamma \vdash M[N/x] : \beta$ is derivable, then there is Γ_0, Γ_1 and $\bar{\alpha}$ such that $\Gamma_0, x : \bar{\alpha} \vdash M : \beta$ and $\Gamma_1 \vdash_! N : \bar{\alpha}$ are derivable, and $\Gamma = \Gamma_0 + \Gamma_1$.

Proof. Straightforward induction on permutative terms. \square

Lemma 23 (Subject expansion). Let $\underline{M}, \underline{N} \in \underline{C}_R$. If $\underline{M} \stackrel{\square}{\rightarrow} \underline{N}$ and $\Gamma \vdash N : \alpha$ is derivable, then $\Gamma \vdash M : \alpha$ is derivable.

Proof. We prove this by induction on the reduction. If $\underline{M} = \underline{as} + \underline{T} \stackrel{\square}{\rightarrow} \underline{aS'} + \underline{T} = \underline{N}$: by Lemma 21, either $\Gamma \vdash T : \alpha$ is derivable, and it is immediate, either, $\Gamma \vdash S' : \alpha$ is derivable and we apply I.H. on $\underline{s} \rightarrow \underline{S'}$. The only other interesting case is the case of a fired redex, in which we apply Lemma 22. \square

Lemma 24. *May-normal forms are typable.*

Proof. Straightforward induction on may-normal forms. \square

Corollary 4. *May-normalizable terms are typable.*

4.2.3 Subject reduction

Lemma 25. *Let $M \in \Lambda_R$. If $\Pi \triangleright \Gamma \vdash M : \alpha$, and if there is no typed redex in Π , M is in may-normal form.*

Proof. Straightforward induction on the proof-tree. \square

In the following lemmas, we will denote by $|\Pi|$ the number of nodes in the proof-tree Π .

Lemma 26. *Let $\underline{M}, \underline{N} \in \mathcal{C}_R$ and x be a variable. If $\Pi_0 \triangleright \Gamma_0, x : \bar{\alpha} \vdash M : \beta, \Pi_1 \triangleright \Gamma_1 \vdash N : \bar{\alpha}$, there is a $\Pi_0[\Pi_1/x]$ such that $\Pi_0[\Pi_1/x] \triangleright \Gamma_0 + \Gamma_1 \vdash M[N/x] : \beta$ and $|\Pi_0[\Pi_1/x]| \leq |\Pi_0| + |\Pi_1|$.*

Proof. By induction on Π_0 . Intuitively, $\Pi_0[\Pi_1/x]$ is Π_0 where each occurrence of $x : [\alpha] \vdash x : \alpha$ is substituted by the corresponding proof for N in Π_1 . \square

Lemma 27. *Let $\underline{s} \in \mathcal{B}_R, \underline{T} \in \mathcal{C}_R, \Pi \triangleright \Gamma \vdash s : \alpha, \underline{s} \rightarrow \underline{T}$, and such that the fired redex is typed in Π . There is a proof tree $\Phi, \underline{U} \in \Lambda_R$ such that $\underline{U} = \underline{T}, \Phi \triangleright \Gamma \vdash U : \alpha$ and $|\Pi| > |\Phi|$.*

Proof. By induction on the reduction, in the case of a fired redex, we apply Lemma 26. \square

Corollary 5. *Let $M \in \Lambda_R$. If M is \mathcal{W}_+ -typable, then M is may-normalizable.*

Proof. Let $N \in \Lambda_R, \Pi, \Gamma, \alpha$ be a proof-tree, context and type such that $\underline{M} = \underline{N}$ and $\Pi \triangleright \Gamma \vdash N : \alpha$. Consider the following mutually inductive sequences of term and proof-tree:

$$\begin{cases} M_0 = N \\ M_{n+1} : M_n \xrightarrow{\text{firing}} M_{n+1}, \text{ firing a typed redex in } \Pi_n \end{cases} \quad \begin{cases} \Pi_0 = \Pi \\ \Pi_{n+1} = \text{the proof tree given} \\ \text{by Lemma 27 from } \Pi_n \end{cases}$$

By Lemma 27, there is a $n_0 \in \mathbb{N}$ such that M_{n_0+1} is undefined. By subject reduction, we must have some typed terms in Π_{n_0} in the support of M_{n_0} . Hence, we reached a typed term but containing no typed redex in Π_{n_0} . Thus M_{n_0} is in may-normal form by Lemma 25. \square

Theorem 4. *Let $\underline{M} \in \mathcal{C}_R$, \underline{M} is typable in system \mathcal{W}_+ if and only if \underline{M} is may-normalizable.*

5 Taylor expansion

In the final section of this paper, we investigate an entirely different aspect of λ -calculus: we introduce a notion of Taylor expansion [ER03] on this calculus. Taylor expansion is an operation on terms, recursively replacing each application by an infinite sum of *multilinear* application. In this section, we require \mathbb{R} to be commutative, which is necessary because dealing with vectors of resource terms involves *multilinear* operations.

5.1 Resource λ -calculus

Definition 22 (Resource terms). We consider the following mutually inductive grammars, where $[\sigma_1, \dots, \sigma_n]$ denotes a finite multiset:

$$\Delta \ni \sigma, \rho, \dots ::= x \mid \lambda x. \sigma \mid \langle \sigma \rangle \bar{\rho} \quad !\Delta \ni \bar{\sigma}, \bar{\rho}, \dots ::= [\sigma_1, \dots, \sigma_n]$$

We write $\bar{\sigma} \cdot \bar{\rho}$ for the concatenation of two such multisets. Any resource term or multiset of resource terms is a *resource expression*. Resource expressions are considered up to α -equivalence. We denote by $(!) \Delta$ either Δ or $!\Delta$. When it appears several times in the same context, each occurrence denotes the same set.

We will see that sums appear along reduction, hence, the actual objects we consider are formal finite sums of resource expressions (without coefficients). We extend syntactic constructs to sums of terms:

$$\begin{aligned} \lambda x. \sum \sigma_i &= \sum \lambda x. \sigma_i \quad \left\langle \sum \sigma_i \right\rangle \sum \bar{\rho}_j = \sum \sum \langle \sigma_i \rangle \bar{\rho}_j \\ \left[\sum \rho_{i_1}^1, \dots, \sum \rho_{i_n}^n \right] &= \sum_{i_1, \dots, i_n} [\rho_{i_1}^1, \dots, \rho_{i_n}^n] \end{aligned}$$

Note application is *bilinear*, as opposed to the application on λ -terms. Furthermore, notice these equations define a rewriting system, from left to right; let $\mathbb{N}[(!)\Delta]$ be the set of such sums.

Definition 23. Let $\Sigma \in \mathbb{N}[(!)\Delta]$, and $\sum_{i=1}^N \sigma_i$ be the normal form of Σ in the above rewriting system, we denote by $\|\Sigma\|$ (resp. $|\Sigma|$) the multiset $[\sigma_1, \dots, \sigma_N]$ (resp. $\{\sigma_1, \dots, \sigma_N\}$).

While reduction in the ordinary λ -calculus is based on the substitution $M[N/x]$, replacing each occurrence of x by N , the resource λ -calculus is based upon a *linear* substitution $\frac{\partial M}{\partial x} \cdot N$, replacing *one* occurrence of x by N .

Definition 24. We define *partial derivatives* by induction on resource terms:

$$\begin{aligned} \frac{\partial y}{\partial x} \cdot \rho &= \begin{cases} \rho & \text{if } x = y \\ \mathbf{0} & \text{otherwise} \end{cases} & \frac{\partial \langle \sigma \rangle \bar{\tau}}{\partial x} \cdot \rho &= \left\langle \frac{\partial \sigma}{\partial x} \cdot \rho \right\rangle \bar{\tau} + \langle \sigma \rangle \frac{\partial \bar{\tau}}{\partial x} \cdot \rho \\ \frac{\partial \lambda y. \sigma}{\partial x} \cdot \rho &= \lambda y. \left(\frac{\partial \sigma}{\partial x} \cdot \rho \right) \quad (y \notin \text{fv}(\sigma) \cup \{x\}) & \frac{\partial [\sigma_1, \dots, \sigma_n]}{\partial x} \cdot \rho &= \sum_{i=1}^n \left[\sigma_1, \dots, \frac{\partial \sigma_i}{\partial x} \cdot \rho, \dots, \sigma_n \right] \end{aligned}$$

Which we naturally extend to formal finite sums by bilinearity:

$$\frac{\partial \sum_{i=1}^n \sigma_i}{\partial x} \cdot \sum_{j=1}^k \rho_j = \sum_{i=1}^n \sum_{j=1}^k \frac{\partial \sigma_i}{\partial x} \cdot \rho_j$$

Definition 25 (Multilinear substitution). Let $\sigma \in \Delta$, $[\rho_1, \dots, \rho_n] \in !\Delta$, we set:

$$\partial_x \sigma \cdot [\rho_1, \dots, \rho_n] = \left(\frac{\partial^n \sigma}{\partial x^n} \cdot \rho_1, \dots, \rho_n \right) [\mathbf{0}/x]$$

Definition 26 (Resource reduction). We define $\rightarrow_r \subseteq (!)\Delta \times \mathbb{N}[(!)\Delta]$ inductively as follows:

- $\lambda x. \sigma \rightarrow_r \lambda x. \sigma'$ if $\sigma \rightarrow_r \sigma'$;
- $\langle \sigma \rangle \bar{\rho} \rightarrow_r \langle \sigma' \rangle \bar{\rho}$ if $\sigma \rightarrow_r \sigma'$;
- $\langle \sigma \rangle \bar{\rho} \rightarrow_r \langle \sigma \rangle \bar{\rho}'$ if $\bar{\rho} \rightarrow_r \bar{\rho}'$;
- $\langle \lambda x. \sigma \rangle \bar{\rho} \rightarrow_r \partial_x \sigma \cdot \bar{\rho}$;
- $[\sigma] \cdot \bar{\rho} \rightarrow_r [\sigma'] \cdot \bar{\rho}$ if $\sigma \rightarrow_r \sigma'$.

We extend \rightarrow_r to sums by setting $\sigma + \tau \rightarrow_r \sigma' + \tau$ when $\sigma \in (!)\Delta$, $\sigma', \tau \in \mathbb{N}[(!)\Delta]$ and $\sigma \rightarrow_r \sigma'$.

Proposition 11 ([ER08]). \rightarrow_r is confluent and strongly normalizing, thus we can define the *normal form of a resource term*: $NF(\sigma)$.

Definition 27 (Parallel resource reduction). Similarly, we define \Rightarrow_r inductively as follows:

- $x \Rightarrow_r x$;
- $\lambda x. \sigma \Rightarrow_r \lambda x. \sigma'$ if $\sigma \Rightarrow_r \sigma'$;
- $\langle \sigma \rangle \bar{\rho} \Rightarrow_r \langle \sigma' \rangle \bar{\rho}'$ if $\sigma \Rightarrow_r \sigma'$ and $\bar{\rho} \Rightarrow_r \bar{\rho}'$;
- $\langle \lambda x. \sigma \rangle \bar{\rho} \Rightarrow_r \partial_x \sigma \cdot \bar{\rho}'$ if $\sigma \Rightarrow_r \sigma'$ and $\bar{\rho} \Rightarrow_r \bar{\rho}'$;
- $[\rho_1, \dots, \rho_n] \Rightarrow_r [\rho'_1, \dots, \rho'_n]$ if $\rho_i \Rightarrow_r \rho'_i$ for all $i \in \{1, \dots, n\}$.

We extend \Rightarrow_r to sums by setting $\sum_{i=1}^n \sigma_i \Rightarrow_r \sum_{i=1}^k \sigma'_i$ when $\sigma_i \Rightarrow_r \sigma'_i$ for $1 \leq i \leq k$.

Definition 28 (Full parallel resource reduct). We define $F(\sigma)$ by induction on σ :

- $F(x) = x$;
- $F(\lambda x.\sigma) = \lambda x.F(\sigma)$;
- $F(\langle \lambda x.\sigma \bar{\rho} \rangle) = \partial_x F(\sigma) \cdot F(\bar{\rho})$;
- $F(\langle \sigma \bar{\rho} \rangle) = \langle F(\sigma) \rangle F(\bar{\rho})$ if σ is not an abstraction;
- $F([\rho_1, \dots, \rho_n]) = [F(\rho_1), \dots, F(\rho_n)]$.

Definition 29. Let $\sigma, \rho \in \Delta$, we write $\sigma \triangleright \rho$ if $\sigma \Rightarrow_r \Sigma$ and $\rho \in |\Sigma|$.

5.2 Resource vectors

Definition 30 (Resource vectors). We set $R\langle\langle X \rangle\rangle = (R \sqcup \{\perp\})^X$. The set of *resource vectors* is then $R\langle\langle \Delta \rangle\rangle$. Let $u \in R\langle\langle X \rangle\rangle, x \in X$, we denote by u_x the coefficient of u in x .

If a term has a \perp coefficient, it means it is not in the support of the resource vector. We extend the operations of the semiring to \perp in order to extend these operations pointwise to resource vectors: $\forall a \in R \sqcup \{\perp\}, \perp + a = a$, and $a\perp = \perp$. We denote by $\mathbf{0}_\Sigma$ the resource vector whose coefficients are all \perp .

Since we would like to write Taylor expansion as formal series, we define the following set:

Definition 31 (Formal series). Let $R\llbracket \Delta \rrbracket = (R \times \Delta)^{\mathbb{N}} \sqcup \bigsqcup_{n \in \mathbb{N}} (R \times \Delta)^n$ be the set of formal series. We will write an element of this set as $\sum_{i=0}^b a_i \sigma_i$ with $b \in \mathbb{N} \sqcup \{+\infty\}$.

Definition 32 (Summable families). Let $(\sigma_i) \in \Delta^I$. We say this family is *summable* when, for all $\sigma \in \Delta$, the set $\{i \in I \mid \sigma = \sigma_i\}$ is finite.

Definition 33. Let $M = \sum_{i=1}^b a_i \sigma_i$ be a formal series and $\sigma \in \Delta$; let $I_{(\sigma)}^M = \{i \in \mathbb{N} \mid \sigma_i = \sigma\}$. If $I_{(\sigma)}^M$ is finite, we set $M_{(\sigma)}$ as: \perp if $I_{(\sigma)}^M = \emptyset$, $\sum_{i \in I_{(\sigma)}^M} a_i$ otherwise.

Definition 34 (Mapping series to resource vectors).

$$\begin{aligned} \text{canseries} : R\llbracket \Delta \rrbracket &\rightarrow R\langle\langle \Delta \rangle\rangle \\ M &\mapsto \sum_{\sigma \in \Delta} M_{(\sigma)} \sigma \end{aligned}$$

Notice this function is surjective and partial: it is defined only when the associated family is summable. We can then use every function defined on formal series on resource vectors.

Definition 35. We define the *support of a resource vector* as: $\text{Supp}(u) = \{\sigma \in \Delta \mid u_\sigma \neq \perp\}$. We naturally extend this function to formal series by: $\text{Supp}(M) = \text{Supp}(\text{canseries}(M))$.

Definition 36 ([Vau19]). Let $\varphi : R\langle\langle X_1 \rangle\rangle \times \dots \times R\langle\langle X_n \rangle\rangle \rightarrow R\langle\langle Y \rangle\rangle$. We say φ is *n-linear continuous* if for all summable families $(\xi_i^1) \in X_1^{I_1}, \dots, (\xi_i^n) \in X_n^{I_n}$ and for all families of scalars $(a_i^1) \in (R \sqcup \{\perp\})^{I_1}, \dots, (a_i^n) \in (R \sqcup \{\perp\})^{I_n}$, the family $(\varphi(\xi_{i_1}^1, \dots, \xi_{i_n}^n))_{(i_1, \dots, i_n) \in I_1 \times \dots \times I_n}$ is summable and:

$$\varphi \left(\sum_{i_1 \in I_1} a_{i_1}^1 \xi_{i_1}^1, \dots, \sum_{i_n \in I_n} a_{i_n}^n \xi_{i_n}^n \right) = \sum_{i_1, \dots, i_n \in I_1 \times \dots \times I_n} a_{i_1}^1 \dots a_{i_n}^n \varphi(\xi_{i_1}^1, \dots, \xi_{i_n}^n)$$

We naturally extend syntactic constructs and linear substitution by multilinearity on resource vectors:

$$\begin{aligned} \lambda x.u &= \sum_{\sigma \in \Delta} u_\sigma \lambda x.\sigma & \langle u \rangle \bar{v} &= \sum_{\sigma \in \Delta, \bar{\rho} \in !\Delta} u_\sigma \bar{v}_{\bar{\rho}} \langle \sigma \rangle \bar{\rho} \\ [u_1, \dots, u_n] &= \sum_{\sigma_1, \dots, \sigma_n \in \Delta} (u_1)_{\sigma_1} \dots (u_n)_{\sigma_n} [\sigma_1, \dots, \sigma_n] & \partial_x u \cdot \bar{v} &= \sum_{\sigma \in \Delta, \bar{\rho} \in !\Delta} u_\sigma v_{\bar{\rho}} \partial_x \sigma \cdot \bar{\rho} \end{aligned}$$

Notice a multiset of vector is well-defined only if multiplication of R is commutative.

Proposition 12. *Syntactic constructs and linear substitution are multilinear continuous.*

We say a semiring \mathbf{R} *has fractions* when for each $n \in \mathbb{N}$, $\frac{1}{n} \in \mathbf{R}$. In the remaining of this section, we'll assume \mathbf{R} has fractions, as to ensure we can state the following definition.

Definition 37 (Exponential). Let $\sigma \in \Delta$, we define the *exponential* of σ as $\sigma^! = \sum_{n \in \mathbb{N}} \frac{1}{n!} \sigma^n \in \mathbf{R}\langle\langle \Delta \rangle\rangle$, where $\sigma^n = [\sigma, \dots, \sigma]$. Note that, thanks to Proposition 12, this definition can be naturally extended to resource vectors by multilinear-continuity.

Lemma 28 (Compatibility of ∂_x with syntax [Vau19]).

$$\begin{aligned} \partial_x x \cdot \rho^! &= \rho & \partial_x \lambda y. \sigma \cdot \rho^! &= \lambda y. \partial_x \sigma \cdot \rho^! & \partial_x [\sigma_1, \dots, \sigma_n] \cdot \rho^! &= [\partial_x \sigma_1 \cdot \rho^!, \dots, \partial_x \sigma_n \cdot \rho^!] \\ \partial_x y \cdot \rho^! &= y & \partial_x \sigma^! \cdot \rho^! &= (\partial_x \sigma \cdot \rho^!)^! & \partial_x \langle \sigma \rangle \bar{\rho} \cdot \rho^! &= \langle \partial_x \sigma \cdot \rho^! \rangle \partial_x \bar{\rho} \cdot \rho^! \end{aligned}$$

The above equations holds for resource terms, and for resource vector as well.

Definition 38 (Taylor expansion). We define $\tau(M) \in \mathbf{R}\langle\langle \Delta \rangle\rangle$ by induction on M :

$$\begin{aligned} \tau(x) &= x & \tau(\lambda x. M) &= \lambda x. \tau(M) & \tau(MN) &= \langle \tau(M) \rangle \tau(N)^! \\ \tau(\mathbf{0}) &= \mathbf{0}_\Sigma & \tau(aM) &= a\tau(M) & \tau(M + N) &= \tau(M) + \tau(N) \end{aligned}$$

We will write $M \simeq_\tau N$ whenever $\tau(M) = \tau(N)$.

Proposition 13. *The following inclusion holds: $\stackrel{\Delta}{=} \subseteq \simeq_\tau$.*

Proof. Notice \simeq_τ is contextual and that it satisfies the equations of definition 2. □

Remark 2. If $aM \simeq_\tau \mathbf{0}$, then $M \simeq_\tau \mathbf{0}$: our modified version of τ encompasses $\stackrel{\Delta}{=}$ well.

Proposition 14. *The following equality holds: $\tau(S[T/x]) = \partial_x \tau(S) \cdot \tau(T)^!$.*

Proof. Proposition 13 allows us to prove it by induction on $\text{can}(S)$, using Lemma 28. □

Definition 39 (Height and size of a resource term). By induction on resource expressions:

$$\begin{aligned} \mathbf{s}(x) &= 1 & \mathbf{h}(x) &= 1 \\ \mathbf{s}(\lambda x. \sigma) &= 1 + \mathbf{s}(\sigma) & \mathbf{h}(\lambda x. \sigma) &= 1 + \mathbf{h}(\sigma) \\ \mathbf{s}(\langle \sigma \rangle \bar{\rho}) &= 1 + \mathbf{s}(\sigma) + \mathbf{s}(\bar{\rho}) & \mathbf{h}(\langle \sigma \rangle \bar{\rho}) &= 1 + \max(\mathbf{h}(\sigma), \mathbf{h}(\bar{\rho})) \\ \mathbf{s}([\rho_1, \dots, \rho_n]) &= \mathbf{s}(\rho_1) + \dots + \mathbf{s}(\rho_n) & \mathbf{h}([\rho_1, \dots, \rho_n]) &= \max(\mathbf{h}(\rho_1), \dots, \mathbf{h}(\rho_n)) \end{aligned}$$

We also define height on resource vector: $\mathbf{h}(u) = \max\{\mathbf{h}(\sigma) \mid \sigma \in \text{Supp}(u)\}$.

Proposition 15. *For any $M \in \Lambda_R$, $\tau(M)$ is of bounded height.*

Proof. Straightforward induction on raw terms. □

5.3 Resource vector reduction

5.3.1 Closure of resource vectors of bounded height under reduction

Definition 40 (Parallel reduction of bounded height resource vectors). If (σ_i) is a summable family of bounded height, and if for all $i \in \mathbb{N}$, $\sigma_i \rightrightarrows_r \sum_{j=1}^{n_i} \rho_j^i$, then for any family of scalars (a_i) , we say $\text{canseries}(\sum a_i \sigma_i) \Rightarrow_B \text{canseries}(\sum a_i \sum_{j=1}^{n_i} \rho_j^i)$.

In this definition, we do not ask any condition on the summability of (ρ_j^i) , we have to make sure this relation is well defined; we dedicate the rest of this section to prove that fact.

Definition 41. We define the following sets of resource expressions:

- $(!)\mathcal{H}_k = \{\sigma \in (!)\Delta \mid \mathbf{h}(\sigma) \leq k\}$;
- $(!)\mathcal{B}_k = \{\sigma \in (!)\Delta \mid \text{every bag occurring in } \sigma \text{ is of size at most } k\}$;
- $(!)\Delta_E = \{\sigma \in (!)\Delta \mid \text{fv}(\sigma) \subseteq E\}$.

Lemma 29. *Let $\sigma \in \Delta, \{\rho \in \Delta \mid \rho \triangleright \sigma\} \subseteq \mathcal{B}_{\mathbf{s}(\sigma)}$.*

Proof. Let $\mu \notin \mathcal{B}_{\mathbf{s}(\sigma)}$ and $\mu', \mu \triangleright \mu'$. μ has $\langle \nu \rangle [\rho_1, \dots, \rho_n]$ as subterm, with $n > \mathbf{s}(\sigma)$.

Either $\langle \nu \rangle [\rho_1, \dots, \rho_n]$ is still a subterm of μ' , in which case $\mathbf{s}(\mu') > \mathbf{s}(\sigma)$ and $\mu' \neq \sigma$.

Either it was contracted along the reduction, in which case there are two further subcases:

Either there is a contraction that rendered $\mathbf{0}$, hence, $\mu' = \mathbf{0} \neq \sigma$;

Either there was no such contraction, in which case each ρ_i appears at least once in μ' , hence, $\mathbf{s}(\mu') \geq \sum_{i=1}^n \mathbf{s}(\rho_i) \geq n > \mathbf{s}(\sigma)$ and thus, $\mu' \neq \sigma$. \square

Proposition 16. *Let $k, l \in \mathbb{N}, E$ a finite set of variables, then $(!)\mathcal{B}_k \cap (!)\mathcal{H}_l \cap (!)\Delta_E$ is finite.*

Proof. By induction on $l \in \mathbb{N}$. If $l = 1$, $(!)\mathcal{B}_k \cap (!)\mathcal{H}_1 \cap (!)\Delta_E \subseteq (!)E$ which is finite. Let $l \in \mathbb{N}^*$, assume the result holds for l . For each constructor, there is a finite set of possible subterm(s) by I.H. The same argument holds for multisets since their sizes are bounded. \square

Proposition 17. *Let $(\sigma_i) \in \Delta^I, (\Sigma_i) \in \mathbb{N}[\Delta]^I$ be two families such that (σ_i) is summable, of bounded height, and for all $i, \sigma_i \rightrightarrows_r \Sigma_i$, then $(\rho'_j)_{j \in J}$ such that $[\rho'_j \mid j \in J] = \sum_{i \in I} \|\Sigma_i\|$ is summable. The set of resource vectors of bounded height is closed under reduction: If a resource vector of bounded height reduces to another term, then it is of bounded height.*

Proof. Let $\mu \in \Delta$. For any $\sigma \in \Delta$ such that $\sigma \triangleright \mu$, $\text{fv}(\sigma) = \text{fv}(\mu)$. Plus, by Lemma 29, $\{\eta \in \Delta \mid \eta \triangleright \mu\} \subseteq \mathcal{B}_{\mathbf{s}(\mu)}$. Hence, let $l = \max\{\mathbf{h}(\sigma_i) \mid i \in I\}$, we have $\{\eta \in \Delta \mid \eta \triangleright \mu\} \cap \{\sigma_i \mid i \in I\} \subseteq \mathcal{B}_{\mathbf{s}(\mu)} \cap \mathcal{H}_l \cap \Delta_{\text{fv}(\mu)}$, which is finite by Proposition 16, hence, (ρ_j) is summable. \square

Definition 42. We extend full parallel reduction to resource vector of bounded height as follows: $F(v) = \sum_{\sigma \in (!)\Delta} v_\sigma F(\sigma)$.

Proposition 18. *Let $v \in R\langle\langle \Delta \rangle\rangle$ be a bounded height resource vector.*

- $v \Rightarrow_{\mathcal{B}} F(v)$;
- For any $u \in R\langle\langle \Delta \rangle\rangle$ such that $v \Rightarrow_{\mathcal{B}} u$, $u \Rightarrow_{\mathcal{B}} F(v)$.

Proof. One can derive these statements from their equivalent on resource terms. \square

Corollary 6. $\Rightarrow_{\mathcal{B}}$ is strongly confluent.

Proposition 19. *Let $u, u', v, v' \in R\langle\langle \Delta \rangle\rangle$. If $u \Rightarrow_{\mathcal{B}} u'$ and $v \Rightarrow_{\mathcal{B}} v'$, then*

- $\langle u \rangle v^1 \Rightarrow_{\mathcal{B}} \langle u' \rangle v'^1$;
- $\langle \lambda x.u \rangle v^1 \Rightarrow_{\mathcal{B}} \partial_x u' \cdot v'^1$.

Proof. Simple rewriting exercises, writing vectors as formal series. \square

Theorem 5. *Let $\underline{M}, \underline{N} \in \underline{C}_R$. If $\underline{M} \rightrightarrows \underline{N}$, then $\tau(M) \Rightarrow_{\mathcal{B}} \tau(N)$.*

Proof. By induction on the reduction, using Proposition 19 and Proposition 14. \square

5.3.2 Taylor and normalization commute

Must-normalization is quite straightforwardly characterized on Taylor expansions (Corollary 7). Such a study would be much more complicated on may-normalization: how could we distinguish a may-normal form among the Taylor expansion of sums of non-uniform terms?

Proposition 20. *Let $u \in R\langle\langle \Delta \rangle\rangle$ of bounded height. u does not reduce under $\Rightarrow_{\mathcal{B}}$ if and only if for all $\sigma \in \text{Supp}(u)$, σ does not reduce under \rightrightarrows_r .*

Proof. Straightforward from the definitions. \square

Corollary 7. *Taylor expansion of must-normal forms does not reduce under $\Rightarrow_{\mathcal{B}}$.*

Theorem 6. *Let $M \in \Lambda_{\mathcal{R}}$, if M is must-normalizable, then $\tau(\mathbf{NF}(M)) = \mathbf{NF}(\tau(M))$.*

Proof. $M \xrightarrow{\beta^*} \mathbf{NF}(M)$, hence, by Theorem 5, $\tau(M) \Rightarrow_{\mathcal{B}}^* \tau(\mathbf{NF}(M))$ which is normal, and we conclude by confluence of $\Rightarrow_{\mathcal{B}}$. \square

5.3.3 Conservativity on normalizable λ -terms

Definition 43. We denote by \simeq_r the symmetric, transitive and reflexive closure of $\Rightarrow_{\mathcal{B}}$.

Theorem 7. *Let $M, N \in \Lambda$ be must-normalizable. $\tau(M) \simeq_r \tau(N)$ if and only if $M \simeq_r N$.*

Proof. The reverse implication is straightforward from the Theorem 5. For the direct implication, by Theorem 5 and 6, $\tau(M) \Rightarrow_{\mathcal{B}} \tau(\mathbf{NF}(M)) = \mathbf{NF}(\tau(M))$ and $\tau(N) \Rightarrow_{\mathcal{B}} \tau(\mathbf{NF}(N)) = \mathbf{NF}(\tau(N))$. Since $\tau(M) \simeq_r \tau(N)$, $\tau(\mathbf{NF}(M)) = \mathbf{NF}(\tau(M)) = \mathbf{NF}(\tau(N)) = \tau(\mathbf{NF}(N))$ and we conclude by injectivity of τ on pure λ -terms. \square

We can't hope for anything better: indeed, the Taylor expansion is not injective.

Example 1. Terms can "hide" under a 0 coefficient: $\tau(xx + 0(x\mathbf{0})) = \tau(xx)$.

6 Conclusion

We introduced a weak variant of the algebraic λ -calculus, extending the ordinary λ -calculus with the possibility of forming linear combinations of terms, but excluding the structural identity $0M = \mathbf{0}$, so that terms form a weak module instead of a module. We then proved several results, which are mostly analogues of similar properties for the original algebraic λ -calculus: the novelty resides in that they are valid for any semiring, removing the positivity assumption that was needed in the original version.

Namely, we established the confluence of reduction, and showed that the resulting equational theory is conservative over β -equivalence on ordinary λ -terms. Moreover, we studied two notions of normalization, arising naturally when studying sums of terms: we showed that each one could be characterized as typability in a suitable intersection type system. Finally, we introduced a notion of Taylor expansion in that weak setting, and showed that β -reduction could be still simulated; from this we deduced the conservativity of the equational theory induced by normalization of Taylor expansion, on normalizable λ -terms.

The weak algebraic λ -calculus thus enjoys satisfactory operational properties, regardless of the semiring of coefficients. Nonetheless, our results leave a number of open questions and routes for future work.

If one considers the original algebraic λ -calculus as the main object of study, it would be interesting to precise its relationship with the weak version. For instance, one could search for a weak normalization scheme, in the spirit of Alberti's [Alb14], by considering pre-normal forms induced by normalization paths in the weak version: under which condition does that give a meaningful result?

Also, our study of Taylor expansion is limited because of its lack of injectivity. Example 1 seems to indicate that there is no hope to obtain a version of Taylor expansion that is both weak and injective. On the other hand, this also calls for a study of injectivity of ordinary Taylor expansion on the algebraic λ -calculus: more precisely, it is still an open problem to find sufficient conditions on the semiring of coefficients to obtain an injective Taylor expansion, which would be a first step towards a general conservativity result.

Finally, we could also try to extend the study the normalization of Taylor expansion, and especially Theorem 6: how far can we generalize the structure of Böhm trees to obtain a commutation theorem for non-necessarily normalizable terms? A partial answer could be to adapt the results of the second author in the ordinary case [Vau19]. An alternative could be to investigate existing denotational models for weak algebraic λ -calculi: for instance, can we adapt Valiron's work [Val13] to obtain a model in an untyped setting?

References

- [AD17] Pablo ARRIGHI et Gilles DOWEK : Lineal: A linear-algebraic lambda-calculus. *Log. Methods Comput. Sci.*, 13(1), 2017.
- [ADP⁺14] Ali ASSAF, Alejandro DÍAZ-CARO, Simon PERDRIX, Christine TASSON et Benoît VALIRON : Call-by-value, call-by-name and the vectorial behaviour of the algebraic λ -calculus. *Log. Methods Comput. Sci.*, 10(4), 2014.
- [ADV17] Pablo ARRIGHI, Alejandro DÍAZ-CARO et Benoît VALIRON : The vectorial λ -calculus. *Inf. Comput.*, 254:105–139, 2017.
- [Alb14] Michele ALBERTI : *On operational properties of quantitative extensions of λ -calculus*. Thesis, Aix Marseille Université ; Università di Bologna, décembre 2014.
- [BKV17] Antonio BUCCIARELLI, Delia KESNER et Daniel VENTURA : Non-idempotent intersection types for the Lambda-Calculus. *Logic Journal of the IGPL*, 25(4):431–464, 07 2017.
- [dC18] Daniel de CARVALHO : Execution time of λ -terms via denotational semantics and intersection types. *Math. Struct. Comput. Sci.*, 28(7):1169–1203, 2018.
- [Ehr05] Thomas EHRHARD : Finiteness spaces. *Math. Struct. Comput. Sci.*, 15(4):615–646, 2005.
- [Ehr10] Thomas EHRHARD : A finiteness structure on resource terms. In *Proceedings of the 25th Annual IEEE Symposium on Logic in Computer Science, LICS 2010, 11-14 July 2010, Edinburgh, United Kingdom*, pages 402–410. IEEE Computer Society, 2010.
- [Ehr23a] Thomas EHRHARD : A coherent differential PCF. *Log. Methods Comput. Sci.*, 19(4), 2023.
- [Ehr23b] Thomas EHRHARD : Coherent differentiation. *Math. Struct. Comput. Sci.*, 33(4-5):259–310, 2023.
- [ER03] Thomas EHRHARD et Laurent REGNIER : The differential lambda-calculus. *Theoretical Computer Science*, 309(1):1–41, 2003.
- [ER06] Thomas EHRHARD et Laurent REGNIER : Böhm trees, krivine’s machine and the taylor expansion of lambda-terms. In Arnold BECKMANN, Ulrich BERGER, Benedikt LÖWE et John V. TUCKER, éditeurs : *Logical Approaches to Computational Barriers, Second Conference on Computability in Europe, CiE 2006, Swansea, UK, June 30-July 5, 2006, Proceedings*, volume 3988 de *Lecture Notes in Computer Science*, pages 186–197. Springer, 2006.
- [ER08] Thomas EHRHARD et Laurent REGNIER : Uniformity and the taylor expansion of ordinary lambda-terms. *Theor. Comput. Sci.*, 403(2-3):347–372, 2008.
- [EW23] Thomas EHRHARD et Aymeric WALCH : Cartesian coherent differential categories. In *38th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2023, Boston, MA, USA, June 26-29, 2023*, pages 1–13. IEEE, 2023.
- [Gir88] Jean-Yves GIRARD : Normal functors, power series and λ -calculus. *Ann. Pure Appl. Log.*, 37(2):129–177, 1988.
- [Kri93] J. L. KRIVINE : *Lambda-calculus, types and models*. Ellis Horwood, USA, 1993.

- [KV23] Axel KERINEC et Lionel VAUX AUCLAIR : The algebraic λ -calculus is a conservative extension of the ordinary lambda-calculus. *HOR 2023 - 11th International Workshop on Higher-Order Rewriting*, 2023.
- [Lev19] Thomas LEVENTIS : A deterministic rewrite system for the probabilistic λ -calculus. *Math. Struct. Comput. Sci.*, 29(10):1479–1512, 2019.
- [PTV16] Michele PAGANI, Christine TASSON et Lionel VAUX : Strong normalizability as a finiteness structure via the Taylor expansion of lambda-terms. In Bart JACOBS et Christof LODING, éditeurs : *Foundations of Software Science and Computation Structures*, pages 408–423, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.
- [Val13] Benoît VALIRON : A typed, algebraic, computational lambda-calculus. *Math. Struct. Comput. Sci.*, 23(2):504–554, 2013.
- [Vau09] Lionel VAUX : The algebraic lambda calculus. *Mathematical Structures in Computer Science*, 19(5):1029–1059, 2009.
- [Vau19] Lionel VAUX : Normalizing the Taylor expansion of non-deterministic λ -terms, via parallel reduction of resource vectors. *Logical Methods in Computer Science*, Volume 15, Issue 3, juillet 2019.
- [vT04] André van TONDER : A lambda calculus for quantum computation. *SIAM J. Comput.*, 33(5):1109–1135, 2004.