A Theory of Effects and Resources:
Adjunction Models and Polarised Calculi

1. Introduction

The modern study of foundations for programming languages involves looking at the subject from a variety of viewpoints. In particular: (i) syntactic calculi, formalising typing disciplines and computational behaviour, are built; (ii) connections to logical systems, for guiding these developments, are sought; and (iii) models for establishing a mathematical basis, and informing the design space, are investigated. In unison, these activities have the overall aim of discovering deep connections as in the Curry-Howard-Lambek correspondence. It is within this landscape that the work presented here falls, specifically in the context of investigating two prominent aspects of computation: computational effects and resource management.

Computational effects The analysis of computational effects in the above light started with the seminal work of Moggi [34]. There, the notion of monad was put forward as a mathematical abstraction for encapsulating effects and used to extract a metalanguage (in indirect style) in which to give semantic interpretations to call-by-value programming languages (in direct style) with effectful computations.

Prompted by the mathematical analysis of Power and Robin-son [42] investigating adjoint resolutions for the monadic theory of effects, Levy [31] developed CBPV (Call-by-Push-Value), a theory based on adjunction models. The mathematical structure of these models is directly reflected in Levy’s CBPV calculus where there is a dichotomy of type and program structure corresponding to each of the categories involved in an adjoint situation. This results in a paradigm with aspects of both call-by-value and call-by-name computation.

Linear logic The analysis of resources in computation started with Girard’s seminal discovery of their importance in denotational semantics [22]. Model theoretically, it corresponds to shifting attention from cartesian (intuitionistic) to symmetric monoidal (linear) structure, recovering the former from the latter by means of a resource (or exponential) modality; which, after much study [4, 5, 8, 43], was revealed to correspond to a comonad arising from a monoidal adjoint resolution.

The study of effects and resources, respectively uncovering the need for adjoint models giving rise to monads and to comonads, offered the tantalising perspective that these two phenomena could somehow be two sides of the same coin. This possibility was first investigated by Benton and Wadler [7], and more recently further pursued by Egger, Mogelberg and Simpson [16]. Our stand in this respect is that there are in fact other phenomena at play, and that the combination of effects and resources should be considered orthogonally.

Polarisation To understand the computational structures at work, one needs to consider another logical development of Girard: polarisation [23]. In a modern guise, polarisation turns an arbitrary
adjunction into an algebraic structure supporting the composition of morphisms with source and target belonging to either of the categories involved; objects are positive or negative depending on their category of origin. We suggest to understand this as a description of direct-style calculi that generalises from monads to adjunctions the direct-style calculus for the Kleisli category in Moggi’s model.

Girard’s work has been related to continuation-passing style (CPS) and with the investigation of the duality between call by value and call by name in this context [10, 12, 13, 29, 30, 41, 44]. However, in continuity with the work of Zeilberger [48, 49] and Mellies [33], we advocate that the perfect symmetry of Girard’s setting is misleading. Polarisation describes a more general model of computation in which strictness and laziness are attributes of the type [36, 39].

Contents of the paper With the above ingredients at hand, our viewpoint is that the adjoint resolution of a monad describing effects underpins polarisation, which may be independently considered in an intuitionistic (cartesian) or in a linear (symmetric monoidal) setting. In the latter case, one may further incorporate resource structure by means of a monoidal adjoint resolution. This is schematically presented in the figure below.

These sequent calculi provide the type systems for the calculi that we introduce, and for which we give sound interpretations in their corresponding models. We employ the standard naming scheme whereby \( \mathcal{M} \) stands for the core multiplicative fragment \((\text{I}, \otimes, \rightarrow)\), \( \mathcal{E} \) for the consideration of a resource (or exponential) modality (\!), and \( \mathcal{A} \) for additives including units \((\top, 0, \&,, \oplus)\).

Our calculi build on the L-calculus \( \lambda \mu \) relating sequent calculus to abstract machines [10] and their variants with polarities [11, 36, 37, 39]. In particular, building on this earlier work, we demonstrate that a model of computation in which strict and lazy datatypes coexist provides a direct computational interpretation to Levy’s CBPV. Further, we characterise depolarised models in a sense close to that of Mellies and Tabareau [33]: there is a correspondence between proof systems where the cuts associate, calculi where the order of evaluation is unimportant, and models where the enriched adjunction is idempotent.

In summary, our contributions are:

- A comprehensive theory of presheaf-enriched adjunction models, with cartesian, linear, additive, and resource structures as per the diagram, that encompasses CBPV, linear/non-linear adjunction models, dialogue categories, and EEC models.
- A lifting theorem of linear models with resources into cartesian models.
- Polarised L-calculi and corresponding polarised logics, with cartesian, linear, additive, and exponential structure as per the diagram, that are direct-style for the above models, with soundness theorems.
- A characterisation of depolarisation.

Organisation of the paper Section 2 presents the rudiments of the adjunction models needed here and explains polarisation putting it in the current context. The formal treatment starts in Section 3 where the calculus for multiplicative polarised structure, \( \text{MLJ}_{\mu}^\ell \) for cartesian (or intuitionistic) and \( \text{IMLL}_{\mu}^\ell \) for linear, are introduced, with \( \text{MLJ}_{\mu}^\ell \) models presented. Section 4 follows with \( \text{IMLL}_{\mu}^\ell \) models, providing the interpretation of the multiplicative calculus, and the characterisation of depolarisation. Sections 5 and 6 modally extend the development to respectively include resource structure, establishing the lifting theorem, and additive structure, ending with \( \text{ILL}_{\mu}^\ell \) and \( \text{LL}_{\mu}^\ell \) calculi and models. Section 7 concludes with perspectives on novel aspects of our calculus in the context of the operational and equational semantics of effects.

2. Adjunction Models and Polarised Calculi

2.1 Notations for Presheaf-Enriched Adjunction Models

We introduce notation for basic presheaf-enriched structure to be used in the paper; for a comprehensive treatment of enriched category theory the reader may consult Kelly [27].

We are interested in this paper in enriched adjoint situations

\[
\mathcal{V} \xrightarrow{F} \mathcal{X} \xleftarrow{G} \mathcal{S}
\]

where the categories \( \mathcal{V} \) and \( \mathcal{S} \) respectively provide mathematical structure for positive and negative worlds to be dwelled upon in the following section. In this context, we will write \( P, Q \ldots \) for the objects of \( \mathcal{V} \) and \( N, M \ldots \) for the objects of \( \mathcal{S} \). Intuitively, the adjoint functors \( F \) and \( G \) allow the passage between the two worlds \( \mathcal{V} \) and \( \mathcal{S} \).

The categories \( \mathcal{G} \) at play here are to be specifically enriched over presheaves. Roughly speaking, this means that their homs are parametrised as follows

\[
\mathcal{G}_C(X, Y)
\]
and may be thought of as consisting of maps from $X$ to $Y$ in an environment $C$. Furthermore, the parametrisation is such that for every environment morphism $D \rightarrow C$ one contravariantly has functions

$$
\mathcal{G}_c(C, X, Y) \rightarrow \mathcal{G}_c(X, Y)
$$

corresponding to the action of changing environment. For instance, in the cartesian setting, the action $\mathcal{G}_c(C, X, Y) \rightarrow \mathcal{G}_c(X, Y)$ induced by the duplicator map $C \rightarrow C \times C$ corresponds to the operation of contracting the environment $C$.

In the cartesian setting, the identities and composition of $\mathcal{G}$ are given pointwise; that is, one has

$$
id^C \in \mathcal{G}_c(C, X, X), \circ^C_{X,Y,Z} : \mathcal{G}_c(Y, Z) \times \mathcal{G}_c(X, Y) \rightarrow \mathcal{G}_c(X, Z).
$$

In the linear setting there is a need for a more refined approach and one instead has

$$
id^X \in \mathcal{G}_l(X, X), \circ^X_{X,Y,Z} : \mathcal{G}_l(Y, Z) \times \mathcal{G}_l(X, Y) \rightarrow \mathcal{G}_l(X, Z).
$$

This is achieved by means of Day’s convolution monoidal structure on presheaves [14].

The above theory suffices for discussing multiplicative structure. Incorporating additive structure calls for the further refinement of enriching over distributive presheaves. This amounts to requiring that

$$
\mathcal{G}_l(Y, X) \cong 1, \quad \mathcal{G}_c(X, Y) \cong \mathcal{G}_c(X, Y) \times \mathcal{G}_l(Y, X).
$$

In this presheaf-enriched context, the adjunction (1) amounts to giving natural bijections

$$
\mathcal{G}_c(FP, N) \cong \mathcal{I}_c(P, GN)
$$

that are invariant under the environment actions; the unit $\eta$ and the counit $\varepsilon$ of the adjunction are natural families of morphisms $\eta_p \in \mathcal{I}_l(P, GFP)$ and $\varepsilon_N \in \mathcal{G}_l(FGN, N)$, where $I$ intuitively stands for the empty environment. Analogous bijective correspondences are used to describe type/structural invariance in the vein of traditional category theory.

2.2 Polarisation and Calculi

**Polarities: two modes of discourse** Our first description of polarisation was given by Girard [23] and underpins the interpretation of our calculi in adjunction models. Girard gives a denotational semantics to classical sequent calculi identifying $\lnot \rightarrow \rightarrow \rightarrow \rightarrow$ via a concrete interpretation based on coherent spaces, and a corresponding abstract construction, given as a negative translation into intuitionistic logic. The negative translation is made by “carefully distinguishing between negative formulas: simply negated, and positive formulas: doubly negated” [24]. However, there is a subtlety in this definition “since it is possible to consider a doubly negated formula as simply negated”. The interpretation then takes advantage of the focusing properties of connectives discovered earlier by Andreoli [1]—a well-studied aspect of polarisation, but which in our sense comes afterwards.

Here is how Girard’s idea is reflected in our interpretation. A cut:

$$
\Gamma_1 \vdash A, \Gamma_2, A \vdash \Delta
$$

is interpreted in two different ways depending on the polarity of $A$. First, sequents are interpreted in the profunctor of oblique morphisms of the adjunction (1):

$$
\mathcal{S}(\rightarrow, \rightarrow) \cong \mathcal{S}(-, =) \cong \mathcal{S}(-, G, =).
$$

To be more precise, a sequent $\Gamma_1 \vdash A \rightarrow B$ is interpreted as $\mathcal{S}_{\rightarrow}(\rightarrow, A, B)$, where $\cdot \rightarrow \cdot \rightarrow$ suitably add $G$ and $F$ wherever necessary.

Then, if $A$ is positive, the cut is:

$$
\mathcal{S}_{\rightarrow}(\Delta, \rightarrow) \times \mathcal{S}_{\rightarrow}(I, FA) \rightarrow \mathcal{S}_{\rightarrow}(\Delta, I)
$$

and is therefore interpreted by a composition in $\mathcal{S}$:

$$
\mathcal{S}_{\rightarrow}(\Delta, \rightarrow) \times \mathcal{S}_{\rightarrow}(I, FA) \rightarrow \mathcal{S}_{\rightarrow}(\Delta, I)
$$

which can be seen as generalising the Kleisli composition for the monad $GF$. On the other hand, if $A$ is negative, then the cut is:

$$
\mathcal{S}_{\rightarrow}(\rightarrow, \Delta) \times \mathcal{S}_{\rightarrow}(I, A) \rightarrow \mathcal{S}_{\rightarrow}(\rightarrow, I)
$$

and is interpreted by a composition in $\mathcal{S}$:

$$
\mathcal{S}_{\rightarrow}(\rightarrow, \Delta) \times \mathcal{S}_{\rightarrow}(I, GA) \rightarrow \mathcal{S}_{\rightarrow}(\rightarrow, GA)
$$

which now can be seen as generalising the Kleisli composition for the comonad $FG$. A new composition is thus obtained; which is
The calculi of polarisation. Our calculi are based on a second description of polarisation. Danos, Joinet and Schellinx [13] reconstruct Girard’s polarised classical logic as a distinguished way of endowing the classical sequent calculi **LK** with a confluent cut-elimination procedure, the system **LK**$_p$. This is in fact the system in which the order of cut-elimination is preserved by $\eta$-expansions. The calculi **LL**$_p$ and **LL**$_{\eta}$ that we introduce are appropriate variants of **LK**$_p$. The meaning of $\eta$ is explained in the next section, where the support of $\mu\nu$-like L-calculi allow a concise presentation and an elegant theoretical development, along the lines of the $\lambda$-calculi in the standard presentation of Barendregt [3].

On the proof-theoretic side, L-calculi can be understood as term assignments for sequent calculus rather than natural deduction. See [10, 47] for introductions to this aspect. On the side of computation, they can be understood as abstract-machine-like calculi, based on two principles, besides polarisation: the inside-out, defunctionalised representation of contexts is primitive, and language constructs are represented abstractly as solutions to their abstract-machine transitions. See [40] for an introduction based on this aspect.

Danos, Joinet and Schellinx recover other classical systems using annotations on formulae that determine the reduction differently, but do not affect provability. These annotations can be understood as the shifts $\psi, \psi \triangleright A, \psi A$, and $\uparrow A$ are always provably equivalent, albeit not isomorphic. In general, polarised calculi let us describe different reduction strategies using appropriately-shifted types. The fa-
miliar decomposition of call-by-value and call-by-name functional types, as given by Levy [31] and others, can thus be retrieved as the composition of such annotations with the construction described previously.

The two qualities we mentioned, compatibility with η-expansions and derivability of known systems, are two main features of the range of calculi that we present. This parallels the general and abstract character of their enriched adjunction models.

3. Cartesian Polarisated Structure

We start the development by focusing on the multiplicative intuitionistic (or cartesian) setting which we refer to as MLJp. Type-theoretically, it corresponds to the unit, strict product, and arrow types. This is model-theoretically simpler than the linear (or symmetric monoidal) variant to be considered next.

3.1 MLJp and IMLLp Calculi

The calculus MLJp (polarised multiplicative intuitionistic logic) is introduced in Figures 1 and 2, at the same time as IMLLp (polarised multiplicative intuitionistic linear logic). On the proof-theoretic side, MLJp corresponds to intuitionistic sequent calculus given with conjunction in multiplicative style (written ⊗), and endowed with a polarised cut-elimination procedure. We write the arrow type as →, which is to be understood as ← or → depending on whether the setting is IMLLp or MLJp. (In this paper, the symbol → does not refer to implication but to a compatible closure.)

Barendregt-style λ-calculus The calculi are introduced along the lines of Barendregt’s presentation of the λ-calculus [3]. There is a distinction to make between the latter and the Church-style λ-calculus: in Church style, well-typed terms are directly defined by induction, whereas in Barendregt style, first a syntax of pseudo-terms is given, and then the legal terms are defined as those pseudo-terms that are well-typed. In Church style, type annotations are located on variables, whereas in Barendregt style, they are located on binders.

The Barendregt technique consists in proving a Basis Lemma (judgements assign a type to expressions and values on the right, and a distinguished type to contexts and stacks on the left, while commands have no type by themselves. Expressions and contexts determine the active (or main) type in the sequent.)

The type system is given by rules of derivation that we obtain are the compatible closure of the theorems in Figures 1 and 2, at the same time as IMLLp (polarised multiplicative intuitionistic linear logic). On the proof-theoretic side, MLJp corresponds to intuitionistic sequent calculus given with conjunction in multiplicative style (written ⊗), and endowed with a polarised cut-elimination procedure. We write the arrow type as →, which is to be understood as ← or → depending on whether the setting is IMLLp or MLJp. (In this paper, the symbol → does not refer to implication but to a compatible closure.)

The Barendregt technique consists in proving a Basis Lemma (judgements assign a type to expressions and values on the right, and a distinguished type to contexts and stacks on the left, while commands have no type by themselves. Expressions and contexts determine the active (or main) type in the sequent.)

The type system is given by rules of derivation that we obtain are the compatible closure of the theorems in Figures 1 and 2, at the same time as IMLLp (polarised multiplicative intuitionistic linear logic). On the proof-theoretic side, MLJp corresponds to intuitionistic sequent calculus given with conjunction in multiplicative style (written ⊗), and endowed with a polarised cut-elimination procedure. We write the arrow type as →, which is to be understood as ← or → depending on whether the setting is IMLLp or MLJp. (In this paper, the symbol → does not refer to implication but to a compatible closure.)

The Barendregt technique consists in proving a Basis Lemma (judgements assign a type to expressions and values on the right, and a distinguished type to contexts and stacks on the left, while commands have no type by themselves. Expressions and contexts determine the active (or main) type in the sequent.)

The type system is given by rules of derivation that we obtain are the compatible closure of the theorems in Figures 1 and 2, at the same time as IMLLp (polarised multiplicative intuitionistic linear logic). On the proof-theoretic side, MLJp corresponds to intuitionistic sequent calculus given with conjunction in multiplicative style (written ⊗), and endowed with a polarised cut-elimination procedure. We write the arrow type as →, which is to be understood as ← or → depending on whether the setting is IMLLp or MLJp. (In this paper, the symbol → does not refer to implication but to a compatible closure.)
\[ (P \otimes Q)(Z) = \int_{X,Y,Z} P(X) \times Q(Y) \times \mathcal{L}(Z, X \otimes Y) . \]

We henceforth restrict attention to the case when the convolution monoidal structure on \( \mathcal{L} \) is closed, which precisely happens when the presheaves \( \mathcal{L}(\cdot \otimes X, Y) \) are small for all \( X, Y \in \mathcal{L} \) [15]; as for instance holds when \( \mathcal{L} \) is small or when it is closed.

The category \( \mathcal{P} \mathcal{L} \), being assumed closed, underlies a \( \mathcal{P} \mathcal{L} \)-category \( \mathcal{P} \mathcal{L} \) and the Yoneda embedding induces a \( \mathcal{P} \mathcal{L} \)-enrichment on \( \mathcal{L} \).

Indeed, the category \( \mathcal{L} \) underlies the \( \mathcal{P} \mathcal{L} \)-category \( \mathcal{L} \) with objects that of \( \mathcal{L} \) and hom-presheaves given by the exponential structure of \( \mathcal{P} \mathcal{L} \); that is, with \( \mathcal{L}(X, Y) = \mathcal{P} \mathcal{L}(X, Y) \). In particular, for cartesian \( \mathcal{L} \), one has that \( \mathcal{L}(X, Y) = \mathcal{L}(Z \times X, Y) \). The Yoneda embedding enriches to a \( \mathcal{P} \mathcal{L} \)-functor.

\( \mathcal{P} \mathcal{L} \)-category is said to be powered whenever it has powers; represented, for presheaves \( P \) in \( \mathcal{P} \mathcal{L} \) and objects \( A \) in \( \mathcal{L} \), by an object \( [P \to A] \) in \( \mathcal{L} \) together with an isomorphism
\[
\mathcal{L}(C, [P \to A]) \cong \mathcal{P} \mathcal{L}(P, \mathcal{L}(C, A)) .
\]

\( \mathcal{P} \mathcal{L} \)-natural for \( C \) in \( \mathcal{L} \). Powers with respect to representable presheaves are referred to as \( \mathcal{L} \)-powers.

A \( \mathcal{P} \mathcal{L} \)-adjunction \( \mathcal{L} \overset{\alpha}{\leftarrow} \mathcal{L} \) consists of \( \mathcal{P} \mathcal{L} \)-functors together with a representation
\[
\mathcal{L}(A, B) \cong \mathcal{L}(A, GB) .
\]

natural for \( A \) in \( \mathcal{L} \) and for \( B \) in \( \mathcal{L} \).

\subsection*{Adjunction models}

The notion of model for \( \text{ML}_{\mathcal{L}} \) follows; it coincides with that of EC model of Egger, Mogelberg and Simpson [16].

\textbf{Definition 4.} An \( \text{ML}_{\mathcal{L}} \) model consists of a cartesian category \( \mathcal{L} \) and a \( \mathcal{P} \mathcal{L} \)-category \( \mathcal{L} \) with \( \mathcal{L} \)-powers together with a \( \mathcal{P} \mathcal{L} \)-adjunction \( \mathcal{L} \overset{\alpha}{\leftarrow} \mathcal{L} \).

To aid the understanding of this structure, we remark that in elementary terms a \( \mathcal{P} \mathcal{L} \)-adjunction \( \mathcal{L} \) is specified by a class of objects \( \mathcal{L} \) and hom-presheaves \( \mathcal{L}(M, N) \) in \( \mathcal{L} \) for all \( M, N \in \mathcal{L} \) together with identities in \( \mathcal{L}(N, N) \) and composition operations
\[
\mathcal{L}(M, N) \times \mathcal{L}(N, N) \to \mathcal{L}(N, N)
\]

natural for \( P, Q \) in \( \mathcal{L} \) and subject to monoid laws.

In addition, \( \mathcal{L} \)-powers are represented, for objects \( P \) in \( \mathcal{L} \) and \( N \) in \( \mathcal{L} \), by an object \( [P \to N] \) in \( \mathcal{L} \) together with an isomorphism
\[
\mathcal{L}(M, [P \to N]) \cong \mathcal{L}(G \mathcal{L}(P, M, N), N)
\]

natural for \( Q \) in \( \mathcal{L} \) and for \( M \) in \( \mathcal{L} \). Therefore, there are evaluation maps \( \mathcal{L}(P \to N) \to_p N \) in \( \mathcal{L} \).

\textbf{Lemma 5.} For every \( P \) in \( \mathcal{L} \) and \( N \) in \( \mathcal{L} \), we have an isomorphism
\[
\mathcal{L}(Q \otimes P, GN) \cong \mathcal{L}(Q, G(P \to N))
\]

natural for \( Q \) in \( \mathcal{L} \). Therefore, there are evaluation maps \( G(P \to N) \to_p GN \) in \( \mathcal{L} \).

\textbf{Example 6.} Every symmetric monoidal closed category \( \mathcal{L} \), powered \( \mathcal{L} \)-category \( \mathcal{L} \), and \( \mathcal{P} \mathcal{L} \)-adjunction \( \mathcal{L} \overset{\alpha}{\leftarrow} \mathcal{L} \) provide an \( \text{ML}_{\mathcal{L}} \) model. Some examples include:

1. \( \mathcal{L} = \text{Set} \) with the monoidal-closure adjunction \( (\cdot) \otimes P \to [P \to (\cdot)] \) for \( \mathcal{L} \). and \( p \) for \( \mathcal{L} \).

2. \( \mathcal{L} = \text{Set} \) with the linear-double-dualisation adjunction \( [- \to -] \) for \( \mathcal{L} \). 

\textbf{Remark 7.} The second example above was considered in the linear case by Mellies and Tabareu [33]. It has extra structure in the form of an obvious strictly involutive negation \( \mathcal{L} \otimes \mathcal{L} \) and thus gives models of polarised classical and linear logic. Its proof theory in fact led to the introduction of polarisation [23].

Corresponding systems \( \text{LL}_{\mathcal{L}} \) (essentially Girard’s LJC, see [13]) and \( \text{LL}_{\mathcal{L}} \) can be deduced from \( \text{LL}_{\mathcal{L}} \) by adding a (strict or not) involutive negation by following the ideas in [36, 38]. This perspective is novel compared to the original intuitions involving "linear continuations" [17]; it does not presuppose that evaluation order is irrelevant and, as we will see, is more faithful to the focusing properties of linear logic.

4. Linear Polarised Structure

We now focus on the linear multiplicative setting, whose calculus \( \text{IMLL}_{\mathcal{L}} \) was introduced in the previous section. We give its models and interpretation.

\textbf{4.1 IMLL}_{\mathcal{L}} Models}

Models of \( \text{IMLL}_{\mathcal{L}} \) refine \( \text{ML}_{\mathcal{L}} \) models by weakening the cartesian structure to being linear, i.e. symmetric monoidal. The main technical tool needed to do so is a canonical symmetric monoidal structure on presheaf categories over symmetric monoidal categories.
4.2 IMLL\textsuperscript{g} Semantics

We present the categorical interpretation of IMLL\textsuperscript{g}.

**Interpretation of types**
To every type \(A\), we associate both a positive interpretation \(A^+ \in \mathcal{C}\) and a negative interpretation \(A^- \in \mathcal{C}\). These are defined by mutual induction as follows:

- \((X^+)\) is an assigned object of \(\mathcal{C}\); \(X = I; (A \otimes B)^+ = A^+ \otimes B^+;\) and \(N^+ = GN^+\) for every negative type \(N;\)

- \((X^-)\) is an assigned object of \(\mathcal{C}\); \(X \rightarrow B^- = \left[ A^+ \rightarrow B^+ \right];\) and \(P^- = FP^+\) for every positive type \(P;\)

The interpretation of types extends pointwise to typing contexts as follows:

- \((x_1 : A_1, \ldots, x_n : A_n) = (\cdots (I \otimes A_1^-) \otimes \cdots) \otimes A_n^+;\)
- \((a : A) = A^-;\)

**Interpretation of judgments**
Values and stacks are interpreted in the categories \(\mathcal{C}\) and \(\mathcal{S}\), respectively:

- \([\Gamma \vdash V : A]; : I^+ \rightarrow A^+\) in \(\mathcal{C},\)
- \([\Gamma ; S : A \vdash \Delta]; : A^- \rightarrow \Gamma \left[ A \right] \left[ \Delta \right] \left[ \mathcal{S} \right] \left[ \mathcal{C} \right]\)

As for expressions, contexts and commands, judgements may be equivalently interpreted as morphisms in \(\mathcal{C}(-, G\mathcal{M})\) or in \(\mathcal{S}(F, =)\), as explained in Section 3.1. In fact we fix it arbitrarily as follows:

- \([\Gamma \vdash t : A]; : I^+ \rightarrow GA^+\) in \(\mathcal{C},\)
- \([\Gamma ; e : A \vdash \Delta]; : FA^+ \rightarrow \Gamma \left[ A \right] \left[ \Delta \right] \left[ \mathcal{S} \right] \left[ \mathcal{C} \right];\)
- \([\Gamma ; c : (\Gamma ; x : A) ; : I^+ \rightarrow GA^+\) in \(\mathcal{C},\)

The interpretation is defined by mutual induction on derivations as follows:

Coercions:

- \([\Gamma \vdash V^+ : A]; : I^+ \rightarrow GA^+\) is obtained from \([\Gamma \vdash V : A]; : I^+ \rightarrow A^+;\) as this morphism for \(A\) negative and, for \(A\) positive, by post-compositing with it \(A^+ \equiv I \otimes A^+ \rightarrow GFA^+\) where the second map is the unit;

- \([\Gamma ; S : A \vdash \Delta]; : FA^+ \rightarrow \Gamma \left[ A \right] \left[ \Delta \right] \left[ \mathcal{S} \right] \left[ \mathcal{C} \right];\)

Identity rules:

- \([\Gamma ; \mu A : A]; : I^+ \rightarrow GA^+\) is the canonical morphism for \(A\) positive and, for \(A\) negative, by pre-compositing with it \(GA^+ \rightarrow A^+;\)

- \([\Gamma ; \mu x A : A]; : FA^+ \rightarrow \Gamma \left[ A \right] \left[ \mu A \right] \left[ \Delta \right] \left[ \mathcal{S} \right] \left[ \mathcal{C} \right];\)

Structural rules:

- \([\Gamma ; \mu x A : A]; : I^+ \rightarrow GA^+\) is obtained from \([\Gamma ; \mu x A : A]; : I^+ \rightarrow GA^+\) post-composed with \(A^+ \otimes A^+ \rightarrow GB^+\) by means of Lemma 5;

- \([\Gamma ; \mu x A : A]; : I^+ \rightarrow GA^+\) is obtained from the evaluation map \(A^+ \rightarrow B^+\) \(\rightarrow \Gamma \left[ A \right] \left[ \mu A \right] \left[ \Delta \right] \left[ \mathcal{S} \right] \left[ \mathcal{C} \right];\)

Theorem 9. The IMLL\textsuperscript{g} semantics is coherent: all derivations of a judgement have the same interpretation; and sound: if \(c \rightarrow_{RE} c': (\Gamma \vdash \Delta)\) then \(\left[ c : (\Gamma \vdash \Delta) \right] \left[ \mathcal{C} \right] = \left[ c' : (\Gamma \vdash \Delta) \right] \left[ \mathcal{C} \right];\)

The key to these results is an extension of the Generation Lemma with the preservation of the denotations in any model, which shows that the interpretation is in fact determined by induction on the terms.

4.3 MLJ\textsuperscript{g} Semantics

The categorical interpretation of MLJ\textsuperscript{g} derivations in its models from Section 3.2 extends that of IMLL\textsuperscript{g} using the canonical cartesian interpretation \([\sigma]; : I^+ \rightarrow \Gamma^\sigma \equiv \Gamma^\sigma \left[ \mathcal{C} \right] \left[ \mathcal{S} \right] \left[ \mathcal{C} \right];\)

Theorem 10. The MLJ\textsuperscript{g} semantics is coherent and sound.
4.4 Depolarisation

Polarisation, in IMLL_p, IMELL_p, and their extensions, is represented by the fact that the following equations (here given with an orientation) are not always derivable from $\vdash_{RE}$ when $A$ is positive and $B$ is negative:

\( \{ t \parallel \mu x \langle \mu a. c \parallel e \} \vdash_{L} \{ \mu a. \langle t \parallel \mu x \parallel c \rangle \parallel e \} \)  
\( \{ t \parallel \mu x \parallel c \} \vdash_{L} \{ c[t/x] \} \)  
\( \{ c[e/a] \} \vdash_{L} \{ \mu a.c \parallel e \} \)

The equation $\vdash_{\chi}$ corresponds to an associativity of cuts. The notations $[t/x]$ and $[e/a]$ refer to the substitutions of $t$ for $x$ and of $e$ for $a$ under the condition that every occurrence of $x$ is of the form $\langle x \parallel S \rangle^\ast$, and every occurrence of $a$ is of the form $(V \parallel a)^\ast$. (They can equivalently be defined as standard substitutions, in the sense of higher-order rewriting, by adding the so-called “\(\zeta\)-rules”.)

Lemma 11. Let $A$ be a type and $t$ be an expression. We denote with $\vdash_{A_0}$ and $\vdash_{L}$ the relations given respectively by (2) and (3) quantified over arbitrary $B$, $c$ and $e$. Symmetrically, let $B$ be a type and $e$ be a context. We denote with $\vdash_{A_0}$ and $\vdash_{L}$ the relations given respectively by (2) and (4) quantified over arbitrary $A$, $c$ and $t$. One has:

\[
(\vdash_{RE} \cdot \vdash_{A_0} \cdot \vdash_{RE}) = (\vdash_{RE} \cdot \vdash_{T_0} \cdot \vdash_{RE})
\]

\[
(\vdash_{RE} \cdot \vdash_{A_0} \cdot \vdash_{RE}) = (\vdash_{RE} \cdot \vdash_{L_0} \cdot \vdash_{RE})
\]

With the additional constraint that $c[\mu a. \langle y \parallel (\langle a \rangle /x) \rangle]$ (in the case of $\vdash_{\chi}$) and $c[\mu x. \langle \beta \parallel (\langle a \rangle /a) \rangle]$ (in the case of $\vdash_{A_0}$) are typable, these equalities restrict to the corresponding typed relations between legal terms.

In particular, $\vdash_{A_0}$, $\vdash_{T_0}$, and $\vdash_{L_0}$ are equal modulo $\vdash_{RE}$. Notice that the final typing constraint is a linearity constraint in the case of IMLL_p and its extensions. A model is *depolarised* if it validates either of these equivalent equation schemes. The following theorem characterises depolarised models and extends the characterisation in [39].

Theorem 12. Let $\mathcal{L} \xleftarrow{\oplus} \mathcal{S}$ be an IMLL_p model. Every interpretation satisfies the typed restriction of (2) if and only if the adjunction is idempotent.

This result generalises the depolarisation criterion for dialogue categories [39]. There, categorical models of linear logic are built from polarised ones by enforcing commutativity of the strong monad $\sim$. For the adjunction $\sim \dashv \sim$, being idempotent is equivalent to $\sim$ being commutative (Führmann [21]).

Example 13. Erratic choice [31, Section 5.5], based on the powerset monad, is an example of a commutative, but non-idempotent effect.

5. Resource Modalities

We now focus on resource structure relating linear models to cartesian ones. In the calculus, we give a treatment of the exponential modality $!$ consistent with its focusing properties. Its sound interpretation by means of resource structure extends that of Melliès and Tabareau [33].

5.1 IMELL_p Calculus

IMELL_p is defined in Figure 3 by adding to IMLL_p the exponential modality $!$ from linear logic [22]. Unlike the other connectives, the pattern-matching form $(\mu \langle a \parallel e \rangle)$ is a positive expression; it corresponds to promotion. For this reason, it is difficult to provide syntactic sugar for IMELL_p in $\lambda$-calculus style, as was noticed for linear logic in its early days [4, 5, 45, 46]. A main novelty of our syntax is that contraction and weakening are treated by merging and introducing variables in the context, instead of introducing explicit grammatical counterparts to the logical rules. This treatment is similar to [36] but is now substantiated by a semantic interpretation.

Our definition reflects the complex focalisation properties of $!$: almost invertible on the right (meaning that the rule $E!$ is ill-typed unless the context is of the form $!V$) and non-invertible on the left. A key property to notice is that if $\Gamma \vdash V : !A$, then $\Gamma$ is of the form $\Gamma'$, by analysis on the derivation. Thus, restricting the rule $E!$ to values ensures that it is valid. For the same reason, any reduction $\langle V \parallel \mu \times x \parallel c \rangle \rightarrow_{\beta} c[V/x]$, which may duplicate or erase $V$ due to contractions or weakening in $x$, is valid because the same contractions or weakening apply to the typing context of $V$, necessarily of the form $\Gamma'$.

Theorem 14. IMELL_p is expressible in IMELL_p.

Theorem 15. IMELL_p has the Barendregt-style properties.

5.2 IMELL_p Models

Adjunction models We extend IMLL_p models with resources in the form of a monoidal adjunction; when this is linear/non-linear [4] it describes IMELL_p models.

Definition 16. An IMLL_p model with a resource modality consists of an IMLL_p model $\mathcal{L} \oplus \mathcal{S}$ together with a symmetric monoidal category $\mathcal{K}$ and a monoidal adjunction $\mathcal{K} \xrightarrow{\oplus} \mathcal{L}$. An IMELL_p model is an IMLL_p model with a resource modality in which the symmetric monoidal structure of $\mathcal{K}$ is cartesian.

Example 17. The dialogue categories with a resource modality of Melliès and Tabareau [33] are IMLL_p models with a resource modality.

Semantics For an IMLL_p model with a resource modality as above, we write $E$ for the monoidal comonad on $\mathcal{L}$ induced by the monoidal adjunction, and define:

$$(!A)^t \equiv EGA^t$$

The terms associated to the exponential are then interpreted as follows:

- $[[\Gamma] \vdash \mu \langle a \parallel e \rangle : !A]] : EGA^t \rightarrow EGA^t$ is obtained by pre-composing $E[: \langle \{ \Gamma \parallel a \langle A \rangle \} ] : EEG^t \rightarrow EGA^t$ with the comultiplication $EG^t \rightarrow EEG^t$ of $E$.

- $[[\Gamma] ; !S : A \vdash A] : FEGA^t \rightarrow GA^t$ arises from the map $[[\Gamma] ; S : A \vdash A] : A^t \rightarrow A^t$ by pre-composition with the composite $FEA^t \rightarrow FGA^t \rightarrow A^t$ where the first map is obtained from the application of $F$ to the map $I \oplus EGA^t \rightarrow GA^t$ arising from the counit of $E$ and the second map is the counit of $F \rightarrow G$.

Assuming that the model is one of IMELL_p, the interpretation of the structural rules is as before relying on a functorial interpretation $[[\sigma]] : \Gamma^+ \rightarrow \Gamma^+$ of maps $\sigma \in \Sigma(\Gamma; \Gamma^+)$:

- $\langle \mu x_1 : A, \ldots, x_n : A \rangle \rightarrow (x : A)$ are interpreted by $\langle (x : A)^t \rangle \equiv (A^t)^{\oplus n} \equiv (x_1 : A, \ldots, x_n : A)^t (n \geq 0)$
IMELL_\textit{\pi} = IMLL_\textit{\pi} + exponentials as follows:

\[ P, Q ::= \ldots | !A \]

(a) Types

values \( V, W ::= \ldots | \mu a^A c \)

stacks \( S ::= \ldots | !S \)

(b) Pseudo-terms

\[(R)! \quad (\mu a^A c \parallel !S) \quad \triangleright \quad c[S/a] \]

\[(E)! \quad \mu a^A c \parallel !a \quad \triangleright \quad V \]

(c) Reduction and extensionality rules

TYING RULES and DERIVABLE RULES (LEGAL TERMS)

- \( \Sigma_c(\Gamma; \Gamma') \) is the subset of \( \Sigma(\Gamma; \Gamma') \) consisting of the maps that restrict to a bijection on the variables that are not of the type \(!A\).

\[
\begin{array}{rcl}
\Gamma \vdash \sigma : A & \text{to} & \Gamma \vdash \sigma : !A; \\
\Gamma \vdash \Delta & \text{to} & \Gamma \vdash \Delta \\
\end{array}
\]

and the rules from Figure 2c extended to all \( \sigma \in \Sigma_c(\Gamma; \Gamma') \).

(d) Structure

\[
\begin{array}{rcl}
\Gamma \vdash e : A & \text{to} & \Gamma \vdash e : !A; \\
\Gamma \vdash !A & \text{to} & \Gamma \vdash !A \\
\end{array}
\]

(e) Remaining rule of sequent calculus

Figure 3. Exponentials

where \( \delta^n \) is the identity for \( n = 1 \) and otherwise, writing \( L \vdash K \) for the adjoint resolution of \( E \), given by the canonical maps

\[ LKP \xrightarrow{\delta^x_{\sigma}} L((KP)^\theta) \cong (LKP)^{\theta^n} \]

with \( P = GB^\perp \) for \( A = LB \).

\bullet arbitrary maps \( \sigma : \Gamma \longrightarrow (x_1 : A_1, \ldots, x_n : A_n) \) are interpreted by

\[
(x_1 : A_1, \ldots, x_n : A_n)^+ \equiv (x_1 : A_1)^+ \otimes \cdots \otimes (x : A_n)^+
\]

\[
\delta_{x_1 \otimes \cdots \otimes x_n} \quad (\Gamma_1^+ \otimes \cdots \otimes \Gamma_n^+) \cong \Gamma^+
\]

where \( \Gamma^+ \) is the restriction of \( \Gamma \) to \( x^{-1}(\chi) \).

**Theorem 18.** The IMELL_\textit{\pi} semantics is coherent and sound.

In the particular case of dialogue categories, this interpretation corresponds, when taking into account the presence of an involutive negation, to Melliès and Tabareau’s “focalised” translation of linear logic into tensor logic [33], but for the omission of a shift in their definition that our interpretation corrects.

In fact, this interpretation suggests a decomposition of the exponential \(!A\) in the calculus as \( \zeta \) for \( A \) where \( (\zeta A)^+ = EA^+ \), not investigated here. This new connective \( \zeta \) has to be understood as a proto-exponential whose promotion rule is restricted to values. Like for the value restriction of polymorphism, it is the constraint on the typing context that prevents the immediate lifting of this restriction to values.

5.3 Lifting Theorem

We establish a model-theoretic lifting theorem for IMELL_\textit{\pi} models with a resource modality. In the context of exponential structure it corresponds to a Girard decomposition of cartesian into linear structure.

Every functor \( L : \mathcal{K} \rightarrow \mathcal{L} \) induces the adjoint situation \( L \dashv L^* : \mathcal{P}^\mathcal{K} \rightarrow \mathcal{P}^\mathcal{L} \) where \( L^* \) is the universal cocontinuous extension of \( L \) and \( L^* P = PL \).

Further, if \( L \) is a (symmetric) strong monoidal functor, as it is the case when \( L \) is a (symmetric) monoidal left adjoint [26], then the adjunction \( L \dashv L^* \) is (symmetric) monoidal [25].

In this situation, as it happens with every monoidal functor, one obtains a 2-functor mapping a \( \mathcal{P}^\mathcal{K} \)-category \( \mathcal{C} \) to the \( \mathcal{P}^\mathcal{L} \)-category \( L^* \mathcal{C} \) with objects those of \( \mathcal{C} \) and hom-pre-sheaves \( L^* \mathcal{C}(A, B) = L^*(\mathcal{C}(A, B)) \). This construction has the following closure property:

if \( \mathcal{C} \) has \( \mathcal{L} \)-powers \( [\_ \smallfrown \_] = \) then \( L^* \mathcal{C} \) has \( \mathcal{K} \)-powers \( [\_ \smallfrown \_] \).

These are given by

\[ [X \smallfrown B] = [LX \smallfrown (\_ \smallfrown B)] \]

where \( L^* \mathcal{C}(A, [LB \smallfrown (\_ \smallfrown B)]) \cong L^* \mathcal{C}(A, B) \) \( \cong \mathcal{K}(LX, L^* A, B) \).

Furthermore, we have a \( \mathcal{P}^\mathcal{K} \)-functor \( L : \mathcal{K} \rightarrow L^* \mathcal{L} \) with object mapping of \( L \) and hom-actions given by

\[
\begin{array}{c}
\mathcal{K}(X, Y) \longrightarrow L^* \mathcal{K}(LX, LY) \\
\mathcal{K}(X \smallfrown Y) \rightarrow \mathcal{K}(L(X \smallfrown Y), LY) \rightarrow L^* \mathcal{K}(L(X \smallfrown Y), LY)
\end{array}
\]

with the property that \( L \) is a left \( \mathcal{P}^\mathcal{K} \)-adjoint whenever \( L \) is a monoidal left adjoint. In particular, note that for a monoidal adjunction \( L : K \rightarrow \mathcal{K} \) one has

\[ L^* \mathcal{K}(LX, Y) \cong L^* (L(X) \smallfrown Y) \cong L^*(L(X \smallfrown Y)) \cong \mathcal{K}(X \smallfrown Y, K) \]

To summarise, we have the following lifting theorem.

**Theorem 19.** Every IMLL_\textit{\pi} model with a resource modality

\[
\begin{array}{c}
\mathcal{K} \xrightarrow{L} \mathcal{L} \\
\mathcal{L} \xrightarrow{L^*} \mathcal{L}
\end{array}
\]

induces an IMELL_\textit{\pi} model, obtained by composing the following two adjunctions:

\[
\begin{array}{c}
\mathcal{K} \xrightarrow{L^* \mathcal{K}} \mathcal{L}^* \xrightarrow{L^*} \mathcal{L}^* \mathcal{K}
\end{array}
\]

**Corollary 20.** Every IMELL_\textit{\pi} model \( \mathcal{L} \xrightarrow{L} \mathcal{K} \xrightarrow{L^*} \mathcal{L} \)

induces an MILL_\textit{\pi} model \( \mathcal{L} \xrightarrow{L^*} \mathcal{L} \).

6. Additive Structure

We now consider additive connectives, that is to say positive co-products and negative products. Computationally, they correspond to strict sums and lazy pairs. The semantic subtlety is due to co-products, whereas the model theory of the negative cartesian product is simpler and could have been introduced earlier in Section 4.
They are grouped here mainly for their similarity on the calculus side.

6.1 IMALL⁺ Calculus

IMALL⁺ is obtained by adding to IMLL⁺ the additives of Figure 4: binary connectives & and ⊕, and units ⊤ and 0.

SUMS

Sums provide a striking example of the simplification brought by L-calculi [40]. For instance, a commuting conversion of the ∆-calculus with sums, as:

δ(t∧B,x∧u,y∧v) \rightarrow V ∧C ∋ δ(t∧B,x∧u,y∧v)∧D

(where x, y \not\in \text{fv}

V) is redundant with the definition in IMALL⁺. Its extensions:

δ(t∧B,x∧u,y∧v) \equiv \mu a.C \langle t \parallel \mu [x∧u \parallel [α]x∧y \parallel [α]v \parallel V ∧C] \rangle

Indeed, in context α, both sides of the above equation reduce to:

\langle t \parallel \mu [x∧u \parallel [α]x∧y \parallel [α]v \parallel V ∧C] \rangle

which can unlock further reductions involving V ∧ C.

UNITS

Because of linearity, additive units require a special treatment. Instead, a naive approach is to introduce the syntax μ< α > and μ(α) (nullary variants of μ< α, α >; β; c > and μ[α, c] [y, α]) together with the following rules:

\Gamma \vdash μ< α > : \top;

\Gamma ; μ(α) : 0 \vdash ∆

However, this gives an incomplete system: the judgement x∧A, y∧B \vdash μ< α >; μ< α > : \top ∧C; \top has two derivations with possibly distinct interpretations (viz. arising from τ∧A, ∧C; τ∧A, ∧C; and τ∧A, ∧C, ∧C; τ∧A, ∧C, ∧C; where τ∧A : \top ∧C → G1 is the transpose of the unique map FP → 1).

The syntax μ[V] and μ[S] and the rules (τ∧A, ∧C; τ∧A, ∧C; and τ∧A, ∧C; τ∧A, ∧C; and τ∧A, ∧C; τ∧A, ∧C, ∧C; τ∧A, ∧C, ∧C; where τ∧A : \top ∧C → G1 is the transpose of the unique map FP → 1).

6.2 Arranging proofs

As we hinted above, a key principle underlying the design of IMALL⁺ is the minimality and efficiency of its axioms. Indeed, IMALL⁺ is the smallest strictly proof-theoretically complete intuitionistic linear logic. This is the result of a careful analysis of the interplay between the structural properties of the underlying calculus and the logical rules.

Figure 4. Additives

They are grouped here mainly for their similarity on the calculus side.

6.1 IMALL⁺ Calculus

IMALL⁺ is obtained by adding to IMLL⁺ the additives of Figure 4: binary connectives & and ⊕, and units ⊤ and 0.

SUMS

Sums provide a striking example of the simplification brought by L-calculi [40]. For instance, a commuting conversion of the ∆-calculus with sums, as:

\delta(t\land B, x\land u, y\land v) \rightarrow V \lor C \ni \delta(t\land B, x\land u, y\land v) \lor D

(where x, y \not\in \text{fv}

V) is redundant with the definition in IMALL⁺. Its extensions:

\delta(t\land B, x\land u, y\land v) \equiv \mu a.C \langle t \parallel \mu [x\land u \parallel [α]x\land y \parallel [α]v \parallel V \lor C] \rangle

Indeed, in context α, both sides of the above equation reduce to:

\langle t \parallel \mu [x\land u \parallel [α]x\land y \parallel [α]v \parallel V \lor C] \rangle

which can unlock further reductions involving V \lor C.

UNITS

Because of linearity, additive units require a special treatment. Instead, a naive approach is to introduce the syntax μ< α > and μ(α) (nullary variants of μ< α, α >; β; c > and μ[α, c] [y, α]) together with the following rules:

\Gamma \vdash μ< α > : \top;

\Gamma ; μ(α) : 0 \vdash ∆

However, this gives an incomplete system: the judgement x\land A, y\land B \vdash μ< α >; μ< α > : \top \land C; \top has two derivations with possibly distinct interpretations (viz. arising from τ\land A, \land C; τ\land A, \land C; and τ\land A, \land C, \land C; τ\land A, \land C, \land C; where τ\land A : \top \land C → G1 is the transpose of the unique map FP → 1).

The syntax μ[V] and μ[S] and the rules (τ\land A, \land C; τ\land A, \land C; and τ\land A, \land C; τ\land A, \land C; and τ\land A, \land C, \land C; τ\land A, \land C, \land C; where τ\land A : \top \land C → G1 is the transpose of the unique map FP → 1).

6.2 Arranging proofs

As we hinted above, a key principle underlying the design of IMALL⁺ is the minimality and efficiency of its axioms. Indeed, IMALL⁺ is the smallest strictly proof-theoretically complete intuitionistic linear logic. This is the result of a careful analysis of the interplay between the structural properties of the underlying calculus and the logical rules.

Figure 4. Additives

They are grouped here mainly for their similarity on the calculus side.

6.1 IMALL⁺ Calculus

IMALL⁺ is obtained by adding to IMLL⁺ the additives of Figure 4: binary connectives & and ⊕, and units ⊤ and 0.

SUMS

Sums provide a striking example of the simplification brought by L-calculi [40]. For instance, a commuting conversion of the ∆-calculus with sums, as:

\delta(t\land B, x\land u, y\land v) \rightarrow V \lor C \ni \delta(t\land B, x\land u, y\land v) \lor D

(where x, y \not\in \text{fv}

V) is redundant with the definition in IMALL⁺. Its extensions:

\delta(t\land B, x\land u, y\land v) \equiv \mu a.C \langle t \parallel \mu [x\land u \parallel [α]x\land y \parallel [α]v \parallel V \lor C] \rangle

Indeed, in context α, both sides of the above equation reduce to:

\langle t \parallel \mu [x\land u \parallel [α]x\land y \parallel [α]v \parallel V \lor C] \rangle

which can unlock further reductions involving V \lor C.
6.2 IMALL\_p^p Models

The relevant notion of additive structure is distributive.

**Positive additive structure** In the linear setting we consider the following.

**Definition 23.** A linear distributive category is a symmetric monoidal category with finite coproducts in which the distributive law of tensor products over coproducts holds; that is, the canonical map $\coprod_{i \in I} X_i \otimes Y \to \coprod_{i \in I} (X_i \otimes Y)$ is an isomorphism, for all finite families of objects $\{X_i\}_{i \in I}$ and objects $Y$.

**Distributive presheaves** Incorporating coproducts into IMLL\_p models (Definition 4) requires a further analysis of presheaves crucial to which is the following definition (cf. Fiore, Di Cosmo and Balat [19, Section 3.1]).

**Definition 24.** For a cocartesian category $\mathcal{L}$, a presheaf $P : \mathcal{L}^{op} \to \mathbf{Set}$ is said to be distributive whenever it maps coproducts in $\mathcal{L}$ to products in $\mathbf{Set}$; that is, for all finite families of objects $\{X_i\}_{i \in I}$ of $\mathcal{L}$, the canonical map $P(\coprod_{i \in I} X_i) \to \prod_{i \in I} P(X_i)$ is an isomorphism.

We let $\mathcal{D}\mathcal{V}$ be the full subcategory of $\mathcal{P}\mathcal{V}$ consisting of all the distributive presheaves.

For a linear distributive category $\mathcal{L}$, the convolution monoidal structure on $\mathcal{P}\mathcal{L}$ restricts to $\mathcal{D}\mathcal{L}$ yielding a strong monoidal Yoneda embedding $\mathcal{Y} : \mathcal{L} \hookrightarrow \mathcal{D}\mathcal{L}$ that induces a $\mathcal{D}\mathcal{L}$-enrichment (with respect to the convolution monoidal structure) on $\mathcal{L}$. Furthermore, $\mathcal{D}\mathcal{L}$ is a full reflective linear-exponential ideal of $\mathcal{P}\mathcal{L}$ and thereby underlies a $\mathcal{D}\mathcal{L}$-category (with respect to the convolution monoidal structure) $\mathcal{D}\mathcal{L}$ for which the Yoneda embedding is a $\mathcal{D}\mathcal{L}$-functor.

We remark that a $\mathcal{D}\mathcal{L}$-category is a $\mathcal{P}\mathcal{L}$-category in which every hom-presheaf is distributive, while the $\mathcal{D}\mathcal{L}$-enriched notions of functor, natural transformation, cartesian structure, and adjunctions coincide with those for $\mathcal{P}\mathcal{L}$-enrichment. For linear distributive $\mathcal{L}$, the same holds for powers.

**Negative additive structure** A $\mathcal{D}\mathcal{L}$-category $\mathcal{C}$ is cartesian whenever every finite family of objects $\{A_i\}_{i \in I}$ can be represented by an object $\prod_{i \in I} A_i$, together with an isomorphism

$$(\prod_{i \in I} A_i) \otimes B \cong \prod_{i \in I} (A_i \otimes B)$$

natural for $X$ in $\mathcal{L}$ and $\mathcal{L}$-natural for $B$ in $\mathcal{C}$.

**Adjunction models** The above leads to the following notion of model.

**Definition 25.** An IMALL\_p^p model consists of a linear distributive category $\mathcal{L}$ and a cartesian $\mathcal{D}\mathcal{L}$-category $\mathcal{S}$ with $\mathcal{L}$-powers together with a $\mathcal{D}\mathcal{L}$-adjunction $\mathcal{S} \rightleftarrows \mathcal{L}$.

In view of the remarks above the relationship between the $\mathcal{D}\mathcal{V}$ and $\mathcal{P}\mathcal{V}$ enriched notions involved in IMLL\_p and IMALL\_p models, there is only one difference between the two notions; namely, that the latter come with cartesian structure in $\mathcal{S}$ and linear distributive structure in $\mathcal{L}$ crucially satisfying

$$\mathcal{S}(M, N) \cong \prod_{i \in I} \mathcal{S}(\pi_i(M), \pi_i(N))$$

for all finite families of objects $\{\pi_i\}_{i \in I}$ in $\mathcal{L}$ and objects $M, N$ in $\mathcal{S}$.

More precisely, we have the following.

**Proposition 26.** An IMALL\_p^p model $\mathcal{S} \rightleftarrows \mathcal{L}$ is an IMALL\_p^p model iff $\mathcal{L}$ is linear distributive, $\mathcal{S}$ is cartesian, and every hom-presheaf of $\mathcal{S}$ is distributive.

Note that the refinement from enriching over $\mathcal{P}\mathcal{L}$ to enriching over $\mathcal{D}\mathcal{L}$ guarantees that:

$$\mathcal{S}(M, N) \cong 1$$

$$\mathcal{S}(\bigoplus_{i \in I} P_i, M, N) \cong \mathcal{S}(\bigoplus_{i \in I} P_i, M, N) \times \mathcal{S}(\bigoplus_{i \in I} P_i, M, N)$$

We mention a few examples of models.

**Example 27.** Every bicartesian symmetric monoidal closed category $\mathcal{F}$, cartesian and powered $\mathcal{F}$-category $\mathcal{G}$, and $\mathcal{F}\mathcal{L}$-adjunction $\mathcal{G} \rightleftarrows \mathcal{F}$ provide a IMALL\_p^p model. Some examples include:

1. $\mathcal{G} = \mathcal{F}$ with the monoidal-closure adjunction $(-) \otimes S \dashv [S \to -]$ for $S$ in $\mathcal{F}$.
2. $\mathcal{G} = \mathcal{F}^op$ with the linear double-dualisation adjunction $[- \to R] \dashv [- \to R]$ for $R$ in $\mathcal{F}$, and
3. $\mathcal{G} = \mathcal{F}$ for a cartesian *-autonomous category $\mathcal{L}$, with the adjoint equivalence provided by the duality.

**IMALL\_p^p semantics** The interpretation of types is given by setting:

$$(A \& B)^* = A^* \times B^* \quad T^* = 1$$

$$(A \oplus B)^* = A^* + B^* \quad 0^* = 0$$

As for terms, the definitions are:

- $[\Gamma : \pi_i : S : A_i \& A_i := \Delta_i : \Delta^{-} \Delta_{\pi_i} := \Delta_{\pi_i}$ is obtained from $[\Gamma : \pi_i : S : A_i := \Delta_i : \Delta^{-} \Delta_{\pi_i} := \Delta_{\pi_i}$ pre-composed with $\pi_i : A_i \to A_i$.
- $[\Gamma : \mu_{\langle a, b \rangle} : \beta^e \rho \rho' \rho'' : A \& B] : \Gamma^{+} \to GA'$ is obtained from the transpose of the map $\Gamma^{+} \to \beta^e \rho \rho' \rho'' : A \& B$ given by the pairing of the transposes of maps arising from $[c : \Gamma^{+} \to A] : \Gamma^{+} \to GA'$ and $[c' : \Gamma^{+} \to A] : \Gamma^{+} \to GB'$;
- $[\Gamma : \mu_{\langle V \rangle T} : \ell_{\langle V \rangle T} : \Gamma^{+} \to G 1]$ arises from the transpose of the unique map $\Gamma^{+} \to 1$;
- $[\Gamma : \pi_i (V) : 
\boxed{\prod_{i \in I} \Delta_i} : \Gamma^{+} \to A_i \& A_i := \Delta_i \pi_i := \Delta_{\pi_i}$ is obtained from the composite $\Gamma^{+} \to \prod_{i \in I} (A_i \& A_i) := \prod_{i \in I} A_i$ pre-composed with $\pi_i : A_i \to A_i$.
- $[\Gamma : \tilde{p} : (x : A) \rho p \rho' \rho'' \rho'' : A \& B \Delta := \Delta \Delta^{-} \Delta_{\rho p} := \Delta_{\rho p}$ is the transpose of the composite $\Gamma^{+} \otimes (A + B) := \Gamma^{+} \otimes A + B$ where the second map is the copairing of $[c : \Gamma^{+} \times A \to \Delta]$ and $[c' : \Gamma^{+} \times B \to \Delta]$;
- $\Gamma^{+} \otimes 0 \Delta := \Gamma^{+} := 0 \Gamma^{+} \otimes 0 \Gamma^{+} := 0 \Gamma^{+} \otimes 0 \Gamma^{+} := 0 \Gamma^{+}$

**Theorem 28.** The IMALL\_p^p semantics is coherent and sound.

6.3 ILL\_p^p Calculus and Models

**Calculus** The calculus ILL\_p^p is given by adding exponentials and additives to IMLL\_p^p.

**Theorem 29.** ILL is expressible in ILL\_p^p.

**Theorem 30.** ILL\_p^p has the Barendregt-style properties.

**Adjunction models** As previously, our modular development allows for a straightforward extension with resources.

**Definition 31.** An IMALL\_p^p model with a resource modality $\mathcal{K} \rightleftarrows \mathcal{L}$ consists of an IMALL\_p^p model $\mathcal{S} \rightleftarrows \mathcal{L}$ together with a linear distributive category $\mathcal{K}$ and a monoidal adjunction $\mathcal{K} \rightleftarrows \mathcal{L}$. An ILL\_p^p model is an IMALL\_p^p model with resources in which the symmetric monoidal structure of $\mathcal{K}$ is cartesian.

The ILL\_p^p semantics is that of IMALL\_p^p and IMELL\_p^p, and thus also sound.

**Lifting theorem** For every symmetric strong monoidal functor $L : \mathcal{K} \rightarrow \mathcal{L}$ between linear distributive categories $\mathcal{K}$ and $\mathcal{L}$, the symmetric monoidal adjunction $L : \Gamma \dashv L^* : \mathcal{D}\mathcal{L} \rightarrow \mathcal{D}\mathcal{K}$. As in
Section 5.2, the \( \mathcal{D}\mathcal{X} \)-category \( L^* \mathcal{D} \) has \( \mathcal{K} \)-powers and, moreover, it is cartesian whenever \( \mathcal{K} \) is:
\[
L^* \mathcal{K}(N, \prod_{i \in I} N_i) \cong \prod_{i \in I} L^* \mathcal{K}(N, N_i).
\]
Hence we have the following lifting theorem.

**Theorem 32.** Every \( \text{IMALL}_p \) model with a resource modality \((\mathcal{D}, \Delta, \mathcal{K}) \) induces an \( \text{IMALL}_p \) model, obtained by composing the following two adjunctions:
\[
\mathcal{K} \rightleftarrows L^* \mathcal{D} \rightleftarrows L^* \mathcal{K}.
\]

### 6.4 \( \text{LJ}_p^* \), Calculus and Models

**Calculus** The calculus \( \text{LJ}_p^* \) is given by adding additives to \( \text{MLL}_p^* \). Thus, the abstract machine for exceptions can, as an alternative to \( \text{try-try} \)-unifiers, handle \( \tau(t, x, u, E(y), v) \) in scope, by solving the following (simplified) equations.
\[
\begin{align*}
(V \parallel \text{return})^* (\mu \mathcal{R}(R)(x,c(E(y),c'), \pi) &\rightarrow^* \epsilon[V/x][\pi] \\
(\text{raise } E(V) \parallel S)^* (\mu \mathcal{R}(R)(x,c(E(y),c'), \pi) &\rightarrow^* \epsilon[V/y][\pi] \\
\tau(t, x, u, E(y), v) &\rightarrow^* (t \parallel \text{return})^* (\mu \mathcal{R}(R)(a \parallel S)^* [E.(v \parallel S)^*], \pi)
\end{align*}
\]

### 7. Perspectives

**Benefits of L-calculi** L-calculi are an alternative to structured operational semantics. They can be seen as providing a principled reconstruction of defunctionalised CPS in direct style [40], while giving a modern view on the dualities of computation: expression/context, producer/consumer, strict/lazy.

In particular, as a legacy of CPS, there are benefits reminiscent of the advantages of CPS over ANF [20] and monadic meta-languages [35] as reported by Kennedy [28]: simplifications of commuting conversions, of inlining, of sharing of contexts... A natural question is, then, whether these simplifications carry over to the operational and equational modelling of effects and resources.

**Commuting conversions and effects** Benton and Kennedy [6] show that lessons from the proof theory of \( \lambda \)-calculus with sums, in particular the role of commuting conversions, help in the design and equational reasoning of exceptions. It is thus tempting to investigate how far advantages of L-calculi over the \( \lambda \)-calculus with sums carry over to the study of effects.

Benton and Kennedy refine the syntax for exceptions using a new form of handler \( \text{try-unless} \) which is suitable for expressing commuting conversions in the vein of the conversions for sums. Commuting conversions are used to turn an abstract-machine-based operational semantics of exceptions into a definition in terms of the source language, which is simpler since “there is a certain amount of clutter involved in using stacks (extra syntax, type rules, etc.)”. They moreover show that it is possible to produce optimised code by combining commuting conversions with effect analysis in the intermediate representation (for instance, compiling a non-tail-recursive source example into a tail-recursive final code). Finally, while commuting conversions duplicate expressions, they mention that it is possible to avoid the explosion in code size by using in the implementation “a special abstraction construct which compiles to a block of code accessed by jumps”.

The calculus \( \text{LJ}_p^* \) can be extended so as to give an operational semantics for exceptions, using an auxiliary stack of stacks, as introduced by Ariola, Herbelin and Sabry [2] in the context of delimited control operators.
\[
\begin{align*}
\gamma \mu \text{[}\parallel S\text{]}^{\gamma}[\pi] &\rightarrow^* \epsilon[S, \pi] \\
(V \parallel \text{return})^* (S, \pi) &\rightarrow^* (V \parallel S)^* [\pi]
\end{align*}
\]

To do so, those stacks are made to correspond to the successive \( \text{try-unless} \) handlers \( \tau(t, x, u, E(y), v) \) in scope, by solving the following (simplified) equations.
\[
\begin{align*}
(V \parallel \text{return})^* (\mu \mathcal{R}(R)(x,c(E(y),c'), \pi) &\rightarrow^* \epsilon[V/x][\pi] \\
(\text{raise } E(V) \parallel S)^* (\mu \mathcal{R}(R)(x,c(E(y),c'), \pi) &\rightarrow^* \epsilon[V/y][\pi] \\
\tau(t, x, u, E(y), v) &\rightarrow^* (t \parallel \text{return})^* (\mu \mathcal{R}(R)(a \parallel S)^* [E.(v \parallel S)^*], \pi)
\end{align*}
\]

In particular, it is possible to derive the commuting conversions that Benton and Kennedy use to define the operational semantics [6].

However, other conversions, characteristic of exceptions [3], are out of reach. Thus, on the one hand, the L-calculus provides a simpler semantics for exceptions, but on the other hand it is not able to produce by itself exception-specific optimisations as in Benton and Kennedy (as one could expect beforehand).

Lastly, the duplications happening during commuting conversions are mediated by \( \mu \). Thus, the mismatch between the reduction theory and the implementation in Benton and Kennedy can be explained as the need to determine an appropriate sharing implementation for a single \( \mu \) binder, as is done with CPS [28].

The challenge is to let L-calculi model the operational semantics of a wider class of effects, such as algebraic effects, along the above lines, all the while simplifying the study of their equational theory.

**Completeness** In connection to the model theory, direct algebraic descriptions of our calculi are to be obtained in terms of the duoids arising from the effect adjunctions. These direct models are to be put in reflection with the enriched adjunction models, to generalise Führmann’s direct characterisation of \( \mathcal{L}_\mu \)-models [21], in the continuity of [39]. In fact, we conjecture that the calculi form initial models, not only for direct duoid models, but also for the adjunction models, provided that the data of values and stacks is appropriately preserved.

In particular, we have observed that the distributivity requirement on presheaves, while needed for the correspondence with CBPV models, and holding in all concrete models we know of, was actually not used for defining the semantic interpretation. However, as syntactic presheaves are naturally distributive, we conjecture that distributivity is required for completeness.

**Biclosed action models** As alluded in Example 8, we are aware of classes of models that display more mathematical structure than the ones presented here and will be studied elsewhere. We are particularly interested in the class given by biclosed symmetric-monoidal skew actions, that seems to be related to delimited control and type-and-effect systems.
Acknowledgements

We thank the anonymous reviewers. G. Munch-Maccagnoni was supported by ERC ECSYM; M. Fiore partially so.

References


[47] ________, Call-by-value is dual to call-by-name, SIGPLAN Not. 38 (2003), no. 9, 189–201.
