Unifying Graded Linear Logic and Differential Operators

Flavien Breuvart
LIPN, USPN, Villetaneuse, France

Marie Kerjean
CNRS, LIPN, USPN, Villetaneuse, France

Simon Mirwasser
LIPN, USPN, Villetaneuse, France

Abstract

Linear Logic refines Classical Logic by taking into account the resources used during the proof and program computation. In the past decades, it has been extended to various frameworks. The most famous are indexed linear logics which can describe the resource management or the complexity analysis of a program. From another perspective, Differential Linear Logic is an extension which allows the linearization of proofs. In this article, we merge these two directions by first defining a differential version of Graded linear logic: this is made by indexing exponential connectives with a monoid of differential operators. We prove that it is equivalent to a graded version of previously defined extension of finitary differential linear logic. We give a denotational model of our logic, based on distribution theory and linear partial differential operators with constant coefficients.

2012 ACM Subject Classification Theory of computation → Linear logic; Mathematics of computing → Partial differential equations; Theory of computation → Denotational semantics

Keywords and phrases Linear Logic, Denotational Semantics, Functional Analysis, Graded Logic

Digital Object Identifier 10.4230/LIPIcs.FSCD.2023.17

Funding *Flavien Breuvart*: Supported by the project ANR-18-CE25-0001
Marie Kerjean: Supported by the projects ANR-20-CE48-0002 and ANR-20-CE48-0014

Acknowledgements We thank the anonymous reviewers for their kind and constructive comments on the first version of the paper, which greatly contributed to its improvement.

1 Introduction

Linear logic (LL) [20] and its differential counterpart [14] give a framework to study resource usages of proofs and programs. These logics were invented by enriching the syntax of proofs with new constructions observed in denotational models of λ-calculus [21, 11]. The exponential connective ! introduces non-linearity in the context of linear proofs and encapsulate the notion of resource usage. This notion was refined into *parametrised exponentials* [22, 13, 17, 19], where exponential connectives are indexed by annotations specifying different behaviors. Our aim here is to follow Kerjean's former works [25] by indexing formulas of Linear Logic with Differential Operators. Thanks to the setting of Bounded Linear Logic, we formalize and deepen the connection between Differential Linear Logic and Differential Operators.

The fundamental linear decomposition of LL is the decomposition of the usual non-linear implication $\Rightarrow$ into a linear one $\multimap$ from a set of resources represented by the new connective !: $(A \Rightarrow B) \equiv (!A \multimap B)$. Bounded Linear Logic (BLL) [22] was introduced as the first attempt to use typing systems for complexity analysis. But our interest for this logic stems from the fact that it extends LL with several exponential connectives which are indexed by *polynomially bounded intervals*. Since then, some other indexations of LL have been developed for many purposes, for example IndLL [13] where the exponential modalities are indexed by some

8th International Conference on Formal Structures for Computation and Deduction (FSCD 2023).
Editors: Marco Gaboardi and Femke van Raamsdonk; Article No. 17; pp. 17:1–17:21
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

functions, or the graded logic $\mathsf{B}_{\mathcal{S}}\mathsf{LL}$ [6, 19, 29] where they are indexed by the elements of a semiring $\mathcal{S}$. This theoretical development finds applications in programming languages [1, 16].

Differential linear logic [14] (DiLL) consists in an a priori distinct approach to linearity, and is based on the denotational semantics of linear proofs in terms of linear functions. In the syntax of LL, the dereliction rule states that if a proof is linear, one can then forget its linearity and consider it as non-linear. To capture differentiation, DiLL is based on a codereliction rule which is the syntactical opposite of the dereliction. It states that from a non-linear proof (or a non-linear function) one can extract a linear approximation of it, which, in terms of functions, is exactly the differential (one can notice that here, the analogy with resources does not work). Then, models of DiLL interpret the codereliction by different kinds of differentiation [10, 3].

A first step towards merging the graded and the differential extension of LL was made by Kerjean in 2018 [25]. In this paper, she defines an extension of DiLL, named D-DiLL, in which the exponential connectives ? and ! are indexed with a *fixed* linear partial differential operator with constant coefficients (LPDOcc) D. There, formulas $!_D A$ and $?_D A$ are respectively interpreted in a denotational model as spaces of functions or distributions which are solutions of the differential equation induced by D. The dereliction and codereliction rules then represent respectively the resolution of a differential equation and the application of a differential operator. This is a significant step forward in our aim to make the theory of programming languages and functional analysis closer, with a Curry-Howard perspective. In this work, we will generalize D-DiLL to a logic indexed by a monoid of LPDOcc.

Contributions. This work considerably generalizes, corrects and consolidates the extention of DiLL to differential operators sketched in [25]. It extends D-DiLL in the sense that the logic is now able to deal with all LPDOcc and combine their action. It corrects D-DiLL as the denotational interpretation of indexed exponential $?_D$ and $!_D$ are changed, leaving the interpretation of *inference rules* unchanged but reversing their type in a way that is now compatible with graded logics. Finally, this work consolidates D-DiLL by proving a cut-elimination procedure in the graded case, making use of an algebraic property on the monoid of LPDOcc.

Outline. We begin this paper in Section 2 by reviewing Differential Linear Logic and its semantics in terms of functions an distributions. We also recall the definition of $\mathsf{B}_{\mathcal{S}}\mathsf{LL}$. Section 3 focuses on the definition of an extension of $\mathsf{B}_{\mathcal{S}}\mathsf{LL}$, where we construct a finitary differential version for it and prove a cut-elimination theorem. The cut-elimination procedure mimicks partly the one of DiLL or $\mathsf{B}_{\mathcal{S}}\mathsf{LL}$, but also deals with completely new interactions with inference rules. Then, Section 4 generalizes D-DiLL into a framework with several indexes and shows that it corresponds to our finitary differential $\mathsf{B}_{\mathcal{S}}\mathsf{LL}$ indexed by a monoid of LPDOcc. It formally constructs a denotational model for it. This gives in particular a new semantics for $\mathsf{B}_{\mathcal{S}}\mathsf{LL}$. Finally, Section 5 discusses the addition of an indexed promotion to differential $\mathsf{B}_{\mathcal{S}}\mathsf{LL}$ and possible definitions for a semiring of differential operators.

2 Linear logic and its extensions

Linear Logic refines Classical Logic by introducing a notion of linear proofs. Formulas are defined according to the following grammar (omitting neutral elements which do not play a role here):

$$A, B := A \otimes B \mid A \invamp B \mid A \,\&\, B \mid A \oplus B \mid ?A \mid !A \mid \cdots .$$

The linear negation $(\_)^{\perp}$ of a formula is *defined* on the syntax and is involutive, with in particular $(!A)^\perp := ?(A)^\perp$. The connector ! enjoys structural rules, respectively called weakening w, contraction c, dereliction d and promotion p:

$$\frac{\Gamma \vdash \Delta}{\Gamma, !A \vdash \Delta}\,\mathsf{w} \qquad \frac{\Gamma, !A, !A \vdash \Delta}{\Gamma, !A \vdash \Delta}\,\mathsf{c} \qquad \frac{\Gamma, A \vdash \Delta}{\Gamma, !A \vdash \Delta}\,\mathsf{d} \qquad \frac{!\Gamma \vdash A}{!\Gamma \vdash !A}\,\mathsf{p}$$

These structural rules can be understood in terms of resources: a proof of $A \vdash B$ uses exactly once the hypothesis A while a proof of $!A \vdash B$ might use A an arbitrary number of times. Notice that the dereliction allows to forget the linearity of a proof by making it non-linear.

▶ Remark 1. The exponential rules for LL are recalled here in a two-sided flavour, making their denotational interpretation in Section 2.1 easier. However, we always consider a *classical* sequent calculus, and the new $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$ will be introduced later in a one-sided flavour to lightens the formalism.

Differentiation is then introduced through a "codereliction" rule $\bar{\mathsf{d}}$, which is symmetrical to d and allows to linearize a non-linear proof [14]. To express the cut-elimination with the promotion rule, other costructural rules are needed, which find a natural interpretation in terms of differential calculus. Note that the first version of DiLL, called DiLL_0, does not feature the promotion rule, which was introduced in later versions [30]. The exponential rules of DiLL_0 are then $\mathsf{w}, \mathsf{c}, \mathsf{d}$ with the following coweakening $\bar{\mathsf{w}}$, cocontraction $\bar{\mathsf{c}}$ and codereliction $\bar{\mathsf{d}}$ rules, given here in a one-sided flavour.

$$\frac{}{\vdash !A}\,\bar{\mathsf{w}} \qquad \frac{\vdash \Gamma, !A \qquad \vdash \Delta, !A}{\vdash \Gamma, \Delta, !A}\,\bar{\mathsf{c}} \qquad \frac{\vdash \Gamma, A}{\vdash \Gamma, !A}\,\bar{\mathsf{d}}$$

In the rest of the paper, as a support for the semantical interpretation of DiLL, we denote by $D_a(f)$ the differential of a function f at a point a, that is:

$$D_a f : v \mapsto \lim_{h \to 0} \frac{f(a + hv) - f(a)}{h}$$

2.1 Distribution theory as a semantical interpretation of DiLL

DiLL originates from vectorial refinements of models of LL [11], which mainly keep their discrete structure. However, the exponential connectives and rules of DiLL can also be understood as operations on smooth functions or distributions [31]. In the whole paper, $(\_)' := \mathcal{L}(\_, \mathbb{R})$ is the dual of a (topological) vector space, and *distributions with compact support* are by definition linear continuous maps on the space of smooth scalar maps, that is elements of $(\mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R}))'$. Distributions are sometimes described as "generalized functions"[1] Let us recall the notation for Dirac operator, which is a distribution with compact support and used a lot in the rest of the paper: $\delta : v \in \mathbb{R}^n \mapsto (f \mapsto f(v)) \in (\mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R}))'$.

Recently, Kerjean [25] gave an interpretation of the connective ? by a space of smooth scalar functions, while ! is interpreted as the space of linear maps acting on those functions, that is a space of *distributions*:

$$[\![?A]\!] := \mathcal{C}^\infty([\![A]\!]', \mathbb{R}) \qquad\qquad [\![!A]\!] := \mathcal{C}^\infty([\![A]\!], \mathbb{R})'.$$

Note that the language of distributions applies to all models of DiLL as noticed by Ehrhard on Köthe spaces [10]. The focus of this model was to find smooth infinite dimensional models

[1] Indeed, any function with compact support $g \in \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R})$ acts as a distribution $T_g \in (\mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R}))'$ with compact support, through integration: $T_g : f \mapsto \int gf$. It is indeed a distribution, as it acts linearly (and continuously) on smooth functions.

of DiLL, whose objects were invariant under double negation, that is a model of *classical* DiLL. This is an intricate issue, see [8], and a simple solution is to consider models of *polarized* calculus. Polarized Linear Logic LL_{pol} [27] separates formulas in two classes:

$$\begin{array}{rl} \text{Negative Formulas: } & N, M := a \mid ?P \mid \uparrow P \mid N \parr M \mid \bot \mid N \,\&\, M \mid \top. \\ \text{Positive Formulas: } & P, Q := a^{\perp} \mid !N \mid \downarrow N \mid P \otimes Q \mid 0 \mid P \oplus Q \mid 1. \end{array}$$

We interpret formulas of LL_{pol} by Nuclear topological vector spaces, and add the condition that the spaces are Fréchet or DF according to the polarity of the formulas. Positive formulas (left stable by $\otimes$!) are interpreted as Nuclear DF spaces while Negative formulas (left stable by $\parr$?) are interpreted by Nuclear Fréchet spaces. We will not dive into the details of these definitions, see [24] for more details, but the reader should keep in mind that the formulas are always interpreted as *reflexive topological vector spaces*, that is spaces E which are isomorphic to their double dual E''. The model of functions and distribution is thus a model of *classical* DiLL, in which $[\![(\_)^{\perp}]\!] := (\_)'$.

Nicely, every exponential rule of DiLL has an interpretation in terms of functions and distributions, through the following natural transformations. In the whole paper, E and F denote topological vector spaces, which will represent the interpretation $[\![A]\!]$ and $[\![B]\!]$ of formulas A, B of DiLL. For the sake of readability, we will denote the natural transformations (*e.g.* $\mathsf{d}, \bar{\mathsf{d}}$) by the same label as the deriving rule they interpret, and likewise for connectors (*e.g.* $?, \otimes, !$) and their associated functors.

- The weakening $\mathsf{w} : \mathbb{R} \to ?E$ maps $1 \in \mathbb{R}$ to the constant function at 1, while the coweakening $\bar{\mathsf{w}} : \mathbb{R} \to !E$ maps $1 \in \mathbb{R}$ to Dirac distribution at 0: $\delta_0 : f \mapsto f(0)$.
- The dereliction $\mathsf{d} : E' \to ?(E')$ maps a linear function to itself while the codereliction $\bar{\mathsf{d}} : E \to !E$ maps a vector v to the distribution mapping a function to its differential at 0 according to the vector v :
$$\mathsf{d} : \ell \mapsto \ell \qquad \bar{\mathsf{d}} : v \mapsto (D_0(\_)(v) : f \mapsto D_0(f)(v))\,.$$
- The contraction $\mathsf{c} : ?E \otimes ?E \to ?E$ maps two scalar functions f, g to their scalar multiplication $f.g$ while the cocontraction $\bar{\mathsf{c}} : !E \otimes !E \to !E$ maps two distributions ψ and ϕ to their *convolution product* $\psi * \phi : f \mapsto \psi\,(x \mapsto \phi(y \mapsto f(x+y)))$, which is a commutative operation over distributions.

These interpretations are natural, while trying to give a semantics of a model with smooth functions and distributions. The dereliction is the one from LL, and the codereliction is the differentiation at 0, which is what differential linear logic provides. The fact that the contraction is interpreted by the scalar product comes from the kernel theorem, and the weakening is the neutral element for this operation. The cocontraction is interpreted by the convolution product, as the natural monoidal operation on distributions, with its neutral element to interpret the coweakening: the dirac operator at 0.

The natural transformations $\mathsf{w}, \bar{\mathsf{w}}, \mathsf{d}, \bar{\mathsf{d}}$ can also be directly constructed from the biproduct on topological vector spaces and Schwartz' Kernel Theorem expressing Seely isomorphisms.

2.2 Differential operators as an extension of DiLL

A first advance in merging the graded and the differential extensions of LL was made by Kerjean in 2018 [25]. In this paper, she defines an extension of DiLL named D-DiLL. This logic is based on a *fixed single* linear partial differential operator D, which appears as a single index in exponential connectives $!_D$ and $?_D$.

The abstract interpretation of ? and ! as spaces of functions and distributions respectively allows to generalize them to spaces of solutions and parameters of differential equations. To do so, we generalize the action of $D_0(\_)$ in the interpretation of $\bar{\mathsf{d}}$ to another differential operator D. The interpretation of $\bar{\mathsf{d}}$ then corresponds to the application of a differential operator while the interpretation of d corresponds to the resolution of a differential equation (which is ℓ itself when the equation is $D_0(\_) = \ell$, but this is specifically due to the involutivity of D_0).

In D-DiLL, the exponential connectives can be indexed by a fixed differential operator. It admits a denotational semantics for a specific class of those, whose resolution is particularly easy thanks to the existence of a fundamental solution. A *Linear Partial Differential Operator with constant coefficients (LPDOcc)* acts linearly on functions $f \in \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R})$, and by duality acts also on distributions. In what follows, each a_α will be an element of $\mathbb{R}$. By definition, only a finite number of such a_α are non-zero.

$$D : f \mapsto \left(z \mapsto \sum_{\alpha \in \mathbb{N}^n} a_\alpha \frac{\partial^{|\alpha|} f}{\partial x^\alpha}(z) \right) \qquad \hat{D} : f \mapsto \left(z \mapsto \sum_{\alpha \in \mathbb{N}^n} (-1)^{|\alpha|} a_\alpha \frac{\partial^{|\alpha|} f}{\partial x^\alpha}(z) \right) \tag{1}$$

▶ Remark 2. The coefficients $(-1)^{|\alpha|}$ in equation 1 originates from the intuition of distributions as generalized functions. With this intuition, it is natural to want that for each smooth function f, $D(T_f) = T_{D(f)}$, where T_f stands for the distribution generalizing the function f. When computing $T_{D(f)}$ on a function g with partial integration one shows that $T_{D(f)}(g) = \int D(f) g = \int f(\hat{D}(g)) = T_f \circ \hat{D}$, hence the definition.

We make D act on distributions through the following equation:

$$D(\phi) := \left(\phi \circ \widehat{D} : f \mapsto \phi(\widehat{D}(f)) \right) \in \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R})'.$$

▶ **Definition 3.** *Let D be a LPDOcc. A* fundamental solution *of D is a distribution $\Phi_D \in \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R})'$ such that $D(\Phi_D) = \delta_0$.*

▶ **Proposition 4** (Hormander, 1963). *LPDOcc* distribute over convolution*, meaning that $D(\phi * \psi) = D(\phi) * \psi = \phi * D(\psi)$ for any $\phi, \psi \in\, !E$.*

The previous proposition is easy to check and means that knowing the fundamental solution of D gives access to the solution $\psi * \Phi_D$ of the equation $\hat{D}(\_) = \psi$. It is also the reason why indexation with several differential operators is possible. Luckily for us, LPDOcc are particularly well-behaved and always have a fundamental solution. The proof of the following well-known theorem can for example be found in [23, 3.1.1].

▶ **Theorem 5** (Malgrange-Ehrenpreis). *Every linear partial differential operator with constant coefficients admits exactly one fundamental solution.*

Using this result, D-DiLL gives new definitions for d and $\bar{\mathsf{d}}$, depending of a LPDOcc D:

$$\mathsf{d}_D : f \mapsto \Phi_D * f \qquad \bar{\mathsf{d}}_D : \phi \mapsto \phi \circ D.$$

These new definitions came from the following ideas. Through the involutory duality, each $v \in E$ corresponds to a unique $\delta_v \in E'' \simeq E$, and $\bar{\mathsf{d}}_D$ is then interpreted as $\phi \in E'' \mapsto \phi \circ D_0$. Then Kerjean considered that $E'' = (D_0(?(E'), \mathbb{R}))'$ and generalized it by replacing D_0 with D, defining $?_D E := D(\mathcal{C}^\infty(E', \mathbb{R}))$. This gave types $\mathsf{d}_D : ?_D E' \to ?E'$ and $\bar{\mathsf{d}}_D : !_D E \to !E$.

The reader should note that these definitions only work for finite dimensional vector spaces: one is able to apply a LPDOcc to a smooth function from $\mathbb{R}^n$ to $\mathbb{R}$ using partial differentiation on each dimension, but this is completely different if the function has an infinite dimensional domain. The exponential connectives indexed by a LPDOcc therefore only apply to *finitary* formulas: that are the formulas with no exponentials.

2.3 Indexed linear logics: resources, effects and coeffects

Since Girard's original BLL [22], several systems have implemented indexed exponentials to keep track of resource usage [9, 15]. More recently, several authors [19, 17, 6] have defined a modular (but a bit less expressive) version $\mathsf{B}_{\mathcal{S}}\mathsf{LL}$ where the exponentials are indexed (more specifically "graded", as in graded algebras) by elements of a given semiring $\mathcal{S}$.

▶ **Definition 6.** *A* semiring $(\mathcal{S}, +, 0, \times, 1)$ *is given by a set $\mathcal{S}$ with two associative binary operations on $\mathcal{S}$: a sum $+$ which is commutative and has a neutral element $0 \in \mathcal{S}$ and a product $\times$ which is distributive over the sum and has a neutral element $1 \in \mathcal{S}$.*
Such a semiring is said to be commutative *when the product is commutative.*
An ordered semiring *is a semiring endowed with a partial order $\leq$ such that the sum and the product are monotonic.*

This type of indexation, named *grading*, has been used in particular to study effects and coeffects, as well as resources [6, 5, 17]. The main feature is to use this grading in a type system where some types are indexed by elements of the semiring. This is exactly what is done in the logic $\mathsf{B}_{\mathcal{S}}\mathsf{LL}$, where $\mathcal{S}$ is an ordered semiring. The exponential rules of $\mathsf{B}_{\mathcal{S}}\mathsf{LL}$ are adapted from those of LL, and agree with the intuitions that the index x in $!_x A$ is a witness for the usage of resources of type A during the proof/program.

$$\frac{\Gamma \vdash B}{\Gamma, !_0 A \vdash B}\,\mathsf{w} \qquad \frac{\Gamma, !_x A, !_y A \vdash B}{\Gamma, !_{x+y} A \vdash B}\,\mathsf{c} \qquad \frac{\Gamma, A \vdash B}{\Gamma, !_1 A \vdash B}\,\mathsf{d} \qquad \frac{!_{x_1} A_1, \dots, !_{x_n} A_n \vdash B}{!_{x_1 \times y} A_1, \dots, !_{x_n \times y} A_n \vdash !_y B}\,\mathsf{p}$$

Finally, a subtyping rule is also added, which uses the order of $\mathcal{S}$. In Section 3, we will use an order induced by the additive rule of $\mathcal{S}$, and this subtyping rule will stand for a generalized dereliction.

$$\frac{\Gamma, !_x A \vdash B \quad x \leq y}{\Gamma, !_y A \vdash B}\,\mathsf{d}_I$$

3 A differential $\mathsf{B}_{\mathcal{S}}\mathsf{LL}$

In this section, we extend a graded linear logic with indexed coexponential rules. We define and prove correct a cut-elimination procedure.

Formulas and proofs

We define a differential version of $\mathsf{B}_{\mathcal{S}}\mathsf{LL}$ by extending its set of exponential rules. Here, we will restrict ourselves to a version without promotion, as it has been done for DiLL originally. Following the ideas behind DiLL, we add *costructural* exponential rules: a coweakening $\bar{\mathsf{w}}$, a cocontraction $\bar{\mathsf{c}}$, an indexed codereliction $\bar{\mathsf{d}}_I$ and a codereliction $\bar{\mathsf{d}}$. The set of exponential rules of our new logic $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$ is given in Figure 1. Note that by doing so we study a *classical* version of $\mathsf{B}_{\mathcal{S}}\mathsf{LL}$, with an involutive linear duality.

▶ Remark 7. In $\mathsf{B}_{\mathcal{S}}\mathsf{LL}$, we consider a semiring $\mathcal{S}$ as a set of indices. With $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$, we do not need a semiring: since this is a promotion-free version, only one operation (the sum) is important. Hence, in $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$, $\mathcal{S}$ will only be a monoid. This modification requires two precisions:

- The indexed dereliction uses the fact that $\mathcal{S}$ is an ordered semiring. Here, the order will *always* be defined through the sum: $\forall x, y \in \mathcal{S},\ x \leq y \iff \exists x' \in \mathcal{S},\ x + x' = y$. This is due to the fact that for compatiblity with coexponential rules, we always need $\forall x, 0 \leq x$.

$$\frac{\vdash \Gamma}{\vdash \Gamma, ?_0 A}\,\mathsf{w} \qquad \frac{\vdash \Gamma, ?_x A, ?_y A}{\vdash \Gamma, ?_{x+y} A}\,\mathsf{c} \qquad \frac{\vdash \Gamma, ?_x A \qquad x \leq y}{\vdash \Gamma, ?_y A}\,\mathsf{d}_I \qquad \frac{\vdash \Gamma, A}{\vdash \Gamma, ?A}\,\mathsf{d}$$

$$\frac{}{\vdash !_0 A}\,\bar{\mathsf{w}} \qquad \frac{\vdash \Gamma, !_x A \qquad \vdash \Delta, !_y A}{\vdash \Gamma, \Delta, !_{x+y} A}\,\bar{\mathsf{c}} \qquad \frac{\vdash \Gamma, !_x A \qquad x \leq y}{\vdash \Gamma, !_y A}\,\bar{\mathsf{d}}_I \qquad \frac{\vdash \Gamma, A}{\vdash \Gamma, !A}\,\bar{\mathsf{d}}$$

Figure 1 Exponential rules of $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$

- In $\mathsf{B}_{\mathcal{S}}\mathsf{LL}$, the dereliction is indexed by 1, the neutral element of the product. In $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$, we will remove this index since we do not have a product operation and simply use $!$ and $?$ instead of $!_1$ and $?_1$.

Since every element of $\mathcal{S}$ is greater than 0, we have two admissible rules which will appear in the cut elimination procedure: an indexed weakening w_I and an indexed coweakening $\bar{\mathsf{w}}_I$:

$$\frac{\vdash \Gamma}{\vdash \Gamma, ?_x A}\,\mathsf{w}_I \quad := \quad \frac{\dfrac{\vdash \Gamma}{\vdash \Gamma, ?_0 A}\,\mathsf{w}}{\vdash \Gamma, ?_x A}\,\mathsf{d}_I \qquad\qquad \frac{}{\vdash !_x A}\,\bar{\mathsf{w}}_I \quad := \quad \frac{\dfrac{}{\vdash !_0 A}\,\bar{\mathsf{w}}}{\vdash !_x A}\,\bar{\mathsf{d}}_I \,.$$

Definition of the cut elimination procedure

Since this work is done with a Curry-Howard perspective, a crucial point is the definition of a cut-elimination procedure. The cut rule is the following one

$$\frac{\vdash \Gamma, A \qquad \vdash A^{\perp}, \Delta}{\vdash \Gamma, \Delta}\,\mathit{cut}$$

which represents the composition of proofs/programs. Defining its elimination, corresponds to express explicitly how to rewrite a proof with cuts into a proof without any cut. It represents exaclty the calculus of our logic.

In order to define the cut elimination procedure of $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$, we have to consider the cases of cuts after each costructural rule that we have been introduced, since the cases of cuts after MALL rules or after w, c, d_I and d are already known. An important point is that we will use the formerly introduced indexed (co)weakening rather than the usual one.

Before giving the formal rewriting of each case, we will divide them into three groups. Since $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$ is highly inspired from DiLL, one can try to adapt the cut-elimination procedure from DiLL. This adaptation would mean that the structure of the rewriting is exactly the same, but the exponential connectives have to be indexed. For most cases, this method works and there is exactly one possible way to index these connectives, since w_I, $\bar{\mathsf{w}}_I$, c, $\bar{\mathsf{c}}$, d and $\bar{\mathsf{d}}$ do not require a choice of the index (at this point, one can think that there is a choice in the indexing of w_I and $\bar{\mathsf{w}}_I$, but this is a forced choice thanks to the other rules).

However, the case of the cut between a contraction and a cocontraction will require some work on the indexes because these two rules use the addition of the monoid. The index of the principal formula x (resp. x') of a contraction (resp. cocontraction) rule is the sum of two indexes x_1 and x_2 (resp. x_3 and x_4). But $x{=}x'$ does not imply that $x_1{=}x_3$ and $x_2{=}x_4$. We will then have to use a technical algebraic notion to decorate the indexes of the cut elimination between c and $\bar{\mathsf{c}}$ in DiLL: the additive splitting.

▶ **Definition 8.** *A monoid $(\mathcal{M}, +, 0)$ is* additive splitting *if for each $x_1, x_2, x_3, x_4 \in \mathcal{M}$ such that $x_1 + x_2 = x_3 + x_4$, there are elements $x_{1,3},\ x_{1,4},\ x_{2,3},\ x_{2,4} \in \mathcal{M}$ such that*

$$x_1 = x_{1,3} + x_{1,4} \qquad x_2 = x_{2,3} + x_{2,4} \qquad x_3 = x_{1,3} + x_{2,3} \qquad x_4 = x_{1,4} + x_{2,4}.$$

This notion appears in [5], for describing particular models of $\mathsf{B}_{\mathcal{S}}\mathsf{LL}$, based on the relational model. Here the purpose is different: it appears from a syntactical point of view. In the rest of this section, we will not only require $\mathcal{S}$ to be a monoid, but to be additive splitting as well.

Now that we have raised some fundamental difference in a possible cut-elimination procedure, one can note that we do not have mentioned how to rewrite the cuts following an indexed (co)dereliction. This is because the procedure from DiLL cannot be adapted at all in order to eliminate those cuts, as d_I and $\bar{\mathsf{d}}_I$ have nothing in common with the exponential rules of DiLL. The situation is even worse: these cuts cannot be eliminated since these rules are not deterministic because of the use of the order relation. These considerations lead to the following division between the cut elimination cases.

Group 1: The cases where DiLL can naively be decorated. These will be cuts involving two exponential rules, with at least one being an indexed (co)weakening or a non-indexed (co)dereliction.

Group 2: The case where DiLL can be adapted using algebraic technicality, which is the cut between a contraction and a cocontraction.

Group 3: The cases highly different from DiLL. Those are the ones involving an indexed dereliction or an indexed codereliction.

The formal rewritings for the cases of groups 1 and 2 are given in Figure 2. The cut-elimination for contraction and a cocontraction uses the additive splitting property with the notations of Definition 8.

Finally, the last possible case of an occurrence of a cut in a proof is the one where d_I or $\bar{\mathsf{d}}_I$ is applied before the cut: the group 3. The following definition introduces rewritings where these rules go up in the derivation tree, and which will be applied before the cut elimination procedure. This technique is inspired from subtyping ideas, which make sense since d_I is originally defined as a subtyping rule.

▶ **Definition 9.** *The rewriting procedures $\rightsquigarrow_{\mathsf{d}_I}$ and $\rightsquigarrow_{\bar{\mathsf{d}}_I}$ are defined on proof trees of $DB_{\mathcal{S}}LL$.*

1. *When d_I (resp. $\bar{\mathsf{d}}_I$) is applied after a rule r and r is either from MALL (except the axiom) or r is $\bar{\mathsf{w}}_I$, $\bar{\mathsf{c}}$, $\bar{\mathsf{d}}_I$ (resp. w_I, c, d_I), $\bar{\mathsf{d}}$ or d, the rewriting $\rightsquigarrow_{\mathsf{d}_I,1}$ (resp. $\rightsquigarrow_{\bar{\mathsf{d}}_I,1}$) exchanges r and d_I (resp. $\bar{\mathsf{d}}_I$) which is possible since r and d_I do not have the same principal formula.*
2. *When d_I or $\bar{\mathsf{d}}_I$ is applied after a (co)contraction, the rewriting is*

$$\dfrac{\dfrac{\overset{\Pi}{\vdash \Gamma, ?_{x_1}A, ?_{x_2}A}}{\vdash \Gamma, ?_{x_1+x_2}A}\,\mathsf{c}}{\vdash \Gamma, ?_{x_1+x_2+x_3}A}\,\mathsf{d}_I \quad\rightsquigarrow_{\mathsf{d}_I,2}\quad \dfrac{\dfrac{\dfrac{\overset{\Pi}{\vdash \Gamma, ?_{x_1}A, ?_{x_2}A}}{\vdash \Gamma, ?_{x_1+x_2}A}\,\mathsf{c}}{\vdash \Gamma, ?_{x_1+x_2}A, ?_{x_3}A}\,\mathsf{w}_I}{\vdash \Gamma, ?_{x_1+x_2+x_3}A}\,\mathsf{c}$$

$$\dfrac{\dfrac{\overset{\Pi_1}{\vdash \Gamma, !_{x_1}A} \qquad \overset{\Pi_2}{\vdash \Delta, !_{x_2}A}}{\vdash \Gamma, \Delta, !_{x_1+x_2}A}\,\bar{\mathsf{c}}}{\vdash \Gamma, \Delta, !_{x_1+x_2+x_3}A}\,\bar{\mathsf{d}}_I \quad\rightsquigarrow_{\bar{\mathsf{d}}_I,2}\quad \dfrac{\dfrac{\overset{\Pi_1}{\vdash \Gamma, !_{x_1}A} \qquad \overset{\Pi_2}{\vdash \Delta, !_{x_2}A}}{\vdash \Gamma, \Delta, !_{x_1+x_2}A}\,\bar{\mathsf{c}} \qquad \dfrac{\vdash}{\vdash !_{x_3}A}\,\bar{\mathsf{w}}_I}{\vdash \Gamma, \Delta, !_{x_1+x_2+x_3}A}\,\bar{\mathsf{c}}$$

3. *If it is applied after an indexed (co)weakening, the rewriting is*

$$\dfrac{\dfrac{\overset{\Pi}{\vdash \Gamma}}{\vdash \Gamma, ?_xA}\,\mathsf{w}_I}{\vdash \Gamma, ?_{x+y}A}\,\mathsf{d}_I \quad\rightsquigarrow_{\mathsf{d}_I,3}\quad \dfrac{\overset{\Pi}{\vdash \Gamma}}{\vdash \Gamma, ?_{x+y}A}\,\mathsf{w}_I \qquad\qquad \dfrac{\dfrac{\overset{\Pi}{\vdash}}{\vdash !_xA}\,\bar{\mathsf{w}}_I}{\vdash !_{x+y}A}\,\bar{\mathsf{d}}_I \quad\rightsquigarrow_{\bar{\mathsf{d}}_I,3}\quad \dfrac{\overset{\Pi}{\vdash}}{\vdash !_{x+y}A}\,\bar{\mathsf{w}}_I$$

4. *And if it is after an axiom, we define*

$$\dfrac{\dfrac{\begin{array}{c}\Pi_1\\ \vdash \Gamma\end{array}}{\vdash \Gamma, ?_x A}\,\mathsf{w}_I \qquad \dfrac{}{\vdash !_x A^\perp}\,\bar{\mathsf{w}}_I}{\vdash \Gamma}\,cut \qquad \rightsquigarrow_{cut} \qquad \begin{array}{c}\Pi_1\\ \vdash \Gamma\end{array}$$

$$\dfrac{\dfrac{\begin{array}{c}\Pi_1\\ \vdash \Gamma, A\end{array}}{\vdash \Gamma, ?A}\,\mathsf{d} \qquad \dfrac{\begin{array}{c}\Pi_2\\ \vdash \Delta, A^\perp\end{array}}{\vdash \Delta, !A^\perp}\,\bar{\mathsf{d}}}{\vdash \Gamma, \Delta}\,cut \qquad \rightsquigarrow_{cut} \qquad \dfrac{\begin{array}{c}\Pi_1\\ \vdash \Gamma, A\end{array} \qquad \begin{array}{c}\Pi_2\\ \vdash \Delta, A^\perp\end{array}}{\vdash \Gamma, \Delta}\,cut$$

$$\dfrac{\dfrac{\begin{array}{c}\Pi_1\\ \vdash \Gamma, ?_x A, ?_y A\end{array}}{\vdash \Gamma, ?_{x+y} A}\,\mathsf{c} \qquad \dfrac{}{\vdash !_{x+y} A^\perp}\,\bar{\mathsf{w}}_I}{\vdash \Gamma}\,cut \qquad \rightsquigarrow_{cut} \qquad \dfrac{\dfrac{\begin{array}{c}\Pi_1\\ \vdash \Gamma, ?_x A, ?_y A\end{array} \qquad \dfrac{}{\vdash !_y A^\perp}\,\bar{\mathsf{w}}_I}{\vdash \Gamma, ?_x A}\,cut \qquad \dfrac{}{\vdash !_x A^\perp}\,\bar{\mathsf{w}}_I}{\vdash \Gamma}\,cut$$

$$\dfrac{\dfrac{\begin{array}{c}\Pi_1\\ \vdash \Gamma, !_x A\end{array} \qquad \begin{array}{c}\Pi_2\\ \vdash \Delta, !_y A\end{array}}{\Gamma, \Delta, !_{x+y} A}\,\bar{\mathsf{c}} \qquad \dfrac{\begin{array}{c}\Pi_3\\ \vdash \Xi\end{array}}{\vdash \Xi, ?_{x+y} A^\perp}\,\mathsf{w}_I}{\vdash \Gamma, \Delta, \Xi}\,cut \qquad \rightsquigarrow_{cut}$$

$$\dfrac{\dfrac{\dfrac{\begin{array}{c}\Pi_1\\ \vdash \Gamma, !_x A\end{array} \qquad \dfrac{\begin{array}{c}\Pi_3\\ \vdash \Xi\end{array}}{\vdash \Xi, ?_x A^\perp}\,\mathsf{w}_I}{\vdash \Gamma, \Xi}\,cut}{\vdash \Gamma, \Xi, ?_y A^\perp}\,\mathsf{w}_I \qquad \begin{array}{c}\Pi_2\\ \vdash \Delta, !_y A\end{array}}{\vdash \Gamma, \Xi, \Delta}\,cut$$

$$\dfrac{\dfrac{\begin{array}{c}\Pi_1\\ \vdash \Gamma, ?_{x_1} A^\perp, ?_{x_2} A^\perp\end{array}}{\vdash \Gamma, ?_{x_1+x_2} A^\perp}\,\mathsf{c} \qquad \dfrac{\begin{array}{c}\Pi_2\\ \vdash \Delta, !_{x_3} A\end{array} \qquad \begin{array}{c}\Pi_3\\ \vdash \Xi, !_{x_4} A\end{array}}{\vdash \Delta, \Xi, !_{x_3+x_4=x_1+x_2} A}\,\bar{\mathsf{c}}}{\vdash \Gamma, \Delta, \Xi}\,cut \qquad \rightsquigarrow_{cut}$$

$$\dfrac{\dfrac{\dfrac{\begin{array}{c}\Pi_b\\ \vdash \Gamma, ?_{x_{1,4}} A^\perp, ?_{x_{2,4}} A^\perp, ?_{x_3} A^\perp\end{array} \qquad \begin{array}{c}\Pi_2\\ \vdash \Delta, !_{x_3} A\end{array}}{\vdash \Gamma, \Delta, ?_{x_{1,4}} A^\perp, ?_{x_{2,4}} A^\perp}\,cut}{\vdash \Gamma, \Delta, ?_{x_4} A^\perp}\,\mathsf{c} \qquad \begin{array}{c}\Pi_3\\ \Xi, !_{x_4} A\end{array}}{\vdash \Gamma, \Delta, \Xi}\,cut$$

in which Π_a and Π_b are as follows:

$$\Pi_a = \quad \dfrac{\dfrac{\dfrac{}{\vdash ?_{x_{2,3}} A^\perp, !_{x_{2,3}} A}\,ax \qquad \dfrac{}{\vdash ?_{x_{2,4}} A^\perp, !_{x_{2,4}} A}\,ax}{\vdash ?_{x_{2,3}} A^\perp, ?_{x_{2,4}} A^\perp, !_{x_2} A}\,\bar{\mathsf{c}} \qquad \begin{array}{c}\Pi_1\\ \vdash \Gamma, ?_{x_1} A^\perp, ?_{x_2} A^\perp\end{array}}{\vdash \Gamma, ?_{x_{2,3}} A^\perp, ?_{x_{2,4}} A^\perp, ?_{x_1} A^\perp}\,cut$$

$$\Pi_b = \quad \dfrac{\dfrac{\begin{array}{c}\Pi_a\\ \vdash \Gamma, ?_{x_{2,3}} A^\perp, ?_{x_{2,4}} A^\perp, ?_{x_1} A^\perp\end{array} \qquad \dfrac{\dfrac{}{\vdash ?_{x_{1,3}} A^\perp, !_{x_{1,3}} A}\,ax \qquad \dfrac{}{\vdash ?_{x_{1,4}} A^\perp, !_{x_{1,4}} A}\,ax}{?_{x_{1,3}} A^\perp, ?_{x_{1,4}} A^\perp, !_{x_1} A}\,\bar{\mathsf{c}}}{\vdash \Gamma, ?_{x_{2,3}} A^\perp, ?_{x_{2,4}} A^\perp, ?_{x_{1,3}} A^\perp, ?_{x_{1,4}} A^\perp}\,cut}{\vdash \Gamma, ?_{x_{1,4}} A^\perp, ?_{x_{2,4}} A^\perp, ?_{x_3} A^\perp}\,\mathsf{c}$$

Figure 2 Cut elimination for $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$: group 1 and group 2

$$\dfrac{\dfrac{}{\vdash\, !_x A, ?_x A^\perp}\,ax}{\vdash\, !_x A, ?_{x+y} A^\perp}\,\mathsf{d}_I \quad \rightsquigarrow_{\mathsf{d}_I,4} \quad \dfrac{\dfrac{\dfrac{}{\vdash\, !_x A, ?_x A^\perp}\,ax}{\vdash\, !_x A, ?_x A^\perp, ?_y A^\perp}\,\mathsf{w}_I}{\vdash\, !_x A, ?_{x+y} A^\perp}\,\mathsf{c}$$

$$\dfrac{\dfrac{}{\vdash\, !_x A, ?_x A^\perp}\,ax}{\vdash\, !_{x+y} A, ?_x A^\perp}\,\bar{\mathsf{d}}_I \quad \rightsquigarrow_{\bar{\mathsf{d}}_I,4} \quad \dfrac{\dfrac{}{\vdash\, !_x A, ?_x A^\perp}\,ax \qquad \dfrac{\vdash}{\vdash\, !_y A}\,\bar{\mathsf{w}}_I}{\vdash\, !_{x+y} A, ?_x A^\perp}\,\bar{\mathsf{c}}$$

One defines $\rightsquigarrow_{\mathsf{d}_I}$ (resp. $\rightsquigarrow_{\bar{\mathsf{d}}_I}$) as the transitive closure of the union of the $\rightsquigarrow_{\mathsf{d}_I,i}$ (resp. $\rightsquigarrow_{\bar{\mathsf{d}}_I,i}$).

Even if this definition is non-deterministic, this is not a problem. Every indexed (co)dereliction goes up in the tree, without meeting another one. This implies that this rewriting is confluent: the result of the rewriting does not depend on the choices made.

▶ Remark 10. It is easy to define a forgetful functor U, which transforms a formula (resp. a proof) of $\mathsf{DB}_\mathcal{S}\mathsf{LL}$ into a formula (resp. a proof) of DiLL. For a formula A of $\mathsf{DB}_\mathcal{S}\mathsf{LL}$, $U(A)$ is A where each $!_x$ (resp. $?_x$) is transformed into $!$ (resp. $?$), which is a formula of DiLL. For a proof-tree without any d_I and $\bar{\mathsf{d}}_I$, the idea is the same: when an exponential rule of $\mathsf{DB}_\mathcal{S}\mathsf{LL}$ is applied in a proof-tree Π, the same rule but not indexed is applied in $U(\Pi)$, which is a proof-tree in DiLL. Moreover, we notice that if $\Pi_1 \rightsquigarrow_{cut} \Pi_2$, $U(\Pi_1) \rightsquigarrow_{\mathsf{DiLL}} U(\Pi_2)$ where $\rightsquigarrow_{\mathsf{DiLL}}$ is the cut-elimination in [12].

We can now define a cut-elimination procedure:

▶ **Definition 11.** *The rewriting $\rightsquigarrow$ is defined on derivation trees. For a tree Π, we apply $\rightsquigarrow_{\mathsf{d}_I}$, $\rightsquigarrow_{\bar{\mathsf{d}}_I}$ and $\rightsquigarrow_{cut}$ as long as it is possible. When there are no more cuts, the rewriting ends.*

▶ **Theorem 12.** *The rewriting procedure $\rightsquigarrow$ terminates on each derivation tree, and reaches an equivalent tree with no cut.*

In order to prove this theorem, we first need to prove a lemma, which shows that the (co)dereliction elimination is well defined.

▶ **Lemma 13.** *For each derivation tree Π, if we apply $\rightsquigarrow_{\mathsf{d}_I}$ and $\rightsquigarrow_{\bar{\mathsf{d}}_I}$ to Π, this procedure terminates such that $\Pi \rightsquigarrow_{\mathsf{d}_I} \Pi_1 \rightsquigarrow_{\bar{\mathsf{d}}_I} \Pi_2$ without any d_I and $\bar{\mathsf{d}}_I$ in Π_2.*

Proof. Let Π be a proof-tree. Each rule has a height (using the usual definition for nodes in a tree). We define the *depth* of a node as the height of the tree minus the height of this node. The procedure $\rightsquigarrow_{\mathsf{d}_I}$ terminates on Π: let $a(\Pi)$ be the number of indexed derelictions in Π and $b(\Pi)$ be the sum of the depth of each indexed derelictions in Π. Now, we define $H(\Pi) = (a(\Pi), b(\Pi))$ and $<_{lex}$ as the lexicographical order on $\mathbb{N}^2$. For each step of $\rightsquigarrow_{\mathsf{d}_I}$ such that $\Pi_i \rightsquigarrow_{\mathsf{d}_I} \Pi_j$, we have $H(\Pi_i) <_{lex} H(\Pi_j)$:

1. If $\Pi_i \rightsquigarrow_{\mathsf{d}_I,1} \Pi_j$, the number of d_I does not change and the sum of depths decreases by 1. Hence, $H(\Pi_i) <_{lex} H(\Pi_j)$.
2. If $\Pi_i \rightsquigarrow_{\mathsf{d}_I,k} \Pi_j$ with $2 \leq k \leq 4$, the number of derelictions decreases, so $H(\Pi_i) <_{lex} H(\Pi_j)$.

Using this property and the fact that $<_{lex}$ is a well-founded order on $\mathbb{N}^2$, this rewriting procedure has to terminates on a tree Π_1. Moreover, if there is an indexed dereliction in Π_1, this dereliction is below an other rule, so $\rightsquigarrow_{\mathsf{d}_I,i}$ for $1 \leq i \leq 4$ can be applied which leads to a contradiction with the definition of Π_1. Then, there is no indexed dereliction in Π_1.

Using similar arguments, the rewriting procedure $\rightsquigarrow_{\bar{\mathsf{d}}_I}$ on Π_1 ends on a tree Π_2 where there is no codereliction (and no dereliction because the procedure $\rightsquigarrow_{\bar{\mathsf{d}}_I}$ does not introduce any derelictions). ◀

Proof of Theorem 12. If we apply our procedure $\rightsquigarrow$ on a tree Π we will, using Lemma 13, have a tree $\Pi_{\mathsf{d}_I,\bar{\mathsf{d}}_I}$ such that $\Pi \rightsquigarrow_{\mathsf{d}_I} \Pi_{\mathsf{d}_I} \rightsquigarrow_{\bar{\mathsf{d}}_I} \Pi_{\mathsf{d}_I,\bar{\mathsf{d}}_I}$ and there is no dereliction and no codereliction in $\Pi_{\mathsf{d}_I,\bar{\mathsf{d}}_I}$. Hence, the procedure $\rightsquigarrow$ applied on Π gives a rewriting

$$\Pi \rightsquigarrow_{\mathsf{d}_I} \Pi_{\mathsf{d}_I} \rightsquigarrow_{\bar{\mathsf{d}}_I} \left(\Pi_{\mathsf{d}_I,\bar{\mathsf{d}}_I} = \Pi_0\right) \rightsquigarrow_{cut} \Pi_1 \rightsquigarrow_{cut} \dots$$

Applying the forgetful functor U from Remark 10 on each tree Π_i (for $i \in \mathbb{N}$), the cut-elimination theorem of DiLL [30] implies that this rewriting terminates at a rank n, because the cut-elimination rules of $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$ which are used in Π_0 are those of DiLL when the indexes are removed. Then, $\Pi \rightsquigarrow^* \Pi_n$ where Π_n is cut-free. ◀

▶ Remark 14. Notice that while DiLL is famous for introducing *formal sums of proofs* with its cut-elimination, we have none of that here. The syntactical reason is that, as exponential are labelled with indices, there is no non-deterministic choices to make here. The semantical reason is that sum is introduced while operating a cut between codereliction and contraction (differentiating a scalar multiplication of functions) or a cut between a dereliction and cocontraction (applying a convolution product of distributions to a linear map). As detailed in Section 4, LPDOcc do not behave like this and fundamental solutions or differential operators are painlessly propagated into the first argument of a distribution or function.

4 An indexed differential linear logic

In the previous section, we have defined a logic $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$ as the syntactical differential of an indexed linear logic $\mathsf{B}_{\mathcal{S}}\mathsf{LL}$, with its cut elimination procedure. It is a syntactical differentiation of BLL, as it uses the idea that differentiation is expressed through co-structural rules that mirror the structural rules of LL. Here we will take a semantical point of view: starting from differential linear logic, we will index it with LPDOcc into a logic named IDiLL, and then study the relation between $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$ and IDiLL.

4.1 IDiLL: a generalization of D-DiLL

As we saw in Section 2, Kerjean generalized $\bar{\mathsf{d}}$ and d in previous work [25], with the idea that in DiLL, the codereliction corresponds to the application of the differential operator D_0 whereas the dereliction corresponds to the resolution of the differential equation associated to D_0. This led to a logic D-DiLL, where $\bar{\mathsf{d}}$ and d have the same effect but with a LPDOcc D instead of D_0, and where the exponential connectives are indexed by this operator D. One would expect that this work could be connected to $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$, but these definitions clash with the traditional intuitions of graded logics. The first reason is syntactical: in graded logics, the exponential connectives are indexed by elements of an algebraic structure, whereas in D-DiLL only one operator is used as an index. We then change the logic D-DiLL into a logic IDiLL, which is much closer to what is done in the graded setting. In this new framework, we will consider the composition of two LPDOcc as our monoidal operation. Indeed, thanks to Proposition 4, we have that $D_1(\phi) * D_2(\psi) = (D_1 \circ D_2)(\phi * \psi)$. The convolution $*$ being the interpretation of the cocontraction rule $\bar{\mathsf{c}}$, the composition is the monoidal operation on the set of LPDOcc that we are looking for. Moreover, the composition of LPDOcc is commutative, which is a mandatory property for the monoidal operation in a graded framework. We describe the exponential rules of IDiLL in Figure 3.

The indexed rules d_D and $\bar{\mathsf{d}}_D$ are generalized to rules d_I and $\bar{\mathsf{d}}_I$ involving a variety of LPDOcc, while rules d and $\bar{\mathsf{d}}$ are ignored for now (see the first discussion of section 5). The interpretations of $?_D A$ and $!_D A$, and hence the typing of d_I and $\bar{\mathsf{d}}_I$ are changed from what

$$\frac{\vdash \Gamma}{\vdash \Gamma, ?_D A}\ \mathsf{w}_I \qquad \frac{\vdash \Gamma, ?_{D_1} A, ?_{D_2} A}{\vdash \Gamma, ?_{D_1 \circ D_2} A}\ \mathsf{c} \qquad \frac{\vdash \Gamma, ?_{D_1} A}{\vdash \Gamma, ?_{D_1 \circ D_2} A}\ \mathsf{d}_I$$

$$\frac{}{\vdash !_D A}\ \bar{\mathsf{w}}_I \qquad \frac{\vdash \Gamma, !_{D_1} A \qquad \vdash \Delta, !_{D_2} A}{\vdash \Gamma, \Delta, !_{D_1 \circ D_2} A}\ \bar{\mathsf{c}} \qquad \frac{\vdash \Gamma, !_{D_1} A}{\vdash \Gamma, !_{D_1 \circ D_2} A}\ \bar{\mathsf{d}}_I$$

Figure 3 Exponential rules of IDiLL

D-DiLL would have directly enforced (see remark 15). Our new interpretations for $?_D A$ and $!_D A$ are now compatible with the intuition that in graded logics, rules are supposed to add information.

$$[\![?_D A]\!] := \{g \mid \exists f \in [\![?A]\!],\ D(g) = f\} \qquad [\![!_D A]\!] := ([\![?_D A]\!]')' = \hat{D}([\![!A]\!])$$
$$\mathsf{d}_I : [\![?_{D_1} A]\!] \to [\![?_{D_1 \circ D_2} A]\!] \qquad \bar{\mathsf{d}}_I : [\![!_{D_1} A]\!] \to [\![!_{D_1 \circ D_2} A]\!]$$

The reader might note that these new definitions have another benefit: they ensure that the dereliction (resp. the codereliction) is well typed when it consists in solving (resp. applying) a differential equation. This will be detailed in Section 4.3.

Notice that a direct consequence of Proposition 4 is that for two LPDOcc D_1 and D_2, $\Phi_{D_1 \circ D_2} = \Phi_{D_1} * \Phi_{D_2}$. It expresses that our monoidal law is also well-defined w.r.t. the interpretation of the indexed dereliction.

▶ Remark 15. Our definition for indexed connectives and thus for the types of d_D and $\bar{\mathsf{d}}_D$ differs from the original one in D-DiLL [25]. Kerjean gave types $\mathsf{d}_D : ?_{D,old} E' \to ?E'$ and $\bar{\mathsf{d}}_D : !_{D,old} E \to !E$. However, graded linear logic carries different intuitions: indices are here to keep track of the operations made through the inference rules. As such, d_D and $\bar{\mathsf{d}}_D$ should introduces indices D and not delete it. Compared with work in [25], we then change the interpretation of $?_D A$ and $!_D A$, and the types of d_D and $\bar{\mathsf{d}}_D$. Thanks to this change, we will see in the rest of the paper D-DiLL as a particular case of $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$.

4.2 Grading linear logic with differential operators

In this section, we will show that IDiLL consists of admissible rules of $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$ for the monoid of LPDOcc. In order to connect IDiLL with our results from Section 3, we have to study the algebraic struture of the set of linear partial differential operators with constant coefficients $\mathcal{D}$. More precisely, our goal is to prove the following theorem.

▶ **Theorem 16.** *The set $\mathcal{D}$ of LPDOcc is an additive splitting monoid under composition, with the identity operator id as the identity element.*

To prove this result, we will use multivariates polynomials: $\mathbb{R}[X^{(\omega)}] := \bigcup_{n \in \mathbb{N}} \mathbb{R}[X_1, \ldots, X_n]$. It is well known that $(\mathbb{R}[X^{(\omega)}], +, \times, 0, 1)$ is a commutative ring. Its monoidal restriction is isomorphic to $(\mathcal{D}, \circ, id)$, the LPDOcc endowed with composition, through the following monoidal isomorphism

$$\chi \colon \begin{cases} (\mathcal{D}, \circ) \to (\mathbb{R}[X^{(\omega)}], \times) \\ \displaystyle\sum_{\alpha \in \mathbb{N}^n} a_\alpha \frac{\partial^{|\alpha|}(\_)}{\partial x^\alpha} \mapsto \sum_{\alpha \in \mathbb{N}^n} a_\alpha X^{\alpha_1} \ldots X_n^{\alpha_n} \end{cases}$$

The following proposition is crucial in the indexation of $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$ by differential operators, since the monoid in $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$ has to be additive splitting.

▶ **Proposition 17.** *The monoid* $(\mathbb{R}[X^{(\omega)}], \times, 1)$ *is additive splitting.*

The proof requires some algebraic definitions to make it more readable.

▶ **Definition 18.** *Let $\mathcal{R}$ be a non-zero commutative ring.*

1. *$\mathcal{R}$ is an integral domain if for each $x, y \in \mathcal{R}\backslash\{0\}$, $xy \neq 0$.*
2. *An element $u \in \mathcal{R}$ is a unit if there is $v \in \mathcal{R}$ such that $uv = 1$.*
3. *Two elements $x, y \in \mathcal{R}$ are associates if x divides y and y divides x.*
4. *$\mathcal{R}$ is a factorial ring if it is an integral domain such that for each $x \in \mathcal{R}\backslash\{0\}$ there is a unit $u \in \mathcal{R}$ and $p_1, \dots, p_n \in \mathcal{R}$ irreducible elements such that $x = up_1 \dots p_n$ and for every other decomposition $vq_1 \dots q_m = up_1 \dots p_n$ (with v unit and q_i irreducible for each i) we have $n = m$ and a bijection $\sigma : \{1, \dots, n\} \to \{1, \dots, n\}$ such that p_i and $q_{\sigma(i)}$ are associated for each i.*

Proof of Proposition 17. For each integer n, the ring $\mathbb{R}[X_1, \dots, X_n]$ is factorial. This classical proposition is for example proved in [4, 2.7 Satz 7].

Let us take four polynomials P_1, P_2, P_3 and P_4 in $\mathbb{R}[X^{(\omega)}]$ such that $P_1 \times P_2 = P_3 \times P_4$. There is $n \in \mathbb{N}$ such that $P_1, P_2, P_3, P_4 \in \mathbb{R}[X_1, \dots, X_n]$.

If $P_1 = 0$ or $P_2 = 0$, then $P_3 = 0$ or $P_4 = 0$, since $\mathbb{R}[X_1, \dots, X_n]$ has integral domain. If for example $P_1 = 0$ and $P_3 = 0$, one can define

$$P_{1,3} = 0 \qquad P_{1,4} = P_4 \qquad P_{2,3} = P_2 \qquad P_{2,4} = 1$$

which gives a correct decomposition. And we can reason symmetrically for the other cases.

Now, we suppose that each polynomials P_1, P_2, P_3 and P_4 are non-zero. By factoriality of $\mathbb{R}[X_1, \dots, X_n]$, we have a decomposition

$$P_i = u_i Q_{n_{i-1}+1} \times \dots Q_{n_i} \qquad \text{(for each } 1 \leq i \leq 4)$$

where $n_0 = 0 \leq n_1 \dots \leq n_4$, u_i are units and Q_i are irreducible. Then, the equality $P_1 P_2 = P_3 P_4$ gives

$$u_1 u_2 Q_1 \dots Q_{n_2} = u_3 u_4 Q_{n_2+1} \dots Q_{n_4}.$$

Since $u_1 u_2$ and $u_3 u_4$ are units, the factoriality implies that $n_2 = n_4 - n_2$ and that there is a bijection $\sigma : \{1, \dots, n_2\} \to \{n_2 + 1, \dots, n_4\}$ such that Q_i and $Q_{\sigma(i)}$ are associates for each $1 \leq i \leq n_2$. It means that for each $1 \leq i \leq n_2$, there is a unit v_i such that $Q_{\sigma(i)} = v_i Q_i$. Hence, defining two sets $A_3 = \sigma^{-1}(\{n_2 + 1, \dots, n_3\})$ and $A_4 = \sigma^{-1}(\{n_3 + 1, \dots, n_4\})$ we can rewrite our polynomials P_1 and P_2 using:

$$\begin{aligned}
A_{1,3} &= A_3 \cap \{1, \dots, n_1\} = p_1, \dots, p_{m_1} & R_{1,3} &= Q_{p_1} \dots Q_{p_{m_1}} & v_{1,3} &= v_{p_1} \dots v_{p_{m_1}} \\
A_{1,4} &= A_4 \cap \{1, \dots, n_1\} = q_1, \dots, q_{m_2} & R_{1,4} &= Q_{q_1} \dots Q_{q_{m_2}} & v_{1,4} &= v_{q_1} \dots v_{q_{m_2}} \\
A_{2,3} &= A_3 \cap \{n_1 + 1, \dots, n_2\} = r_1, \dots, r_{m_3} & R_{2,3} &= Q_{r_1} \dots Q_{r_{m_3}} & v_{2,3} &= v_{r_1} \dots v_{r_{m_3}} \\
A_{2,4} &= A_4 \cap \{n_1 + 1, \dots, n_2\} = s_1, \dots, s_{m_4} & R_{2,4} &= Q_{s_1} \dots Q_{s_{m_4}} & v_{2,4} &= v_{s_1} \dots v_{s_{m_4}}
\end{aligned}$$

which leads to

$$P_1 = u_1 R_{1,3} R_{1,4} \qquad P_2 = u_2 R_{2,3} R_{2,4} \qquad P_3 = u_3 v_{1,3} R_{1,3} v_{2,3} R_{2,3} \qquad P_4 = u_4 v_{1,4} R_{1,4} v_{2,4} R_{2,4}$$

Finally, we define our new polynomials

$$P_{1,3} = u_1 R_{1,3} \qquad P_{1,4} = R_{1,4} \qquad P_{2,3} = \frac{u_3 v_{1,3} v_{2,3}}{u_1} R_{2,3} \qquad P_{2,4} = \frac{u_1 u_2}{u_3 v_{1,3} v_{2,3}} R_{2,4}$$

gives the wanted decomposition: this is straightforward for P_1, P_2 and P_3 (the coefficients are chosen for that), and for P_4, it comes from the fact that $u_1 u_2 = u_3 u_4$ (which is in the definition of a factorial ring), and that $v_{1,a} v_{1,b} v_{2,a} v_{2,b} = 1$ which is easy to see using our new polynomials $R_{1,3}, R_{1,4}, R_{2,3}, R_{2,4}$ and the equality $P_1 P_2 = P_3 P_4$. ◀

This result ensures that $(\mathcal{D}, \circ, id)$ is an additive splitting monoid. Then, $\mathcal{D}$ induces a logic $\mathsf{DB}_{\mathcal{D}}\mathsf{LL}$. In this logic, since the order of the monoid is defined through the composition rule, for D_1 and D_2 in $\mathcal{D}$ we have

$$D_1 \leq D_2 \iff \exists D_3 \in \mathcal{D},\ D_2 = D_1 \circ D_3$$

which expresses that the rules d_I and $\bar{\mathsf{d}}_I$ from IDiLL and those from $\mathsf{DB}_{\mathcal{D}}\mathsf{LL}$ are exactly the same. In addition, the weakening and the coweakening from $\mathsf{DB}_{\mathcal{D}}\mathsf{LL}$ are rules which exists in IDiLL (the (co)weakening with $D = id$), and a weakening (resp. a coweakening) in IDiLL can be expressed in $\mathsf{DB}_{\mathcal{D}}\mathsf{LL}$ as an indexed weakening (resp. an indexed coweakening). In fact, this indexed weakening is the one that appears in the cut elimination procedure of $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$. Hence, this gives the following proposition.

▶ **Proposition 19.** *Each rule of IDiLL is admissible in $\mathsf{DB}_{\mathcal{D}}\mathsf{LL}$, and each rule of $\mathsf{DB}_{\mathcal{D}}\mathsf{LL}$ except d and $\bar{\mathsf{d}}$ is admissible in IDiLL.*

With this proposition, Theorem 12 ensures that IDiLL enjoys a cut elimination procedure, which is the same as the one defined for $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$. This procedure will even be easier in the case of IDiLL. One issue in the definition of the cut elimination of $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$ is to define w_I and $\bar{\mathsf{w}}_I$. This is no longer a problem in IDiLL because these rules already exist in this framework.

4.3 A concrete semantics for IDiLL

Now that we have defined the rules and the cut elimination procedure for a logic able to deal with the interaction between differential operators in its syntax, we should express how it semantically acts on smooth maps and distributions. For MALL formulas and rules, the interpretation is the same as the one for DiLL (or D-DiLL), given in Section 2. First, we give the interpretation of our indexed exponential connectives. Beware that we are still here in a *finitary setting*, in wich exponential connectives only apply to finite dimensional vector spaces, meaning that $[\![A]\!] = \mathbb{R}^n$ for some n in equation (2) below. This makes sense syntactically as long as we do not introduce a promotion rule, and corresponds to the denotational model exposed originally by Kerjean. As mentioned in the conclusion, we think that work in higher dimensional analysis should provide an higher-order interpretation for indexed exponential connectives [18].

Consider $D \in \mathcal{D}$. Then D applies independently to any $f \in \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R})$ for any n, by injecting smoothly $\mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R}) \subseteq \mathcal{C}^\infty(\mathbb{R}^m, \mathbb{R})$ for any $m \geq n$. We give the following interpretation of graded exponential connectives:

$$\begin{aligned} [\![!_D A]\!] &:= (\{f \in \mathcal{C}^\infty([\![A]\!], \mathbb{R}) \mid \exists g \in \mathcal{C}^\infty([\![A]\!], \mathbb{R}),\ D(f) = g\})' = \hat{D}([\![!A]\!]) \\ [\![?_D A]\!] &:= \{f \in \mathcal{C}^\infty([\![A]\!]', \mathbb{R}) \mid \exists g \in \mathcal{C}^\infty([\![A]\!]', \mathbb{R}),\ D(f) = g\} = D^{-1}([\![?A]\!]) \end{aligned} \tag{2}$$

From this definition, one can note that when $D = id$, we get

$$[\![!_{id} A]\!] = (\mathcal{C}^\infty([\![A]\!], \mathbb{R}))' = [\![!A]\!] \qquad [\![?_{id} A]\!] = \mathcal{C}^\infty([\![A]\!]', \mathbb{R}) = [\![?A]\!].$$

▶ Remark 20. One can notice that, as differential equations always have solutions in our case, the space of solutions $[\![?_D A]\!]$ is *isomorphic* to the function space $[\![?A]\!]$. The isomorphism in question is plainly the dereliction $\mathsf{d}_D : f \mapsto \Phi_D * f$. While our setting might be seen as too simple from the point of view of analysis, it is a first and necessary step before extending IDiLL to more intricate differential equations. If we were to explore the abstract categorical setting for our model, these isomorphisms would be relevant in a *bicategorical* setting.

The next step is to give a semantical interpretation of the exponential rules. Most of these interpretations will be quite natural, in the sense that they will be based on the intuitions given in Section 4.1 and on the model of DiLL described in previous work [25]. However, the contraction rule will require some refinements. The contraction takes two formulas $?_{D_1}A$ and $?_{D_2}A$, and contracts them into a formula $?_{D_1 \circ D_2}A$. In our model, it corresponds to the contraction of two functions $f \in \mathcal{C}^\infty(E', \mathbb{R})$ such that $D_1(f) \in \mathcal{C}^\infty(E', \mathbb{R})$ and $g \in \mathcal{C}^\infty(E', \mathbb{R})$ such that $D_2(g) \in \mathcal{C}^\infty(E', \mathbb{R})$ into a function $h \in \mathcal{C}^\infty(E', \mathbb{R})$ such that $D_1 \circ D_2(h) \in \mathcal{C}^\infty(E', \mathbb{R})$. In differential linear logic, the contraction is interpreted as the pointwise product of functions. This is not possible here, since we do not know how to compute $D_1 \circ D_2(f.g)$. We will then use the fundamental solution, which has the property that $D(\Phi_D * f) = f$. This leads to the following definition.

▶ **Definition 21.** *We define the interpretation of each exponential rule of IDiLL by:*

$$\mathsf{w} : \begin{cases} \mathbb{R} \to ?_{id}E \\ 1 \mapsto cst_1 \end{cases} \qquad \bar{\mathsf{w}} : \begin{cases} \mathbb{R} \to !_{id}E \\ 1 \mapsto \delta_0 \end{cases}$$

$$\mathsf{c} : \begin{cases} ?_{D_1}E \mathbin{\hat{\otimes}} ?_{D_2}E \to ?_{D_1 \circ D_2}E \\ \qquad f \otimes g \mapsto \Phi_{D_1 \circ D_2} * (D_1(f).D_2(g)) \end{cases} \qquad \bar{\mathsf{c}} : \begin{cases} !_{D_1}E \mathbin{\hat{\otimes}} !_{D_2}E \to !_{D_1 \circ D_2}E \\ \qquad \psi \otimes \phi \mapsto \psi * \phi \end{cases}$$

$$\mathsf{d}_I : \begin{cases} ?_{D_1}E \to ?_{D_1 \circ D_2}E \\ \qquad f \mapsto \Phi_{D_2} * f \end{cases} \qquad \bar{\mathsf{d}}_I : \begin{cases} !_{D_1}E \to !_{D_1 \circ D_2}E \\ \qquad \psi \mapsto \psi \circ D_2 \end{cases}$$

▶ Remark 22. One can note that we only have defined the interpretation of the (co)weakening when it is indexed by the identity. This is because, as well as for $\mathsf{DB}_\mathcal{S}\mathsf{LL}$, the one of w_I and $\bar{\mathsf{w}}_I$ can be deduced from this one, using the definition of d_I and $\bar{\mathsf{d}}_I$. This leads to

$$\mathsf{w}_I : 1 \mapsto \Phi_D * cst_1 = cst_{\Phi_D(cst_1)} \qquad \bar{\mathsf{w}}_I : 1 \mapsto \delta_0 \circ D = (f \mapsto D(f)(0)).$$

The interpretation for $\bar{\mathsf{c}}$ and c is justified by the fact that in Nuclear Fréchet or Nuclear DF spaces [25], both the $\parr$ and $\otimes$ connectors of LL are interpreted by the same completed topological tensor product $\hat{\otimes}$. They however do not apply to the same kind of spaces, as $?E$ is Fréchet while $!E$ isn't. Thus, basic operations on the interpretation of $A \parr B$ or $A \otimes B$ are first defined on elements $a \otimes b$ on the tensor product, and then extended by linearity and completion.

In order to ensure that Definition 21 gives a correct model of IDiLL, we should verify the well-typedness of each morphism. First, this is obvious for the weakening and the coweakening. The function cst_1 defined on E is smooth, and δ_0 is the canonical example of a distribution. Moreover, we interpret w and $\bar{\mathsf{w}}$ in the same way as in the model of DiLL on which our intuitions are based. The indexed dereliction is well-typed, because for $f \in ?_{D_1}E$, there is $g \in \mathcal{C}^\infty(E', \mathbb{R})$ such that $D_1(f) = g$ by definition. Hence, $D_1 \circ D_2(\Phi_{D_2} * f) = D_1(f) = g \in \mathcal{C}^\infty(E', \mathbb{R})$ so $\mathsf{d}_I(f) \in ?_{D_1 \circ D_2}E$. For the contraction, if $f \in ?_{D_1}E$ and $g \in ?_{D_2}E$, $D_1(f)$ and $D_2(g)$ are in $\mathcal{C}^\infty(E', \mathbb{R})$, and so is their scalar product. Hence, $D_1 \circ D_2(\mathsf{c}(f \otimes g)) = D_1(f).D_2(g)$ which is in $\mathcal{C}^\infty(E', \mathbb{R})$. The indexed codereliction is also well-typed: for $\psi \in !_{D_1}E$, equation (2) ensures that $\psi = \hat{D}_1(\psi_1)$ with $\psi_1 \in !E$, so $\psi \circ D_2 = (\psi_1 \circ D_1) \circ D_2 \in !_{D_1 \circ D_2}E$. Finally, using similar arguments for the cocontraction, if $\psi \in !_{D_1} E$ and $\phi \in !_{D_2} E$, then $\psi = \hat{D_1}(\psi_1)$ and $\phi = \hat{D_2}(\phi_1)$, with $\psi_1, \phi_1 \in !E$. Hence,

$$\psi * \phi = (\psi_1 \circ D_1) * (\phi_1 \circ D_2) = (\psi_1 * \phi_1) \circ (D_1 \circ D_2) = \widehat{D_1 \circ D_2}(\psi_1 * \phi_1) \in !_{D_1 \circ D_2} E.$$

We have then proved the following proposition.

▶ **Proposition 23.** *Each morphism* $\mathsf{w}, \bar{\mathsf{w}}, \mathsf{c}, \bar{\mathsf{c}}, \mathsf{d}_I$ *and* $\bar{\mathsf{d}}_I$ *is well-typed.*

Another crucial point to study is the compatibility between this model and the cut elimination procedure $\rightsquigarrow$. In denotational semantics, one would expect that a model is invariant w.r.t. the computation. In our case, that would mean that for each step of rewriting of $\rightsquigarrow$, the interpretation of the proof-tree has the same value.

It is easy to see that this is true for the cut $\mathsf{w}_I/\bar{\mathsf{w}}_I$, since $D(\Phi_D * cst_1)(0) = cst_1(0) = 1$. For the cut between a contraction and an indexed coweakening, the interpretation before the reduction is $\delta_0(D_1 \circ D_2)(\Phi_{D_1 \circ D_2}(D_1(f).D_2(g))) = D_1(f)(0).D_2(g)(0)$, which is exactly the interpretation after the reduction[2].

Finally, proving the invariance of our semantics over the cut between a contraction or a weakening, and a cocontraction takes slightly more work. The weakening case is enforced by linearity of the distributions, while the contraction case relies on the density of $\{\delta_x \mid x \in E\}$ in $!E$.

▶ **Lemma 24.** *The interpretation of* $\mathsf{DB_{\mathcal{S}}LL}$ *with* $\mathcal{D}$ *as indexes is invariant over the* $\mathsf{c}/\bar{\mathsf{c}}$ *and the* $\bar{\mathsf{c}}/\mathsf{w}_I$ *cut-elimination rules, as given in Figure 2.*

Proof. Before cut-elimination, the interpretation of the $\bar{\mathsf{c}}/\mathsf{w}$ as given in Figure 2 is:

$$\begin{aligned}
&(\psi * \phi)(\Phi_{D_1 \circ D_2} * cst_1) \\
&= \psi(x \mapsto \phi(y \mapsto \Phi_{D_1} * (\Phi_{D_2} * cst_1)(x+y))) \\
&= \psi(x \mapsto \phi(y \mapsto \Phi_{D_1}(z \mapsto \Phi_{D_2} * cst_1(x+y-z)))) \\
&= \psi(x \mapsto \phi(y \mapsto \Phi_{D_1}(cst_{\Phi_{D_2}(cst_1)}))) \\
&= \psi(x \mapsto \phi(y \mapsto \Phi_{D_1}(\Phi_{D_2}(cst_1).cst_1))) \\
&= \psi(x \mapsto \phi(y \mapsto \Phi_{D_2}(cst_1).\Phi_{D_1}(cst_1))) && \text{(by homogeneity of } \phi) \\
&= \psi(x \mapsto \phi(cst_{\Phi_{D_2}(cst_1).\Phi_{D_1}(cst_1)})) \\
&= \psi(x \mapsto \phi(\Phi_{D_1}(cst_1).cst_{\Phi_{D_2}(cst_1)})) \\
&= \psi(x \mapsto \Phi_{D_1}(cst_1).\phi(cst_{\Phi_{D_2}(cst_1)})) && \text{(by homogeneity of } \phi) \\
&= \psi(cst_{\Phi_{D_1}(cst_1).\phi(cst_{\Phi_{D_2}(cst_1)})}) \\
&= \psi(\phi(cst_{\Phi_{D_2}(cst_1)}).cst_{\Phi_{D_1}(cst_1)}) \\
&= \phi(cst_{\Phi_{D_2}(cst_1)}).\psi(cst_{\Phi_{D_1}(cst_1)}) && \text{(by homogeneity of } \psi)
\end{aligned}$$

which corresponds to the interpretation of the proof after cut-elimination.

Let us tackle now the $\bar{\mathsf{c}}/\mathsf{c}$ cut-elimination case. Suppose that we have $D_1, D_2, D_3, D_4 \in \mathcal{D}$ such that $D_1 \circ D_2 = D_3 \circ D_4$. By the additive splitting property we have $D_{1,3}, D_{1,4}, D_{2,3}, D_{2,4}$ such that

$$D_1 = D_{1,3} \circ D_{1,4} \qquad D_2 = D_{2,3} \circ D_{2,4} \qquad D_3 = D_{1,3} \circ D_{2,3} \qquad D_4 = D_{1,4} \circ D_{2,4}.$$

[2] The scalar product $(\_.\_)$ appears as the tensor product in $\mathbb{R} \otimes \mathbb{R}$, and is transparent in sequent interpretation as $\mathbb{R} = [\![\bot]\!]$.

The diagrammatic translation of the cut-elimination rule in Figure 2 is the following.

$$\begin{tikzcd}
!_{D_1}E \otimes !_{D_2}E \arrow[r, "\mathsf{c}'_{D_{1,3},D_{1,4}} \otimes \mathsf{c}'_{D_{2,3},D_{2,4}}"] \arrow[d, "\bar{\mathsf{c}}_{D_1,D_2}"] & !_{D_{1,3}}E \otimes !_{D_{1,4}}E \otimes !_{D_{2,3}}E \otimes !_{D_{2,4}}E \arrow[dd] \\
!_{D_1 \circ D_2}E = !_{D_3 \circ D_4}E \arrow[d, "\mathsf{c}'_{D_3,D_4}"] & \\
!_{D_3}E \otimes !_{D_4}E & !_{D_{1,3}}E \otimes !_{D_{2,3}}E \otimes !_{D_{1,4}}E \otimes !_{D_{2,4}}E \arrow[l, "\bar{\mathsf{c}}_{D_{1,3},D_{2,3}} \otimes \bar{\mathsf{c}}_{D_{1,4},D_{2,4}}"]
\end{tikzcd}$$

As we interpret formulas by reflexive spaces, we can without loss of generality interpret contraction as a law $\mathsf{c}'_{D_a,D_b} : !_{D_a \circ D_b}E \to !_{D_a}E \otimes !_{D_b}E$. Because we are working on finite dimensional spaces E, an application of Hahn-Banach theorem gives us that the span of $\{\delta_x \mid x \in E\}$ is dense in $!E$. As such, the interpretation of c' can be restricted to elements of the form $\delta_x \circ D_a \circ D_b \in !_{D_a \circ D_b}E$, and one checks easily that the dual of c (Definition 21) is : $\mathsf{c}'_{D_a,D_b} : \delta_x \circ D_a \circ D_b \mapsto (\delta_x \circ D_a) \otimes (\delta_x \circ D_b)$. Remember that the convolution of Dirac operators is the Dirac of the sum of points, and as such we have : $\bar{\mathsf{c}}_{D_a,D_b} : (\delta_x \circ D_a) \otimes (\delta_y \circ D_b) \mapsto (\delta_{x+y} \circ D_b \circ D_a)$. Now one can compute easily that the diagram above commutes on elements $(\delta_x \circ D_1) \otimes (\delta_y \circ D_2)$ of $!_{D_1}E \otimes !_{D_2}E$, and as such commutes on all elements by density and continuity of $\bar{\mathsf{c}}$ and c'. ◀

In order to ensure that this model is fully compatible with $\rightsquigarrow$, it also has to be invariant by $\rightsquigarrow_{\mathsf{d}_I}$ and by $\rightsquigarrow_{\bar{\mathsf{d}}_I}$. For $\rightsquigarrow_{\mathsf{d}_I}$, the interpretation of the reduction step when the indexed dereliction meets a contraction is

$$\begin{aligned}
&\Phi_{D_3} * (\Phi_{D_1 \circ D_2} * (D_1(f).D_2(g))) \\
&= \Phi_{D_1 \circ D_2 \circ D_3} * ((D_1(f).D_2(g)).cst_1) \\
&= \Phi_{D_1 \circ D_2 \circ D_3} * ((D_1(f).D_2(g)).D_3(\Phi_{D_3} * cst_1)) \\
&= \Phi_{D_1 \circ D_2 \circ D_3} * (D_1 \circ D_2(\Phi_{D_1 \circ D_2} * (D_1(f).D_2(g))).D_3(\Phi_{D_3} * cst_1))
\end{aligned}$$

which is the interpretation after the application of $\rightsquigarrow_{\mathsf{d}_I,2}$. The case with a weakening translates the fact that $\Phi_{D_1 \circ D_2} = \Phi_{D_1} * \Phi_{D_2}$. Finally, the axiom rule introduces a distribution $\psi \in !_{D_1}E$ and a smooth map $f \in !_{D_1}E$, and $\rightsquigarrow_{\mathsf{d}_I,4}$ corresponds to the equality $\Phi_{D_1 \circ D_2} * D_1(f) = \Phi_{D_2} * f$.

The remaining case is the procedure $\rightsquigarrow_{\bar{\mathsf{d}}_I}$, which is quite similar to $\rightsquigarrow_{\mathsf{d}_I}$. The invariance of the model with the cocontraction case follows from Proposition 4. For the weakening, this is just the associativity of the composition, and the axiom works because δ_0 is the neutral element of the convolution product. We can finally deduce that our model gives an interpretation which is invariant by the cut elimination procedure of Section 3.

▶ **Proposition 25.** *Each morphism* $\mathsf{w}, \bar{\mathsf{w}}, \mathsf{c}, \bar{\mathsf{c}}, \mathsf{d}_I$ *and* $\bar{\mathsf{d}}_I$ *is compatible with the cut elimination procedure* $\rightsquigarrow$.

5 Promotion and higher-order differential operators

In the previous section, we have defined a differential extension of graded linear logic, which is interpreted thanks to exponentials indexed by a monoid of differential operators. This extension is done *up-to promotion*, meaning that we do not incorporate promotion in the set of rules. There are two reasons why it makes sense to leave promotion out of the picture:

- DiLL was historically introduced without it, with a then perfectly symmetric set of rules.

- Concerning semantics, LPDOcc are only defined when acting on functions with finite dimensional codomain: $D : \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R}) \to \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R})$. Introducing a promotion rule would mean extending the theory of LPDOcc to higher-order functions.

In this section, we sketch a few of the difficulties one faces when trying to introduce promotion and dereliction rules indexed by differential operators, and explore possible solutions.

Graded dereliction

Indexing the promotion goes hand-in-hand with indexing the *dereliction*. In Figure 1, we introduced a basic (not indexed) dereliction and codereliction rule d and $\bar{\mathsf{d}}$. The original intuition of DiLL is that codereliction computes the differentiation at 0 of some proof. Following the intuition of $\mathsf{D\text{-}DiLL}$, dereliction computes a solution to the equation $D_0(\_) = \ell$ for some ℓ. Therefore, as indexes are here to *keep track* of the computations, and following equation (2), we should have (co)derelictions indexed by D_0 as below:

$$\frac{\vdash \Gamma, A}{\vdash \Gamma, !A}\ \bar{\mathsf{d}} \qquad \frac{\vdash \Gamma, A}{\vdash \Gamma, ?A}\ \mathsf{d} \qquad\qquad \frac{\vdash \Gamma, A}{\vdash \Gamma, !_{D_0} A}\ \bar{\mathsf{d}}_{D_0} \qquad \frac{\vdash \Gamma, A}{\vdash \Gamma, ?_{D_0} A}\ \mathsf{d}_{D_0}$$

Mimicking what happens in graded logics, D_0 should be the identity element for the second law in the semiring interpreting the indices of exponentials in $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$. However, D_0 is *not* a linear partial differential operator (even less with constant coefficient). Let us briefly compare how a LPDOcc D and D_0 act on a function $f \in \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R})$:

$$D : f \mapsto \left(y \in \mathbb{R}^n \mapsto \sum_{\alpha \in \mathbb{N}^n} a_\alpha \frac{\partial^{|\alpha|} f}{\partial x^\alpha}(y) \right) \qquad D_0 : f \mapsto \left(y \in \mathbb{R}^n \mapsto \sum_{0 \le i \le n} y_i \frac{\partial f}{\partial x_i}(0) \right)$$

where $(x_i)_i$ is the canonical base of $\mathbb{R}^n$, y_i is the i-th coordinate of y in the base $(x_i)_i$, and $a_\alpha \in \mathbb{R}$. To include LPDOcc and D_0 in a single semiring structure, one would need to consider global differential operators generated by:

$$\mathsf{D} : f \mapsto \left((y, v) \mapsto \textstyle\sum_{\alpha \in \mathbb{N}^n} a_\alpha(v) \frac{\partial^{|\alpha|} f}{\partial x^\alpha}(y) \right), \text{ with } a_\alpha \in \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R}).$$

The algebraic structure of such a set would be more complicated, and the composition in particular would not be commutative, and as such not suitable for the first law of a semi-ring which is essential since it ensures the symmetry of the contraction and the cocontraction.

Graded promotion

To introduce a promotion law in $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$, we need to define a multiplicative law $\odot$ on $\mathcal{D}$, with D_0 as a unit. We will write it under a digging form:

$$\frac{\vdash \Gamma, ?_{D_1} ?_{D_2} A}{\vdash \Gamma, ?_{D_1 \odot D_2} A}\ \mathsf{dig}$$

This relates with recent work by Kerjean and Lemay [26], inspired by preexisting mathematical work in infinite dimensional analysis [18]. They show that in particular quantitative models, one can define the exponential of elements of $!A$, such that $e^{D_0} : \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R}) \to \mathcal{C}^\infty(\mathbb{R}^n, \mathbb{R})$ is the identity. It hints at a possible definition of the multiplicative law as $D_1 \odot D_2 := D_1 \circ e^{D_2}$.

Even if one finds a semi-ring structure on the set of all LPDOcc, the introduction of promotion in the syntax means higher-order functions in denotational models. Indexed exponential connectives are defined so-far thanks to the action of LPDOcc on functions with a finite number of variable. To make LPDOcc act on higher order function (*e.g.* elements of $\mathcal{C}^\infty(\mathcal{C}^\infty(\mathbb{R}^n,\mathbb{R}),\mathbb{R})$ and not only $\mathcal{C}^\infty(\mathbb{R}^n,\mathbb{R})$) one would need to find a definition of partial differential operators independent from any canonical base, which seems difficult. Moreover, contrarily to what happens regarding the differentiation of the composition of function, no higher-order version of the chain rule exists for the action of LPDOcc on the composition of functions. A possible solution could come from differentiable programming [7], in which differentials of first-order functions are propagated through higher-order primitives.

As a trick to bypass some of these issues, we could consider that the $!_D$ modalities are not composable. This is possible in a framework similar to the original BLL or that of IndLL [13], where indexes have a source and a target.

6 Conclusion

In this paper, we define a multi-operator version to D-DiLL, which turns out to be the finitary differential version of Graded Linear Logic. We describe the cut-elimination procedure and give a denotational model of this calculus in terms of differential operators. This provides a new and unexpected semantics for Graded Linear Logic, and tighten the links between Linear Logic and Functional Analysis.

There are several directions to explore now that the proof theory of $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$ has been established. The obvious missing piece in our work is the *categorical axiomatization* of our model. In a version with promotion, that would consist in a differential version of bounded linear exponentials [6]. A first study based on with differential categories [2] was recently done by Pacaud-Lemay and Vienney [28]. While similarities will certainly exist in categorical models of $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$, differences between the dynamic of LPDOcc and of differentiation at 0 will certainly require adaptation. In particular, the treatment of the sum will require attention (proof do not need to be summed here while differential categories are additive). Finally, beware that our logic does not yet extend to higher-order and that without a concrete higher-model it might be difficult to design elegant categorical axioms.

Another line of research would consist in introducing more complex differential operators as indices of exponential connectives. Equations involving LPDOcc are extremely simple to manipulate as they are solved in a one step computation (by applying a convolution product with their fundamental solution). The vast majority of differential equations are difficult if not impossible to solve. One could introduce fixpoint operators within the theory of $\mathsf{DB}_{\mathcal{S}}\mathsf{LL}$, to try and modelize the resolution of differential equation by fixed point. This could also be combined with the study of particularly stable classes of differential operators, as *D-finite operators*. We would also like to understand the link between our model, where exponentials are graded with differential operators, with another new model of linear logic where morphisms corresponds to linear or non-linear differential operators [32] [28].

References

1 Jean-Philippe Bernardy, Mathieu Boespflug, Ryan R. Newton, Simon Peyton Jones, and Arnaud Spiwack. Linear haskell: practical linearity in a higher-order polymorphic language. In *Principles of Programming Languages 2018 (POPL 2018)*. ACM, January 2018.

2 Rick Blute, Robin Cockett, and Robert Seely. Differential categories. *Mathematical Structures in Computer Science*, 16(6), 2006.

3 Rick Blute, Thomas Ehrhard, and Christne Tasson. A convenient differential category. *Les cahiers de topologie et de géométrie différentielle catégorique*, 2012.

4 Siegfried Bosch. *Algebra*. Springer-Lehrbuch. Springer, 2009.

5 Flavien Breuvart and Michele Pagani. Modelling Coeffects in the Relational Semantics of Linear Logic. In *Computer Science Logic (CSL)*, Leibniz International Proceedings in Informatics (LIPIcs). Schloss Dagstuhl, 2015.

6 Aloïs Brunel, Marco Gaboardi, Damiano Mazza, and Steve Zdancewic. A core quantitative coeffect calculus. In *Programming Languages and Systems*. Springer Berlin Heidelberg, 2014.

7 Aloïs Brunel, Damiano Mazza, and Michele Pagani. Backpropagation in the Simply Typed Lambda-calculus with Linear Negation. *Principles of Programming Languages*, 2020.

8 Y. Dabrowski and M. Kerjean. Models of Linear Logic based on the Schwartz epsilon product. *Theory and Applications of Categories*, 2020.

9 Ugo Dal Lago and Martin Hofmann. Bounded linear logic, revisited. In *Typed Lambda Calculi and Applications (TLCA)*. Springer Berlin Heidelberg, 2009.

10 Thomas Ehrhard. On Köthe Sequence Spaces and Linear Logic. *Mathematical Structures in Computer Science*, 12(5), 2002.

11 Thomas Ehrhard. Finiteness spaces. *Mathematical Structures in Computer Science*, 15(4), 2005.

12 Thomas Ehrhard. An introduction to differential linear logic: proof-nets, models and antiderivatives. *Mathematical Structures in Computer Science*, 28(7), 2018.

13 Thomas Ehrhard and Antonio Bucciarelli. On phase semantics and denotational semantics: the exponentials. *Annals of Pure and Applied Logic*, 109(3), 2001.

14 Thomas Ehrhard and Laurent Regnier. Differential interaction nets. *Theoretical Computer Science*, 364(2), 2006.

15 Yōji Fukihara and Shin-ya Katsumata. Generalized bounded linear logic and its categorical semantics. In *Foundations of Software Science and Computation Structures (FoSSaCS)*. Springer International Publishing, 2021.

16 Marco Gaboardi, Andreas Haeberlen, Justin Hsu, Arjun Narayan, and Benjamin Pierce. Linear Dependent Types for Differential Privacy. POPL '13. ACM, 2013.

17 Marco Gaboardi, Shin-ya Katsumata, Dominic Orchard, Flavien Breuvart, and Tarmo Uustalu. Combining effects and coeffects via grading. International Conference on Functional Programming, ICFP. Association for Computing Machinery, 2016.

18 R. Gannoun, R. Hachaichi, H. Ouerdiane, and A. Rezgui. Un théorème de dualité entre espaces de fonctions holomorphes à croissance exponentielle. *Journal of Functional Analysis*, 171(1), 2000.

19 Dan Ghica and Alex I. Smith. Bounded linear types in a resource semiring. In *Programming Languages and Systems,*, European Symposium on Programming, (ESOP). Springer Berlin Heidelberg, 2014.

20 Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50(1), 1987.

21 Jean-Yves Girard. Normal functors, power series and λ-calculus. *Annals of Pure and Applied Logic*, (2), 1988.

22 Jean-Yves Girard, Andre Scedrov, and Philip Scott. Bounded linear logic. 9, 08 1991.

23 Lars Hormander. *Linear partial differential operators*. Springer Berlin, 1963.

24 Hans Jarchow. Locally convex spaces. B. G. Teubner Stuttgart, 1981. Mathematical Textbooks.

25 Marie Kerjean. A logical account for linear partial differential equations. In *Logic in Computer Science (LICS), Proceedings*. Association for Computing Machinery, 2018.

26 Marie Kerjean and Jean-Simon Pacaud Lemay. Taylor Expansion as a Monad in Models of Dill, 2023. preprint.

27 O. Laurent. *Etude de la polarisation en logique*. Thèse de Doctorat, Université Aix-Marseille II, March 2002.

28 Jean-Simon Pacaud Lemay and Jean-Baptiste Vienney. Graded differential categories and graded differential linear logic, 2023. preprint.

29 Paul-André Melliès. Parametric monads and enriched adjunctions. 2012.

30 Michele Pagani. The Cut-Elimination Theorem for Differential Nets with Promotion. In *Typed Lambda Calculi and Applications, TLCA 2009, Proceedings*.

31 L. Schwartz. *Théorie des distributions*. Publications de l'Institut de Mathématique de l'Université de Strasbourg, No. IX-X. Hermann, Paris, 1966.

32 James Wallbridge. Jets and differential linear logic. *Mathematical Structures in Computer Science*, 30(8), 2020. `doi:10.1017/S0960129520000249`.