A monadic approach to differentiation

Jad Koleilat
under the supervision of Marie Kerjean

September 2024

Université Paris Cité, Université Sorbonne Paris Nord, LIPN

Contents

1 Introduction

Linear logic was invented by Girard [9] after investigating models of λ-calculus. Differential linear logic was introduced by Ehrhard and Regnier [7] after investigating some models of linear logic. Codigging was introduced by Kerjean and Lemay in [10] after investigating models of differential linear logic. This illustrate a trend that is the motivation of this memoir: using semantics to extend syntax. This not an usual way of considering semantics hence, a point needs to be clarified.

Often, one of the requirement of a good definition of semantics is completeness. This is not absurd (quite the opposite) as completeness allows us to understand the syntax, often a mathematical theory, from properties of the semantics. Here, we are not interested in finding syntactic properties of our logical systems but, we are very interested in finding meaningful way to extend our syntax. When talking about models, we mean categories (as hinted by the abundance of diagrams) such that we have soundness ($\Gamma \vdash A \Rightarrow \Gamma \vDash A$) and not necessarily completeness, no family of models we describe here is in fact complete. We use semantics to find families of worlds in which our syntax lives then, by adding more syntax we try to capture only those in which we are interested.

The goal of the internship was to use semantics of codigging, particular the fact that it forms a monad, to find a type theory with a monad capturing the semantics of "quantitative differentiation" (as explained in section 4). Since this semantic is not simple, I chose to explain it step by step and with accuracy. Section 2 is a description of the semantic of (propositional) linear logic. Since everything that follows is based on it, I chose to give a lot more technical details that usually found in the literature. It's not an exhaustive description but covers most of what is needed for this memoir. Section 3 presents in detail the intuition and semantics of differential linear logic. The last section, section 4, is focused on codigging and a description of a possible path to complete our goal.

The results obtained in this internship are still preliminary: a big part of it was spent in studying and deeply understanding the litterature, as well as attending a research school and applying for a PhD grant (successfully obtained). However, we will continue this work in the forthcoming months. The memoir is a testimony of the science learned during the internship. I tried to make this memoir as accessible as possible and hope it will constitute a good introduction to the subject.

This memoir was made with the assumption that the reader has a basic understanding of categorical semantics, is familiar with linear logic and category theory (monads, adjunctions) as we only provide quick reminders. Some well known categorical results are presented in the appendix (see A).

Notations: When dealing with natural transformation we will often omit the indices, writing

$$\begin{array}{ccc} F(A) & \xrightarrow{Ff} & F(B) \\ \downarrow{\scriptstyle \nu} & & \downarrow{\scriptstyle \nu} \\ G(A) & \xrightarrow{Gf} & G(B) \end{array} \qquad \text{instead of} \qquad \begin{array}{ccc} F(A) & \xrightarrow{Ff} & F(B) \\ \downarrow{\scriptstyle \nu_A} & & \downarrow{\scriptstyle \nu_B} \\ G(A) & \xrightarrow{Gf} & G(B) \end{array}$$

When dealing with monoidal product, we write $f \otimes A$ for $f \otimes \mathrm{id}_A$. By default, the monoid operation is parenterised to the left, for example: $A \otimes B \otimes C$ is $(A \otimes B) \otimes C$. All implications ($\Rightarrow$ or $\multimap$) parenterise to the right, for example: $A \multimap B \multimap C$ is $A \multimap (B \multimap C)$.

2 Categorical semantics of Linear Logic

In this section we motivate and show that linear-non-linear adjuctions are a good basis for building models of Linear Logic. We also give a well know example: the relational model.

2.1 Reminders on LL

Linear Logic (LL for short), introduced in 1987 by Girard [9], is part of a family of logics called sub-structural logics. They are deduction systems where the use of structural rule (contraction, weakening, ...) is restricted in some way. In the case of LL, the use of structural rule is very restricted and monitored, this gives LL the ability to capture more nuances in reasoning than usual classical logic, distinguishing between "linear steps" and "non-linear steps".

A presentation of LL and some of its fragments is given in this subsection, for a more detailed explanation we recommend [12]. For now, we assume sequents to be constructed on *lists* of formulas and not multisets, we discus this choice in 2.2.4.

Definition 2.1.1 (LL). *Formulas of linear logic are defined as follows:*

$$A := \ X \mid A \parr A \mid A \,\&\, A \mid A \otimes A \mid A \oplus A \mid X^\perp \mid 0 \mid 1 \mid \perp \mid \top \mid {!A} \mid {?A}$$

For any formula A of Linear Logic we define $A^\perp$ as:

$$\begin{aligned}
(X^\perp)^\perp &:= X\\
(A \otimes B)^\perp &:= A^\perp \parr B^\perp\\
(A \parr B)^\perp &:= A^\perp \otimes B^\perp\\
0^\perp &:= \top\\
1^\perp &:= \perp\\
(A \,\&\, B)^\perp &:= A^\perp \oplus B^\perp\\
(A \oplus B)^\perp &:= A^\perp \,\&\, B^\perp\\
\top^\perp &:= 0\\
\perp^\perp &:= 1\\
(!A)^\perp &:= {?A^\perp}\\
(?A)^\perp &:= {!A^\perp}
\end{aligned}$$

linear logic is the following one sided sequent calculus:

$$\frac{}{\vdash A^\perp, A}\,\mathbf{ax} \qquad \frac{\vdash \Gamma, A \qquad \vdash \Delta, A^\perp}{\vdash \Gamma, \Delta}\,\mathbf{cut} \qquad \frac{\vdash \Gamma, A, B, \Delta}{\vdash \Gamma, B, A, \Delta}\,\mathbf{ex}$$

$$\frac{\vdash \Gamma, A \qquad \vdash \Delta, B}{\vdash \Gamma, \Delta, A \otimes B}\,\otimes \qquad \frac{\vdash \Gamma, A, B}{\vdash \Gamma, A \parr B}\,\parr$$

$$\frac{\vdash \Gamma, A \qquad \vdash \Gamma, B}{\vdash \Gamma, A \,\&\, B}\ \& \qquad \frac{\vdash \Gamma, A}{\vdash \Gamma, A \oplus B}\ \oplus_1 \qquad \frac{\vdash \Gamma, B}{\vdash \Gamma, A \oplus B}\ \oplus_2$$

$$\frac{}{\vdash 1}\ \mathbf{1} \qquad \frac{\vdash \Gamma}{\vdash \Gamma, \bot}\ \bot \qquad \frac{}{\vdash \Gamma, \top}\ \top$$

$$\frac{\vdash ?\Gamma, A}{\vdash ?\Gamma, !A}\ \mathbf{P} \qquad \frac{\vdash \Gamma, A}{\vdash \Gamma, ?A}\ \mathbf{d}$$

$$\frac{\vdash \Gamma}{\vdash \Gamma, ?A}\ \mathbf{w} \qquad \frac{\vdash \Gamma, ?A, ?A}{\vdash \Gamma, ?A}\ \mathbf{c}$$

ax *stands for "axiom",* **ex** *for "exchange",* **P** *for "promotion",* **d** *for "dereliction",* **w** *for "weakening" and* **c** *for "contraction".*

Definition 2.1.2 (ILL)**.** *Intuisionistic Linear Logic (ILL) is a fragment of linear logic obtained by restricting LL to the connectors* $\otimes$*,* $\multimap$*,* $!$ *and considering only intuitionist sequents (allowing only one formula on the right):*

$$\frac{}{A \vdash A}\ \mathbf{ax} \qquad \frac{\Gamma \vdash A \qquad \Delta_1, A, \Delta_2 \vdash B}{\Delta_1, \Gamma, \Delta_2 \vdash B}\ \mathbf{cut} \qquad \frac{\Delta_1, A_1, A_2, \Delta_2 \vdash B}{\Delta_1, A_2, A_1, \Delta_2 \vdash B}\ \mathbf{ex}$$

$$\frac{\Delta_1, A, B, \Delta_2 \vdash C}{\Delta_1, A \otimes B, \Delta_2 \vdash C}\ \otimes\mathbf{L} \qquad \frac{\Gamma \vdash A \qquad \Delta \vdash B}{\Gamma, \Delta \vdash A \otimes B}\ \otimes\mathbf{R}$$

$$\frac{\Gamma \vdash A \qquad \Delta_1, B, \Delta_2 \vdash C}{\Delta_1, \Gamma, A \multimap B, \Delta_2 \vdash C}\ \multimap\mathbf{L} \qquad \frac{A, \Gamma \vdash B}{\Gamma \vdash A \multimap B}\ \multimap\mathbf{R}$$

$$\frac{\Delta_1, \Delta_2 \vdash A}{\Delta_1, 1, \Delta_2 \vdash A}\ \mathbf{1L} \qquad \frac{}{\vdash 1}\ \mathbf{1R}$$

$$\frac{!\Gamma \vdash A}{!\Gamma \vdash !A}\ \mathbf{P} \qquad \frac{\Delta_1, A, \Delta_2 \vdash B}{\Delta_1, !A, \Delta_2 \vdash B}\ \mathbf{d}$$

$$\frac{\Delta_1, \Delta_2 \vdash B}{\Delta_1, !A, \Delta_2 \vdash B}\ \mathbf{w} \qquad \frac{\Delta_1, !A, !A, \Delta_2 \vdash B}{\Delta_1, !A, \Delta_2 \vdash B}\ \mathbf{c}$$

Definition 2.1.3 (ILL+&)**.** *Intuisionistic Linear Logic with finite products (ILL+&) is an expansion of ILL adding the connector & and the rules:*

$$\frac{\Delta_1, A, \Delta_2 \vdash C}{\Delta_1, A \,\&\, B, \Delta_2 \vdash C}\ \&_1\mathbf{L} \qquad \frac{\Delta_1, A, \Delta_2 \vdash C}{\Delta_1, B \,\&\, A, \Delta_2 \vdash C}\ \&_2\mathbf{L} \qquad \frac{\Gamma \vdash A \qquad \Gamma \vdash B}{\Gamma \vdash A \,\&\, B}\ \&\mathbf{R}$$

$$\frac{}{\Gamma \vdash \top}\ \top$$

LL and its fragments enjoy the cut elimination property. For a description of the cut elimination procedure and further discutions on it, we refer the reader to [14, section 3].

2.2 Linear-non-linear adjunctions as models of LL

2.2.1 Reminders on categorical semantics

Categorical semantics is a branch of logic that consists in associating a category to a particular system, in our case a sequent calculus. This association is done such that each proposition A is associated to an object of the category, traditionally denoted $[\![A]\!]$ and each proof π of a sequent $A \vdash B$ is associated to a morphism, denoted $[\![\pi]\!]$, from $[\![A]\!]$ to $[\![B]\!]$ the objects respectively associated to A and B. Moreover, this association operation $[\![\ ]\!]$ must satisfy an other property: if π and π' are two proofs that reduce to the same cut free proof then, $[\![\pi]\!] = [\![\pi']\!]$.

The last condition yields commutative diagrams that must be verified in the model category. The difficulty in this process is often to find the right category such that all those diagrams commutes.

2.2.2 What should be a model of LL?

We first discus what categorical models of ILL should be in hope of giving some intuition for what follows, for a more detailed discussion we recommend [14]. Considering linear logic without the exponential modalities, it is clear that we need our model to be:

- A closed symmetric monoidal category to interpret ILL
- A cartesian category to interpret ILL+&
- A $\star$-autonomous[1] category to interpret LL

Now, let's consider the exponentials. The connector ! is functorial:

$$\dfrac{\dfrac{A \vdash B}{!A \vdash B}\ \mathbf{d}}{!A \vdash !B}\ \mathbf{P}$$

Hence, we should look to interpret it as an endofunctor. ! has more structure than simply an endofunctor, it's a comonad with counit given by:

$$\dfrac{A \vdash A}{!A \vdash A}\ \mathbf{d}$$

and comultiplication:

$$\dfrac{\dfrac{\dfrac{A \vdash A}{!A \vdash A}\ \mathbf{d}}{!A \vdash !A}\ \mathbf{P}}{!A \vdash !!A}\ \mathbf{P}$$

Moreover, ! should be a lax monoidal functor since we can deduce in ILL the sequent $!A \otimes !B \vdash !(A \otimes B)$. It should also be strongly monoidal from the cartesian structure to the

[1] see 2.2.4

monoidal structure since we have the isomorphisms $!(A \& B) \simeq !A \otimes !B$. Finally, looking at the last two rules remaining, contraction and weakening, see that ! has more structure than simply a comonad. With this in mind, it's reasonable to search for a model among different kinds of adjunctions. A particular kind of adjunctions are linear-non-linear adjunction and, as we will see, they are a basis to build models of LL.

Definition 2.2.1. *A linear-non-linear adjunction is a symmetric monoidal adjunction between lax symmetric monoidal functors (see A.4):*

$$(\mathbb{M}, \times, e) \underset{(M,n)}{\overset{(L,m)}{\rightleftarrows}} (\mathbb{L}, \otimes, 1) \qquad \perp$$

Were $\times$ is a product and e a terminal object.

2.2.3 Semantics of ILL

Before tackling the casse of the full LL, we start by presenting how linear-non-linear adjunctions are a basis for models of ILL. From this simpler case we build in 2.2.5 models of ILLFP and then models of LL 2.2.6.

Proposition 2.2.2. *Every linear-non-linear adjunction between* $(\mathbb{L}, \otimes, 1)$ *a symmetric monoidal closed category and a cartesian category* $(\mathbb{M}, \times, e)$ *give rise to a model of ILL.*

Proof. Let us consider a linear-non-linear adjunction.

$$(\mathbb{M}, \times, e) \underset{(M,n)}{\overset{(L,m)}{\rightleftarrows}} (\mathbb{L}, \otimes, 1) \qquad \perp$$

were $(\mathbb{L}, \otimes, 1)$ is symmetric monoidal closed, $A \multimap -$ is the right adjoint of $A \otimes -$, $\eta : \mathbb{1}_{\mathbb{M}} \Rightarrow M \circ L$ the unit and $\varepsilon : L \circ M \Rightarrow \mathbb{1}_{\mathbb{L}}$ the counit. $\mathbb{L}$ will be our denotational category. First we need to choose a function ρ from the set of propositional variable to the objects of $\mathbb{L}$, we call it an environment. Once ρ is chosen, we now define the interpretation function $[\![\ ]\!]$. Firstly for terms:

- For a variable X, $[\![X]\!] := \rho(X)$
- $[\![1]\!] := 1$
- For a term of the form $A \otimes B$, $[\![A \otimes B]\!] := [\![A]\!] \otimes [\![B]\!]$
- For a term of the form $A \multimap B$, $[\![A \multimap B]\!] := [\![A]\!] \multimap [\![B]\!]$

- For a term of the form $!A$, $[\![!A]\!] \coloneqq L \circ M\,[\![A]\!]$. To simplify notations, let us write $!$ for $L \circ M$. Hence, with this notation $[\![!A]\!] \coloneqq !\,[\![A]\!]$

Every adjunction gives rise to a comonad, in our case $(!, \delta, \varepsilon)$ is a comonad with ε the counit of the adjunction and $\delta \coloneqq L\eta_M$.

Now for the interpretation of proofs, by induction on the rules of derivations:

ax The axiom rule is interpreted as the identity morphism

cut Let $f : [\![\Gamma]\!] \to [\![A]\!]$ and $g : [\![\Delta_1]\!] \otimes [\![A]\!] \otimes [\![\Delta_2]\!] \to [\![B]\!]$ be the interpretation of 2 proofs then, the interpretation of the proof obtained by applying the cut rule is the morphism: $g \circ ([\![\Delta_1]\!] \otimes f \otimes [\![\Delta_2]\!]) : [\![\Delta_1]\!] \otimes [\![\Gamma]\!] \otimes [\![\Delta_2]\!] \to [\![B]\!]$

ex Let $f : [\![\Delta_1]\!] \otimes [\![A_1]\!] \otimes [\![A_2]\!] \otimes [\![\Delta_2]\!] \to [\![B]\!]$ be the interpretation of a proof then, the interpretation of the proof obtained by applying the exchange rule is the morphism: $f \circ ([\![\Delta_1]\!] \otimes \gamma_{[\![A_2]\!],[\![A_1]\!]} \otimes [\![\Delta_2]\!]) : [\![\Delta_1]\!] \otimes [\![A_2]\!] \otimes [\![A_1]\!] \otimes [\![\Delta_2]\!] \to [\![B]\!]$

⊗L The left $\otimes$ rule is interpreted as composing with the identity morphism

⊗R Let $f : [\![\Gamma]\!] \to [\![A]\!]$ and $g : [\![\Delta]\!] \to [\![B]\!]$ be the interpretation of 2 proofs then, the interpretation of the proof obtained by applying the right $\otimes$ rule is the morphism: $f \otimes g : [\![\Gamma]\!] \otimes [\![\Delta]\!] \to [\![A]\!] \otimes [\![B]\!]$

⊸ L Let $f : [\![\Gamma]\!] \to [\![A]\!]$ and $g : [\![\Delta_1]\!] \otimes [\![B]\!] \otimes [\![\Delta_2]\!] \to [\![C]\!]$ be the interpretation of 2 proofs then, the interpretation of the proof obtained by applying the left $\multimap$ rule is the morphism:

$$\begin{array}{c}
C \\
\big\uparrow{\scriptstyle g} \\
[\![\Delta_1]\!] \otimes [\![\Gamma]\!] \otimes [\![B]\!] \otimes [\![\Delta_2]\!] \\
\big\uparrow{\scriptstyle [\![\Delta_1]\!] \otimes\ \mathrm{eval}_{[\![A]\!],[\![B]\!]}\ \otimes [\![\Delta_2]\!]} \\
[\![\Delta_1]\!] \otimes [\![A]\!] \otimes [\![A]\!] \multimap [\![B]\!] \otimes [\![\Delta_2]\!] \\
\big\uparrow{\scriptstyle [\![\Delta_1]\!] \otimes f \otimes [\![A]\!] \multimap [\![B]\!] \otimes [\![\Delta_2]\!]} \\
[\![\Delta_1]\!] \otimes [\![\Gamma]\!] \otimes [\![A]\!] \multimap [\![B]\!] \otimes [\![\Delta_2]\!]
\end{array}$$

⊸ R Let $f : [\![A]\!] \otimes [\![\Gamma]\!] \to [\![B]\!]$ be the interpretation of a proof then, the interpretation of the proof obtained by applying the right $\multimap$ rule is the left transpose of f, $\widehat{f} : [\![\Gamma]\!] \to [\![A]\!] \multimap [\![B]\!]$

1L Let $f : [\![\Delta_1]\!] \otimes [\![\Delta_2]\!] \to [\![A]\!]$ be the interpretation of a proof then, the interpretation of the proof obtained by applying the left 1 rule if the morphism: $f \circ (\rho \otimes [\![\Delta_2]\!]) : [\![\Delta_1]\!] \otimes 1 \otimes [\![\Delta_2]\!] \to [\![A]\!]$. Notice that we could also have taken $f \circ ([\![\Delta_1]\!] \otimes \lambda) \circ \alpha$ as the interpretation but, since the axioms of monoidal category garanties that this diagram commutes, this choice does not matter.

$$\begin{array}{ccc}
\llbracket\Delta_1\rrbracket \otimes \llbracket\Delta_2\rrbracket & \xrightarrow{\quad f \quad} & C \\
\uparrow{\scriptstyle \rho\otimes\llbracket\Delta_2\rrbracket} & & \uparrow{\scriptstyle f} \\
\llbracket\Delta_1\rrbracket \otimes 1 \otimes \llbracket\Delta_2\rrbracket & \xrightarrow{(\llbracket\Delta_1\rrbracket\otimes\lambda)\circ\alpha} & \llbracket\Delta_1\rrbracket \otimes \llbracket\Delta_2\rrbracket
\end{array}$$

1R The right 1 rule is interpreted as the identity morphism of 1

P Let $f : \llbracket !\Gamma \rrbracket \to \llbracket A \rrbracket$ be the interpretation of a proof and assume for simplicity $\Gamma = B_1, B_2$. Then, the interpretation of the proof obtained by applying the promotion rule is the morphism:

$$\begin{array}{ccccc}
!!\llbracket B_1\rrbracket \otimes !!\llbracket B_2\rrbracket & \xrightarrow{L(m)\circ n} & !(!\llbracket B_1\rrbracket \otimes !\llbracket B_2\rrbracket) & \xrightarrow{\quad !f \quad} & !\llbracket A\rrbracket \\
\uparrow{\scriptstyle \delta_{B_1}\otimes\delta_{B_2}} & & & & \\
\llbracket !\Gamma\rrbracket = !\llbracket B_1\rrbracket \otimes !\llbracket B_2\rrbracket & & & &
\end{array}$$

d Let $f : \llbracket\Delta_1\rrbracket \otimes \llbracket A\rrbracket \otimes \llbracket\Delta_2\rrbracket \to \llbracket B\rrbracket$ be the interpretation of a proof, then the interpretation of the proof obtained by applying the derelection rule is the morphism: $f \circ (\llbracket\Delta_1\rrbracket \otimes \varepsilon_A \otimes \llbracket\Delta_2\rrbracket) : \llbracket\Delta_1\rrbracket \otimes !\llbracket A\rrbracket \otimes \llbracket\Delta_2\rrbracket \to \llbracket B\rrbracket$

w Let $f : \llbracket\Delta_1\rrbracket \otimes \llbracket\Delta_2\rrbracket \to \llbracket B\rrbracket$ be the interpretation of a proof and A a formula of ILL. Let t be the unique morphism of $M(\llbracket A\rrbracket)$ to e. Then, the interpretation of the proof obtained by applying the weakening rule is the morphism:

$$\begin{array}{ccccc}
\llbracket\Delta_1\rrbracket \otimes 1 \otimes \llbracket\Delta_2\rrbracket & \longrightarrow & \llbracket\Delta_1\rrbracket \otimes \llbracket\Delta_2\rrbracket & \xrightarrow{\ f\ } & \llbracket B\rrbracket \\
\uparrow{\scriptstyle \llbracket\Delta_1\rrbracket\otimes\varepsilon\otimes\llbracket\Delta_2\rrbracket} & & & & \\
\llbracket\Delta_1\rrbracket \otimes !1 \otimes \llbracket\Delta_2\rrbracket & & & & \\
\uparrow{\scriptstyle \llbracket\Delta_1\rrbracket\otimes L(n^0\circ t)\otimes\llbracket\Delta_2\rrbracket} & & & & \\
\llbracket\Delta_1\rrbracket \otimes !\llbracket A\rrbracket \otimes \llbracket\Delta_2\rrbracket & & & &
\end{array}$$

c Let $f : \llbracket\Delta_1\rrbracket \otimes !\llbracket A\rrbracket \otimes !\llbracket A\rrbracket \otimes \llbracket\Delta_2\rrbracket \to \llbracket B\rrbracket$ be the interpretation of a proof and δ the diagonal morphism in $(M, \times)$. Applying theorem A.4.6 to the linear-non-linear adjunction we deduce that (L, m) is a strong monoidal functor. Hence, we can define the interpretation of the proof obtained by applying the contraction rule as the morphism:

$$
\begin{array}{ccc}
[\![\Delta_1]\!] \otimes {!}[\![A]\!] \otimes {!}[\![A]\!] \otimes [\![\Delta_2]\!] & \xrightarrow{\;f\;} & [\![B]\!] \\
\uparrow {\scriptstyle [\![\Delta_1]\!]\otimes m^{-1}\otimes[\![\Delta_2]\!]} & & \\
[\![\Delta_1]\!] \otimes L(M([\![A]\!]) \times M([\![A]\!])) \otimes [\![\Delta_2]\!] & & \\
\uparrow {\scriptstyle [\![\Delta_1]\!]\otimes(L\circ\delta)\otimes[\![\Delta_2]\!]} & & \\
[\![\Delta_1]\!] \otimes {!}[\![A]\!] \otimes [\![\Delta_2]\!] & &
\end{array}
$$

What is left is checking that our semantic is stable by cut-elimination. This verification is left to the brave reader. ☐

2.2.4 Digging

Notice that the promotion rule is interpreted as the composition of the mediating map of !, functoriality of ! and comultiplication . From this semantic point of view, it's natural to split the promotion rule into two rules: $!_{\mathbf{f}}$ expressing the functoriality of ! and it's lax monoidal structure, and the digging rule **p** expressing the comultiplication.

$$
\dfrac{\Gamma \vdash A}{!\Gamma \vdash !A}\ !_{\mathbf{f}} \qquad \dfrac{\Gamma \vdash !A}{\Gamma \vdash !!A}\ \mathbf{p}
$$

Notice that as deduction systems, ILL and ILL with $!_f$ and digging are equivalent, meaning that both systems can infer the same sequents. In fact, $!_f$ and digging are admissible rules of ILL:

$$
\dfrac{\dfrac{\Gamma \vdash A}{!\Gamma \vdash A}\ \mathbf{d}}{!\Gamma \vdash !A}\ \mathbf{P}
\qquad\qquad
\dfrac{\Gamma \vdash !A \qquad \dfrac{\dfrac{\dfrac{}{A \vdash A}\ \mathbf{ax}}{!A \vdash !A}\ !_f}{!A \vdash !!A}\ \mathbf{P}}{\Gamma \vdash !!A}\ \mathbf{cut}
$$

and promotion is an admissible rule in ILL with digging and $!_f$ (for simplicity, assume $\Gamma = B, C$):

$$
\dfrac{\dfrac{\dfrac{\dfrac{}{C \vdash C}\ \mathbf{ax}}{!C \vdash !C}\ !_f}{!C \vdash !!C}\ \mathbf{p} \qquad \dfrac{\dfrac{\dfrac{\dfrac{}{B \vdash B}\ \mathbf{ax}}{!B \vdash !B}\ !_f}{!B \vdash !!B}\ \mathbf{p} \qquad \dfrac{!B, !C \vdash A}{!!B, !!C \vdash !A}\ !_{\mathbf{f}}}{!B, !!C \vdash !A}\ \mathbf{cut}}{!B, !C \vdash !A}\ \mathbf{cut}
$$

The semantics of $!_{\mathbf{f}}$ and **p** is the following:

- $!_{\mathbf{f}}$ Let $f : [\![\Gamma]\!] \to [\![A]\!]$ and assume for simplicity $\Gamma = B_1, B_2$. Then, the $!_{\mathbf{f}}$ rule applied to it is interpreted as the morphism

$$
{!}[\![B_1]\!] \otimes {!}[\![B_2]\!] \xrightarrow{L(m)\circ n} {!}([\![B_1]\!] \otimes [\![B_2]\!]) \xrightarrow{!f} {!}[\![A]\!]
$$

p Let $f : [\![\Gamma]\!] \to [\![!A]\!]$ then, the digging rule applied to it is interpreted as the morphism $\delta \circ f$.

This observation is also valid in LL defining $!_{\mathbf{f}}$ in the same way (only one formula to the right) and **p** by replacing "B" with "Γ" (an arbitrary number of formulas).

Since this new syntax is quite close to the semantics, it's common to use the vocabulary of the syntax for the semantics. For exemple, it's not rare to present a comonad as: a category equipped with a triplet $(!, \mathbf{p}, \mathbf{d})$ were $!$ is an endo-functor, $\mathbf{p}_A$ a natural transformation from $!A$ to $!!A$ called a digging and $\mathbf{d}_A$ a natural transformation from $!A$ to A called a derelection.

From a categorical point of view, one might thing that considering sequents as lists and adding an exchange rule is more general then considering sequents as multisets: the first approach corresponding with symmetric monoidal categories and the second one, strict symmetric monoidal categories. But, it's not the case since every symmetric monoidal category is monoidaly equivalent to a strict symmetric monoidal category [13, ch.7]. We can always assume that our categories are strict symmetric monoidal. Hence, the choice of lists or multistets has no impact on the syntax and the semantics. Moreover, considering a strict symmetric monoidal category eliminates the need to explicit the unit, associative and symmetric isomorphisms, bringing even closer the syntax and semantics.

2.2.5 Semantics of ILL+&

Proposition 2.2.3. *Let $(\mathbb{L}, \otimes, 1)$ be a symmetric monoidal closed category also equipped with a cartesian structure $(\mathbb{L}, \&, \top)$. Every linear-non-linear adjunction between $(\mathbb{L}, \otimes, 1)$ and a cartesian category $(\mathbb{M}, \times, e)$ give rise to a model of ILL+&.*

Proof. Let us consider a linear-non-linear adjunction.

$$(\mathbb{M}, \times, e) \underset{(M,n)}{\overset{(L,m)}{\rightleftarrows}} (\mathbb{L}, \otimes, 1) \qquad (\bot)$$

were $(\mathbb{L}, \otimes, 1)$ is symmetric monoidal closed, $(\mathbb{L}, \&, \top)$ is a cartesian category, $A \multimap -$ is the right adjoint of $A \otimes -$, $\eta : \mathbb{1}_{\mathbb{M}} \Rightarrow M \circ L$ the unit and $\varepsilon : L \circ M \Rightarrow \mathbb{1}_{\mathbb{L}}$ the counit. $\mathbb{L}$ will be our denotational category. As in the previous case (2.2.3), we define $!$ as $L \circ M$ which is a comonad $(!, \varepsilon, \mu)$ and the interpretation function $[\![-]\!]$ we are defining is a strict extension of what has already been done for the ILL case. Hence, what is left is the case of the & connector. The interpretation of a formula of the shape $A \,\&\, B$ is defined as $[\![A]\!] \,\&\, [\![B]\!]$ and the rules $\&_{\mathbf{1}}\mathbf{L}$, $\&_{\mathbf{2}}\mathbf{L}$, $\&\mathbf{R}$ and $\top$ are interpreted in the usual way. There needs to be compatibility between the multiplicative structure and the additive structure for this structure to be a model of ILL+&. In our case, it arrise naturally from the linear-non-linear adjunction. First, notice that we have an adjunction:

$$(\mathbb{M}, \times, e) \underset{M}{\overset{L}{\rightleftarrows}} (\mathbb{L}, \&, \top) \qquad L \dashv M$$

Which lifts from proposition A.4.9 to a symmetric monoidal adjunction between colax symmetric monoidal functors:

$$(\mathbb{M}, \times, e) \underset{(M,l)}{\overset{(L,k)}{\rightleftarrows}} (\mathbb{L}, \&, \top) \qquad (L,k) \dashv (M,l)$$

Now, by the dual of theorem A.4.9, (M, l) is a strong symmetric monoidal morphism. This makes

$$! : (\mathbb{L}, \&, \top) \xrightarrow{(M,l^{-1})} (\mathbb{M}, \times, e) \xrightarrow{(L,m)} (\mathbb{L}, \otimes, 1)$$

A strong symmetric monoidal functor. Finaly, we indeed have the isomorphisms $!A \otimes !B \simeq !(A \ \& \ B)$ and $1 \simeq !\top$. This results show compatibility between the multiplicative and additive world in our model. Again, what is left is checking that this interpretation is stable by cut elimination. This is left to the brave reader. □

2.2.6 Semantics of LL

Before building models of LL, some general categorical results are needed.

Definition 2.2.4 ($\star$-autonomous category)**.** *A $\star$-autonomous category is a symmetric closed monoidal category with a distinguished object $\bot$ such that for all objects A, $\partial_{A,\bot} : A \to (A \multimap \bot) \multimap \bot$ is an isomorphism. In the context of $\star$-autonomous categories, we write ∂_A for $\partial_{A,\bot}$ (see A.2.8).*[2]

The $\star$-autonomy allows us to model classical reasoning. In particular, we can define new connective from $\otimes$ and $\&$ that will behave as expected.

Proposition 2.2.5. *Let $(\mathbb{L}, \otimes, 1)$ be a $\star$-autonomous category that is also equipped with a cartesian structure $(\mathbb{L}, \&, \top)$ then, it is also cocartesian. Let us write $A^\perp$ for $A \multimap \bot$ then, the coproduct $A \oplus B$ is defined as $(A^\perp \ \& \ B^\perp)^\perp$ and the initial object is $\top^\perp$ denoted 0.*

Proof. Let $\phi : \mathrm{Hom}(B \otimes A, C) \to \mathrm{Hom}(B, A \multimap C)$ and $\psi : \mathrm{Hom}(A \otimes B, C) \to \mathrm{Hom}(B, A \multimap C)$ be the isomorphisms generated by the adjunction (see A.2.6).

[2] This definition is not the most general one. It can be defined in the context of biclosed monoidal categories

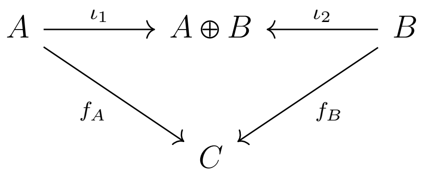
First, we show that for every objects A and B, there is a map from A to $A \oplus B$ and a map from B to $A \oplus B$. Notice that

$$\mathrm{Hom}(A^\perp \,\&\, B^\perp, A^\perp) \simeq \mathrm{Hom}((A^\perp \,\&\, B^\perp) \otimes A, \perp) \simeq \mathrm{Hom}(A, (A^\perp \,\&\, B^\perp)^\perp) = \mathrm{Hom}(A, A \oplus B)$$

Since the first projection is an element of $\mathrm{Hom}(A^\perp \,\&\, B^\perp, A^\perp)$, this transformation generates a morphism from A to $A \oplus B$ which we call ι_1. Explicitly,

$$\iota_1 := \ \psi(\phi^{-1}(\pi_1)) : A \to (A \oplus B)$$

Similarly, by taking the second projection we get a morphism ι_2 from B to $A \oplus B$. Now, we must check that $A \oplus B$ satisfies the coproduct universal property. Suppose given the following diagram:



Notice that we have the natural isomrophisms:

$$\begin{aligned}
\mathrm{Hom}(A \oplus B, C) &\simeq \mathrm{Hom}(A \oplus B, (C^\perp)^\perp) \\
&\simeq \mathrm{Hom}(C^\perp \otimes (A \oplus B), \perp) \\
&\simeq \mathrm{Hom}(C^\perp, (A \oplus B)^\perp) \\
&\simeq \mathrm{Hom}(C^\perp, A^\perp \,\&\, B^\perp) \\
&\simeq \mathrm{Hom}(C^\perp, A^\perp) \times \mathrm{Hom}(C^\perp, B^\perp) \\
&\simeq \mathrm{Hom}(A, C) \times \mathrm{Hom}(B, C)
\end{aligned}$$

Following this chain from bottom up, we get an explicit construction

$$\partial^{-1} \circ \psi(\phi^{-1}(\partial \circ \langle f_A^\perp, f_B^\perp \rangle)) : (A \oplus B) \to C$$

We leave to the reader the proof that this morphism commutes with the diagram and that it is unique.
We now show that 0 is an initial object. Once again, using isormphisms between Hom sets:

$$\mathrm{Hom}(A^\perp, \top) \simeq \mathrm{Hom}(\top^\perp, (A^\perp)^\perp) \simeq \mathrm{Hom}(0, A)$$

Since $\top$ is terminal, 0 must be initial. □

Proposition 2.2.6. *Let* $(\mathbb{L}, \otimes, 1)$ *be a* $*$*-autonomous category and let us define* $A \parr B$ *as* $(A^\perp \otimes B^\perp)^\perp$. *Then,* $\perp$ *is a neutral object for the* $\parr$ *operation and the functor* $A^\perp \parr -$ *is right adjoint to* $- \otimes A$.

Proof. First, we show that $1 \simeq \bot^\bot$. Using the fact that the yoneda embeding is full and faithfull:

$$\begin{aligned}\mathrm{Hom}(1, X) &\simeq \mathrm{Hom}(X^\bot, 1^\bot)\\ &\simeq \mathrm{Hom}(X^\bot \otimes 1, \bot)\\ &\simeq \mathrm{Hom}(X^\bot, \bot)\\ &\simeq \mathrm{Hom}(\bot^\bot, (X^\bot)^\bot)\\ &\simeq \mathrm{Hom}(\bot^\bot, X)\end{aligned}$$

Hence, $1 \simeq \bot^\bot$ and equivalently $\bot \simeq 1^\bot$. Let A be an object then:

$$\bot \parr A = (\bot^\bot \otimes A^\bot)^\bot \simeq (1 \otimes A^\bot)^\bot \simeq (A^\bot)^\bot \simeq A$$

Symmetrically, $A \parr \bot \simeq A$.
Let's show that $A^\bot \parr -$ is right adjoint to $- \otimes A$:

$$\begin{aligned}\mathrm{Hom}(B \otimes A, C) &\simeq \mathrm{Hom}(B \otimes A, (C^\bot)^\bot)\\ &\simeq \mathrm{Hom}(B \otimes A \otimes C^\bot, \bot)\\ &\simeq \mathrm{Hom}(B \otimes (A \otimes C^\bot), \bot)\\ &\simeq \mathrm{Hom}(B, (A \otimes C^\bot)^\bot)\\ &\simeq \mathrm{Hom}(B, ((A^\bot)^\bot \otimes C^\bot)^\bot)\\ &= \mathrm{Hom}(B, A^\bot \parr C)\end{aligned}$$

Since all isomorphisms are natural, $- \otimes A \dashv A^\bot \parr -$ and by uniqueness of adjoint $A^\bot \parr - \simeq A \multimap -$. □

Now that it has been established that everything works as expected in classical logic, it's very easy to see that the following is a model of LL. The only connector that was not tackled is ?, as expected it suffises to define it as $?- := (!-^\bot)^\bot$ which is an endofunctor.

Proposition 2.2.7. *Let* $(\mathbb{L}, \otimes, 1)$ *be a* $*$*-autonomous category also equipped with a cartesian structure* $(\mathbb{L}, \&, \top)$*. Every linear-non-linear adjunction between* $(\mathbb{L}, \otimes, 1)$ *and a cartesian category* $(\mathbb{M}, \times, e)$ *give rise to a model of LL.*

2.3 Seely categories as models of LL

What has been seen untill now is that linear-non-linear adjunctions give rise to models of LL. In this section we ask a dual question: when does category equipped with comonad define a linear-non-linear adjunction? In other words: what structure should we require on a category for it to be a model of LL? There are multiple answers to this question, we will only focus on one of them, Seely categories.

Definition 2.3.1 (Seely category)**.** *A Seely category is a symmetric closed monoidal category* $(\mathbb{L}, \otimes, 1)$ *that has finite products* $(\mathbb{L}, \&, \top)$ *together with:*

- *a comonad* $(!, \mathbf{p}, \mathbf{d})$.

- *two natural isomorphisms* $m^2_{A,B} : !A \otimes !B \to !(A \,\&\, B)$ *and* $m^0 : 1 \to !\top$ *which induce as strong symmetric monoidal structure* m *on* $!$.

- *the commutation, for all* A *and* B*, of the following coherence diagram:*

$$\begin{array}{ccc}
!A \otimes !B & \xrightarrow{\;m\;} & !(A \,\&\, B) \\
 & & \downarrow{\scriptstyle \mathbf{p}} \\
{\scriptstyle \mathbf{p}_A \otimes \mathbf{p}_B}\downarrow & & !!(A \,\&\, B) \\
 & & \downarrow{\scriptstyle !\langle !\pi_1, !\pi_2\rangle} \\
!!A \otimes !!B & \xrightarrow[\;m\;]{} & !(!A \,\&\, !B)
\end{array}$$

Theorem 2.3.2. *Every Seely category* $\mathbb{L}$ *is involved in a linear-non-linear adjunction with its coKleisli category* $\mathbb{L}_!$ *(see* **??***), making it a model of ILL+&. Hence, every* $\star$*-autonomous Seely category is a model of LL.*

To prove this theorem we must show two things:

- The coKleisli category can be equipped with a cartesian structure
- The adjunction lifts to a symmetric lax monoidal adjunction

Proof. Let $\mathbb{L}$ be a Seely category with the notation given in 2.3.1 and $\mathbb{L}_!$ the coKleisli category associated to the comonad $(!, \mathbf{p}, \mathbf{d})$. We have the following adjunction:

$$\mathbb{L}_! \;\underset{M}{\overset{L}{\rightleftarrows}}\; \mathbb{L} \qquad (\perp)$$

First, let's show that $\mathbb{L}_!$ is a cartesian category. Notice that the functor M is surjective on objects and since it's a right adjoint, it preserves limits. Hence, we can define $A \times B$ as $M(A \,\&\, B)$ and e as $M(\top)$. $(\mathbb{L}_!, \times, e)$ is a cartesian category.
Secondly, we must show that the following adjunction

$$(\mathbb{L}_!, \times, e) \;\underset{M}{\overset{L}{\rightleftarrows}}\; (\mathbb{L}, \otimes, 1) \qquad (\perp)$$

lifts to a symmetric monoidal adjunction between lax monoidal functors. Equivalently (see A.4.9) have to check that (L, m) is a strong symmetric monoidal functor. The axioms of Seely categories are there specially to ensure that this is the case. The rest of the proof is straightforward and can be found in [14]. □

2.4 The relational model, an example of models of LL

We construct an important denotational model of LL: the relational model. This model is somewhat "degenerate" as it does not distinguish the interpretation of a formula from the interpretation of its negation. Nevertheless, it's an important model and enjoy the property of being quite simple to define. First, we need to define the category **Rel**.

Definition 2.4.1. **Rel** *is the category of sets are relations, objects are sets and morphisms are binary relations, in other words:* $\mathbf{Rel}(X,Y) = \mathcal{P}(X \times Y)$*. If* $R \in \mathbf{Rel}(X,Y)$ *and* $S \in \mathbf{Rel}(Y,Z)$ *then composition is given by:* $S \circ R := \{(x,z) \in X \times Z \mid \exists y \in Y,\ x\,R\,y \wedge y\,S\,z\}$*. Identity is given by the diagonal relation.*

To show that it's a model of LL we will prove that it's a $\star$-autonomous Seely category. Firstly, $(\mathbf{Rel}, \otimes, 1)$ is a closed symmetric monoidal category were $X \otimes Y := X \times$ [3] Y and 1 is a singleton set. The closed structure is defined as $A \multimap B := A \otimes B$, the natural bijection $\mathbf{Rel}(X \otimes Y, Z) \simeq \mathbf{Rel}(X, Y \otimes Z)$ is simply the associativity of the cartesian product in **Set**. It's a $\star$-autonomous category for $\bot := \{\star\}$ the singleton. It has finite products given by $A \,\&\, B := A \sqcup$ [4] B and a terminal object $\top := \varnothing$. The projection maps: $\pi_i : X_1 \,\&\, X_2 \to X_i$ are given by $\pi_i := \{((i,x),x) \in (X_1 \,\&\, X_2) \times X_1 \mid x \in X_i\}$. Let $R_1 : Y \to X_1$ and $R_2 : Y \to X_2$ then, the paring $\langle R_1, R_2 \rangle$ is given by $\{(y,(i,x)) \in Y \times (X_1 \,\&\, X_2) \mid y\, R_i\, x \text{ for } i = 1,2\}$. **Rel** has a comonad structure $(!, \mathbf{p}, \mathbf{d})$ defined as:

- $!X$ is the set of all finite multisets of X. We denote multisets with square brackets instead of brackets and write $+$ for the union of multisets. Let $R : X \to Y$ then $!R := \{([x_1, \dots, x_n], [y_1, \dots, y_n]) \mid n \in \mathbb{N} \text{ and } x_i\, R\, y_i \text{ for } i = 1, \dots, n\}$.
- the digging is defined as $\mathbf{p}_X := \{(m_1 + \dots + m_n, [m_1, \dots, m_n]) \in\, !X \times !!X \mid n \in \mathbb{N} \text{ and } m_1, \dots, m_n \in\, !X\}$.
- the derelection is defined as $\mathbf{d}_X := \{([x], x) \mid x \in X\}$

For the next part we will need the following lemma

Lemma 2.4.2. *Isomorphisms in* **Rel** *are exactly the relations that are bijections.*

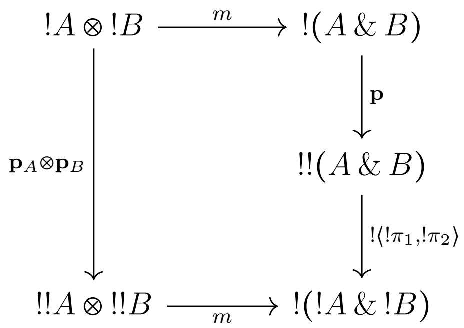
The transitions maps m^2 and m^0 making $!$ a strong monoidal functor from $(\mathbf{Rel}, \&, \top)$ to $(\mathbf{Rel}, \otimes, 1)$ are defined as follows:

[3] the cartesian product in **Set**

[4] The disjoint union in **Set**

- $m^2_{A,B} : !A \otimes !B \to !(A \,\&\, B)$ is defined as the bijection $([a_1, \dots, a_n], [b_1, \dots, b_m]) \mapsto [(1, a_1), \dots, (1, a_n), (2, b_1), \dots, (2, b_m)]$.
- $m^0 : 1 \to !\top$ is defined as the obvious bijection from $1 = \{\star\}$ to $!\top = \{[]\}$.

By computing each path, we see that the following diagram commutes:



This makes **Rel** a $\star$-autonomous Seely category, thus a model of LL. We can look in more details at the semantics of his model for example, the formula are interpreted as follows:

$$
\begin{aligned}
&[\![A \otimes B]\!] = [\![A \parr B]\!] = [\![A]\!] \times [\![B]\!] \\
&[\![A \,\&\, B]\!] = [\![A \oplus B]\!] = [\![A]\!] \amalg [\![B]\!] \\
&[\![1]\!] = [\![\bot]\!] = \{\star\} \\
&[\![0]\!] = [\![\top]\!] = \varnothing \\
&[\![!A]\!] = [\![?A]\!] = \mathcal{M}_{\text{fin}}(A)
\end{aligned}
$$

Where $\mathcal{M}_{\text{fin}}(A)$ is the set of finite multisets of A. The confusion in this model between a formula and its negation is due to the $\star$-autonomous structure: the dual operation $-^\perp$ is the identity. Looking at the semantics of a few rules, see that the weakening rule and contraction rule are respectively interpreted as natural transformations: $\mathbf{w}_X = \{([], \star)\} : !X \to 1$ and $\mathbf{c}_X = \{(m, (m_1, m_2)) \mid m = m_1 + m_2\} : !X \to !X \otimes !X$.

3 Differential Linear Logic

In this section we describe differential linear logic (DiLL), the intuition relating it to differentiation and differential categories as a semantics for it.

3.1 The calculus and intuitions

Let's consider for now only the models of LL that are Seely categories. Every model involves two categories, the denotational (or model) category $\mathbb{L}$ and it's coKleisli category $\mathbb{L}_!$. $\mathbb{L}$ can be thought as a "linear world" and the coKleisli category, corresponds to a "non linear world". Every morphism $A \to B$ in $\mathbb{L}_!$ is a morphism from $!A \to B$ in $\mathbb{L}$ and, according to the ressource interpretation of ! or the translation of classical logic to linear logic, we can think of morphism of this form as "non linear".

Differentiation is a process to transform a differentiable function into a linear function. Differential linear logic, is an extension of LL adding rules that capture in the syntax the notion of differentiation at the point 0 (in the semantics). It was first discovered in 2004 by Ehrhard and Regnier, see [7].

Definition 3.1.1 (DiLL). *Formulas of DiLL are the same as the ones of LL:*

$$A \coloneqq\; X \mid A \parr A \mid A \,\&\, A \mid A \otimes A \mid A \oplus A \mid X^\perp \mid 0 \mid 1 \mid \bot \mid \top \mid\, !A \mid\, ?A$$

As with linear logic, we can choose to present the calculus as a one sided or two sided calculus. Here, we make the choice of presenting it as a one side calculus with multisets of formulas. The $A^\perp$ operation is defined in the same way as LL. DiLL is the following calculus:

$$\frac{}{\vdash A^\perp, A}\,\mathbf{ax} \qquad \frac{\vdash \Gamma, A \qquad \vdash \Delta, A^\perp}{\vdash \Gamma, \Delta}\,\mathbf{cut} \qquad \frac{\vdash \Gamma, A, B, \Delta}{\vdash \Gamma, B, A, \Delta}\,\mathbf{ex}$$

$$\frac{\vdash \Gamma, A \qquad \vdash \Delta, B}{\vdash \Gamma, \Delta, A \otimes B}\,\otimes \qquad \frac{\vdash \Gamma, A, B}{\vdash \Gamma, A \parr B}\,\parr$$

$$\frac{\vdash \Gamma, A \qquad \vdash \Gamma, B}{\vdash \Gamma, A \,\&\, B}\,\& \qquad \frac{\vdash \Gamma, A}{\vdash \Gamma, A \oplus B}\,\oplus_{\mathbf{1}} \qquad \frac{\vdash \Gamma, B}{\vdash \Gamma, A \oplus B}\,\oplus_{\mathbf{2}}$$

$$\frac{}{\vdash 1}\,\mathbf{1} \qquad \frac{\vdash \Gamma}{\vdash \Gamma, \bot}\,\bot \qquad \frac{}{\vdash \Gamma, \top}\,\top$$

$$\boxed{\frac{\vdash ?\Gamma, A}{\vdash ?\Gamma, !A}\,\mathbf{P}} \qquad \frac{\vdash \Gamma, A}{\vdash \Gamma, ?A}\,\mathbf{d} \qquad \frac{\vdash \Gamma, A}{\vdash \Gamma, !A}\,\overline{\mathbf{d}}$$

$$\frac{\vdash \Gamma}{\vdash \Gamma, ?A}\,\mathbf{w} \qquad \frac{}{\vdash !A}\,\overline{\mathbf{w}} \qquad \frac{\vdash \Gamma, ?A, ?A}{\vdash \Gamma, ?A}\,\mathbf{c} \qquad \frac{\vdash \Gamma, !A \qquad \vdash \Delta, !A}{\vdash \Gamma, \Delta, !A}\,\overline{\mathbf{c}}$$

The rules $\overline{\mathbf{d}}$, $\overline{\mathbf{w}}$ *and* $\overline{\mathbf{c}}$ *are respectively the coderelection, the coweakeking and the cocontraction rules. The promotion rule was not included in the first paper defining DiLL [7] as it did not (yet) have a dual rule. For our purpose, we consider that the promotion rule (or digging +* $!_{\mathbf{f}}$*, as per the discussion in 2.2.4) is included in DiLL.*

Differential linear logic is obtained by symmetrising the derelection, weakening and contraction rules of LL into the coderelection, coweakeking and cocontraction rules.[5] In order to give intuitive meaning to those rules and the link with differentiation, we must first place ourself into a context of functions spaces. In DiLL, the duality linear/non linear of LL is replaced by the duality linear maps/smooth maps.

Some models of LL have vector spaces as objects and smooth functions as morphisms. In those models, $[\![X]\!]$ is a vector space and $[\![X \multimap Y]\!]$ is the linear maps $\mathcal{L}(X,Y)$. $[\![\bot]\!]$ is $\mathbb{R}$ therefore, $[\![X^\perp]\!] := [\![X \multimap \bot]\!]$ is interpreted as the dual space $\mathcal{L}(X,\mathbb{R})$. $[\![X \Rightarrow Y]\!]$[6] is the space of smooth maps $\mathcal{C}^\infty([\![X]\!],[\![Y]\!])$. For the interpretation of !, notice that $!X \simeq (!X \to \bot) \to \bot = (X \Rightarrow \bot) \to \bot$, therefore, $[\![!X]\!] = \mathcal{C}^\infty([\![X]\!],\mathbb{R})'$ the space of distribution with compact support on X. $?A \simeq (!A^\perp)^\perp$ hence, $[\![?A]\!] = \mathcal{C}^\infty(A^\perp,\mathbb{R})$. It's in those spaces that the rules of DiLL get their real significance:

- The derelection rule (presented in a monolateral and bilateral style)

  $$\frac{l : A \vdash B}{l : !A \vdash B}\,\mathbf{d} \qquad \frac{\vdash \Gamma, v : A}{\vdash \Gamma, (a' \mapsto a'(v)) : ?A}\,\mathbf{d}$$

  transforms a linear map $l : A \multimap B$ into a non linear one $A \Rightarrow B$ by forgetting it's a linear map.

- The coderelection rule

  $$\frac{\vdash \Gamma, v : A}{\vdash \Gamma, f \mapsto D_0(f)(v) : !A}\,\overline{\mathbf{d}} \qquad \frac{f : (!A \vdash \Gamma)}{(v \mapsto D_0(f)(v)) : (A \vdash \Gamma)}\,\overline{\mathbf{d}}$$

  corresponds to the "differential" operation. This is seen in the cut elimination procedure were

  $$\dfrac{\dfrac{\vdash \Gamma, v : A}{\vdash \Gamma, f \mapsto D_0(f)(v) : !A}\,\overline{\mathbf{d}} \qquad \dfrac{l : A \vdash B}{l : !A \vdash B}\,\mathbf{d}}{\vdash \Gamma, D_0(l)(v) : B}\,\mathbf{cut} \quad \rightsquigarrow \quad \dfrac{\vdash \Gamma, v : A \qquad l : A \vdash B}{\vdash \Gamma, l(v) = D_0(l)(v) : B}\,\mathbf{cut}$$

  Composing derelection with coderelection lead to identity since the derivative of a linear map is itself.

[5] One might wonder if it's possible to go one step further by symmetrising the promotion or digging rule and still obtain a meaningful system, this will be answered positively in section 4.

[6] $X \Rightarrow Y := !X \multimap Y$

Remark 3.1.2. Be wary that we giving a backwards explanation: giving the intuitions then showing that they behave as expected relatively to cut-elimination. It's important to keep in mind that the cut-elimination procedure is what gives us information on the semantics of each rule. It allows us to build intuition for those rules, not the other way arround. If the coderelection is interpreted as a diferentiation, it's because of the way it behave in the cut-elimation procedure. We will not detail the full procedure here, it can be found in [7] for the fragment of DiLL without promotion and in [16] for DiLL with promotion. The cut-elimination procedure will be illustrated in subsection 3.2 when specifying the required diagram commutations.

- The weakening rule

$$\dfrac{\vdash \gamma : \Gamma}{a \mapsto \gamma : (!A \vdash \Gamma)}\ \mathbf{w} \qquad \dfrac{\vdash \Gamma}{\vdash \Gamma, (a' \mapsto 1) : ?A}\ \mathbf{w}$$

  builds a constant (hence, non linear) function.

- The coweakeking rule

$$\dfrac{}{\vdash \delta_0 : !A}\ \overline{\mathbf{w}}^{7} \qquad \dfrac{f : (!A \vdash B)}{\vdash f(0) : B}\ \overline{\mathbf{w}}$$

  is the evaluation function at 0. This is illustrated by the cut-elimation procedure

$$\dfrac{\dfrac{\vdash \gamma : \Gamma}{\vdash \Gamma, (a' \mapsto 1) : ?A}\ \mathbf{w} \qquad \dfrac{}{\vdash \delta_0 : !A}\ \overline{\mathbf{w}}}{\vdash \Gamma, 1 : \bot (= \mathbb{R})}\ \mathbf{cut} \quad \rightsquigarrow \quad \vdash \Gamma$$

- The contraction rule

$$\dfrac{\vdash \Gamma, f : ?A, g : ?A}{\vdash \Gamma, f \cdot g : ?A}\ \mathbf{c} \qquad \dfrac{f : (!A, !A \vdash \Gamma)}{(x \mapsto f(x, x)) : (!A \vdash \Gamma)}\ \mathbf{c}$$

  multiplies two scalar functions.

- The cocontraction rule

$$\dfrac{\vdash \Gamma, \phi : !A \qquad \vdash \Delta, \psi : !A}{\vdash \Gamma, \Delta, \phi * \psi : !A}\ \overline{\mathbf{c}} \qquad \dfrac{f : !A \vdash B}{(\phi, \psi \mapsto f(\phi * \psi)) : !A, !A \vdash B}\ \overline{\mathbf{c}}$$

  is the convolution of distribution.

- The digging

$$\dfrac{f : !!A \vdash \Gamma}{(\psi \mapsto f(\delta_\psi)) : !A \vdash \Gamma}\ \mathbf{p}$$

  maps a distribution ψ to its Diract δ_ψ.

[7] The Diract function in 0 : $\delta_0(f) = f(0)$

3.2 Differential categories, a semantic for DiLL

Differential categories were first invented by Blute, Cockett and Seely in [2] to provide a semantics for DiLL but, quickly became a subject of interest on its own. Here, we present and use some of the formalism of differential category to provide a semantics for DiLL. We use the notations and terminology described by Ehrhard in [5], as they are better suited for this goal. This subsections takes also inspiration from [10] and [1].

First, models of DiLL should also be models of LL hence, we place our self in a similar setting: symmetric monoidal categories with finite product. As per the discussion in 2.2.4, we will assume our categories to be strict symmetric monoidal. As explained in 3.1, the codereliction rule is interpreted as the differential at point 0. Categorically, to be able to talk about "0" we need additional structure: pre-additive categories[8][9].

Definition 3.2.1 (CMon-enriched category)**.** *A CMon-enriched category is a category such that each* Hom*-set is a commutative monoid and the composition operation*

$$\circ : \mathrm{Hom}(A,B) \times \mathrm{Hom}(B,C) \to \mathrm{Hom}(A,C)$$

is bilinear.

Definition 3.2.2 (Pre-additive category)**.** *A pre-additive category is a CMon-enriched category that is also symmetric monoidal with a compatibility condition between the two structures:*

$$k \otimes (f+g) \otimes h = (k \otimes f \otimes h) + (k \otimes g \otimes h) \qquad k \otimes 0 \otimes h = 0$$

To interpret DiLL one needs finite product, this leads us to additive categories.

Definition 3.2.3 (Additive category)**.** *An additive category $\mathcal{C}$ is as pre-additive category that has all finite biproducts:*

- *$\mathcal{C}$ has all finite products and coproducts.*
- *The products and coproducts coincide: all products are coproducts and all coproducts are products. We denote by the same symbol $\oplus$, the product and coproduct.*
- *Let A, B be two objects, π_1, π_2 be the two projections of the product $A \oplus B$, ι_1, ι_2 the two injections of the coproduct $A \oplus B$ then,*

$$\pi_1 \iota_1 = \mathrm{id}$$
$$\pi_2 \iota_2 = \mathrm{id}$$
$$\iota_1 \pi_1 \iota_2 \pi_2 = \iota_2 \pi_2 \iota_1 \pi_1$$

[8] In [10] and [1], pre-additive categories are called additive categories.

[9] pre-additive categories usually refer to abelian-enriched categories (see pre-additive category), in the context of DiLL and differential categories it refers to CMon-enriched symmetric monoidal ategories (see CMon-enriched categories)

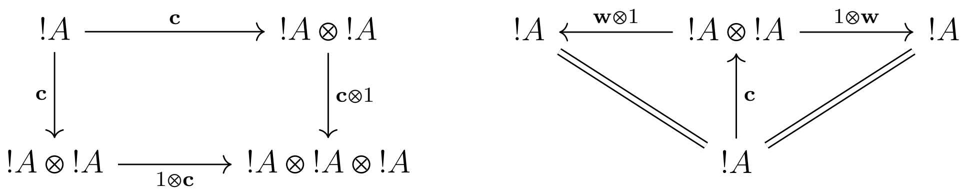
Theorem 3.2.4. *A pre-additive category that has all finite (co)products, has all finite biproducts hence, it is an additive category.*

Proof. The proof of this theorem is well know and can be found here. □

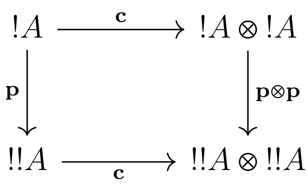
The consequence of this last theorem is that, in our context, models of DiLL necessarily equates $A\&B$ and $A\oplus B$. This is expected since we are trying to capture a semantic related to category of vector spaces (see 3.1) which are additive hence, $[\![A \& B]\!] = [\![A \oplus B]\!]$[10].

From now on, we write $\times$ for the biproduct and $\top$ for the zero object (a terminal object which is also an initial object) as we are more interested in its "product nature". We now give definitions that will lead to a categorical semantics for DiLL.

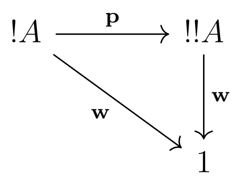
Definition 3.2.5 (coalgebra modality)**.** *A coalgebra modality on a symmetric monoidal category is a tuple* $(!, \mathbf{p}, \mathbf{d}, \mathbf{c}, \mathbf{w})$ *such that,* $(!, \mathbf{p}, \mathbf{d})$ *is a comonad and* $\mathbf{c}_A : !A \to !A \otimes !A$ *and* $\mathbf{w}_A : !A \to 1$ *are two natural transformations making all* $(!A, \mathbf{c}_A, \mathbf{w}_A)$ *a cocommutative comonoid. This means that for all A object, the following diagram commutes:*



Moreover, we require that $\mathbf{p}$ *preserve the comultiplication, that is, that the following diagram commutes:*

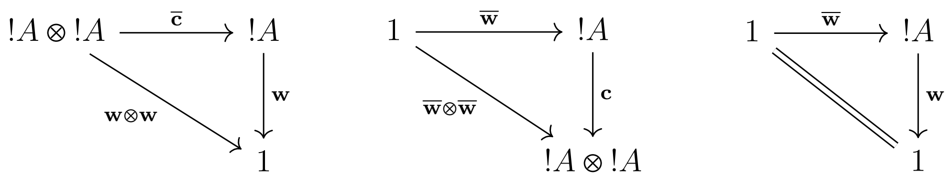


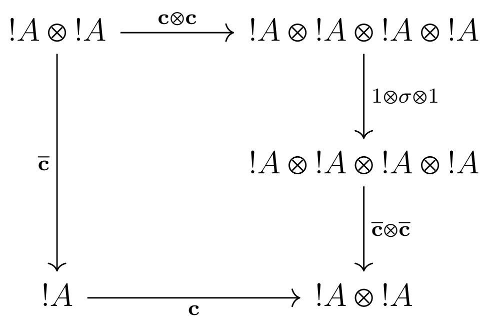
Lemma 3.2.6. *For a coalgebra modality* $(!, \mathbf{p}, \mathbf{d}, \mathbf{c}, \mathbf{w})$*,* $\mathbf{p}$ *also preserves the counit* $\mathbf{w}_A$*, the following diagram commutes:*



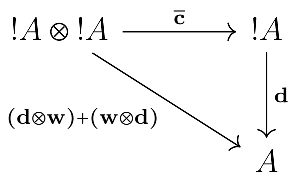
Definition 3.2.7 (bialgebra modality)**.** *A coalgebra modality on a pre-additive symmetric monoidal category is a tuple* $(!, \mathbf{p}, \mathbf{d}, \mathbf{c}, \mathbf{w}, \overline{\mathbf{c}}, \overline{\mathbf{w}})$ *where,* $(!, \mathbf{p}, \mathbf{d}, \mathbf{c}, \mathbf{w})$ *is a coalgebra modality,* $\overline{\mathbf{c}}_A : !A \otimes !A \to !A$ *and* $\overline{\mathbf{w}}_A : 1 \to !A$ *are natural transformations such that, for all A,* $(!A, \overline{\mathbf{c}}_A, \overline{\mathbf{w}}_A)$ *is a comutative monoid (that is, satisfying the opposite diagrams of 3.2.5) and* $(!A, \mathbf{c}_A, \mathbf{w}_A, \overline{\mathbf{c}}_A, \overline{\mathbf{w}}_A)$ *is a bialgebra, that is, the following diagrams commutes:*

[10] We refer the reader to section 1 where we discus the notion of model and completeness.

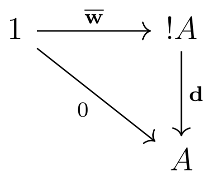




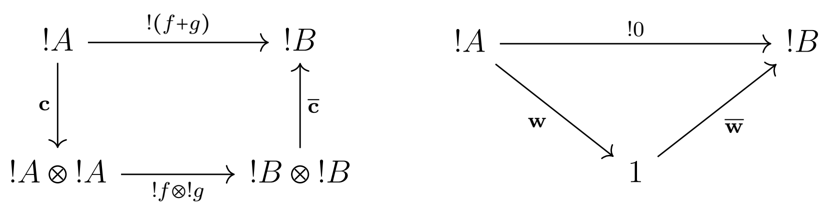
Moreover, we require that $\mathbf{d}$ *be compatible with* $\overline{\mathbf{c}}$ *meaning that the following diagram commutes:*



Lemma 3.2.8. *For a bialgebra modality* $(!, \mathbf{p}, \mathbf{d}, \mathbf{c}, \mathbf{w}, \overline{\mathbf{c}}, \overline{\mathbf{w}})$*,* $\mathbf{d}$ *also preserves the unit* $\overline{\mathbf{w}}_A$*, the following diagram commutes:*



Definition 3.2.9 (Additive bialgebra modality)**.** *An additive bialgebra modality on a pre-additive symmetric monoidal category is a bialgebra modality* $(!, \mathbf{p}, \mathbf{d}, \mathbf{c}, \mathbf{w}, \overline{\mathbf{c}}, \overline{\mathbf{w}})$ *compatible with the pre-additive structure, that is, the following diagrams commutes:*



Definition 3.2.10. *In a symmetric monoidal category with finite product* $\times$ *and terminal object* $\top$*, a coalgebra modality has Seely isomorphisms if the maps*

$$\chi_\top : !\top \xrightarrow{\mathbf{w}} 1 \qquad \chi : !(A \times B) \xrightarrow{\mathbf{c}} !(A \times B) \otimes !(A \times B) \xrightarrow{!\pi_1 \otimes !\pi_2} !A \otimes !B$$

are isomorphisms.

Definition 3.2.11 (Storage modality)**.** *A storage modality on a symmetric monoidal category with finite products is a coalgebra modality* $(!, \mathbf{p}, \mathbf{d}, \mathbf{c}, \mathbf{w})$ *that has Seely isomorphisms.*

Proposition 3.2.12 (Monoidality of $!$)**.** *Every storage modality gives rise to a lax monoidal structure* m *on* $!$ *defined as*

$$m : !A \otimes !B \xrightarrow{\chi^{-1}} !(A \times B) \xrightarrow{\mathbf{p}} !!(A \times B) \xrightarrow{!\chi} !(!A \otimes !B) \xrightarrow{!(\mathbf{d} \otimes \mathbf{d})} !(A \otimes B)$$

$$m_1 : 1 \xrightarrow{\chi_\top^{-1}} !\top \xrightarrow{\mathbf{p}} !!\top \xrightarrow{!\chi_\top} !1$$

Theorem 3.2.13. *The following are equivalent:*

- *An additive category equipped with a storage modality.*
- *An additive category equipped with an additive bialgebra modality.*

Moreover, there is a bijective correspondence between them.

Proof. Given a storage modality $(!, \mathbf{p}, \mathbf{d}, \mathbf{c}, \mathbf{w})$, we must construct two natural transformations $\overline{\mathbf{c}}$ and $\overline{\mathbf{w}}$ in order to get a bialgebra modality. We can do it using the additive structure and the Seely maps:

$$\overline{\mathbf{c}}_A : !A \otimes !A \xrightarrow{\chi^{-1}} !(A \times A) \xrightarrow{!\nabla_\times} !A \qquad \overline{\mathbf{w}}_A : 1 \xrightarrow{\chi_\top^{-1}} !\top \xrightarrow{!0} !A$$

Given an additive bialgebra modality $(!, \mathbf{p}, \mathbf{d}, \mathbf{c}, \mathbf{w}, \overline{\mathbf{c}}, \overline{\mathbf{w}})$, it's also a coalgebra modality, what is left to show is that it has Seely isomorphisms. Using the additive structure, $\overline{\mathbf{c}}$ and $\overline{\mathbf{w}}$, we construct inverses of the Seely maps:

$$\chi^{-1} : !A \otimes !B \xrightarrow{!\iota_1 \otimes !\iota_2} !(A \times B) \otimes !(A \times B) \xrightarrow{\overline{\mathbf{c}}} !(A \times B) \qquad \chi_\top^{-1} : 1 \xrightarrow{\overline{\mathbf{w}}} !\top$$

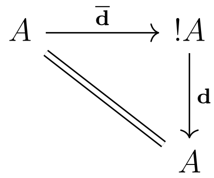
The proof of bijective correspondance can be found in [1]. ☐

Definition 3.2.14 (Differential storage category)**.** *A differential storage category is an additive category equipped with an additive bialgebra modality* $(!, \mathbf{p}, \mathbf{d}, \mathbf{c}, \mathbf{w}, \overline{\mathbf{c}}, \overline{\mathbf{w}})$*, that comes equipped with a natural transformation* $\overline{\mathbf{d}}_A : A \to !A$ *such that, the following diagrams commutes:*

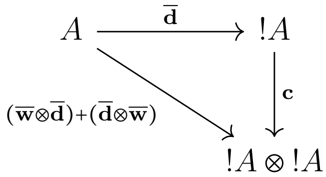
- *The chain rule*

$$\begin{array}{ccccc}
!A \otimes A & \xrightarrow{A \otimes \overline{\mathbf{d}}} & !A \otimes !A & \xrightarrow{\overline{\mathbf{c}}} & !A \\
\downarrow{\scriptstyle \mathbf{c} \otimes \overline{\mathbf{d}}} & & & & \\
!A \otimes !A \otimes !A & & & & \downarrow{\scriptstyle \mathbf{p}} \\
\downarrow{\scriptstyle A \otimes \overline{\mathbf{c}}} & & & & \\
!A \otimes !A & \xrightarrow[\mathbf{p} \otimes \overline{\mathbf{d}}]{} & !!A \otimes !!A & \xrightarrow[\overline{\mathbf{c}}]{} & !!A
\end{array}$$

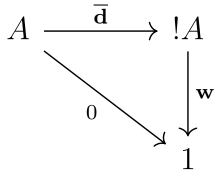
- *The linear rule*



- *The product rule*



- *The constant rule*



Theorem 3.2.15. *$\star$-autonomous differential storage categories are models of DiLL. The interpretation map is given by our choice of notations (see 2.2.4).*

Remark 3.2.16. Differential storage categories are models of LL because they are linear categories (they are not Seely categories). We did not give a definition of linear categories as it's not central to the subject of this memoir (it can be found in [14]). Like a Seely category, a linear category is a category satisfying conditions garantying it's involved in a linear-non-linear adjunction (see section 2), in this case with its Eilenberg-Moore category.

3.3 The relational model, an example of models of DILL

Now that we gave a definition of models of DiLL, differential storage categories, we provide an example: the category **Rel** (defined in 2.4). The choice to explicit this example can be seen as paradoxal since the intuition given for DiLL is based on functional analysis yet, **Rel** is not of topological or vectorial nature. There exists in the literature multiple examples of models from functional analysis:

- Blute, Ehrhard and Tasson describe in [3] models in terms of convenient vector spaces. Convenient vector spaces are vectors spaces endowed with a bornology (a structure between the notions of topological vector spaces and normed vector space, that allows one to talk about bounded sets without the need for a norm) and additional conditions.

- Dabrowski and Kerjean describe in [4] models in terms of particular locally convex separated topological vector spaces.
- Kerjean and Tasson describe in [11] in terms of particular bornological vectors spaces.

Since those models are quite mathematically complex, they take lots of pages to properly describe and explain. On the other hand, **Rel** is quite simple and is an example of most notions seen throughout this memoir.

Rel is a differential storage category. First, for the additive structure, the addition of maps $R + V$ is defined as the union $R \cup V$, the 0 morphism is the empty relation and the finite biproduct is given by the disjoint union. Second, the cocontraction $\overline{\mathbf{c}}_X : !X \otimes !X \to !X$, coweakening $\overline{\mathbf{w}}_X : 1 \to !X$ and coderelection $\overline{\mathbf{d}}_X : X \to !X$ are defined as:

$$\begin{aligned} \overline{\mathbf{c}}_X &:= \{((m_1, m_2), m_1 + m_2) \mid m_1, m_2 \in !X\} \\ \overline{\mathbf{w}}_X &:= \{(\star, [])\} \\ \overline{\mathbf{d}}_X &:= \{(x, [x]) \mid x \in X\} \end{aligned}$$

One easily check that, in addition to what has been defined in 2.4, this makes **Rel** a $\ast$-autonomous differential storage category.

4 Codigging, monads and quantitative semantics

In this section we present codigging, we explain why it corresponds to a notion of quantitative differentiation and why it seams possible to integrate it into functional programming languages.

4.1 Codigging

This section is an explanation of some key ideas of [10].

When looking at the semantics of DiLL one sees that, $(!, \mathbf{p}, \mathbf{d})$ is a comonad, $(!X, \mathbf{c}_X, \mathbf{w}_X)$ is a cocomutative comonoid and $(!X, \overline{\mathbf{c}}_X, \overline{\mathbf{w}}_X)$ is a commutative monoid. On the other hand, the coderelection rule looks like it could be a unit of a monad $(!, -, \overline{\mathbf{d}})$ but the multiplication map has no reason to exists in an arbitrary model. Having presented things this way, it seams natural that one looking for a dual rule to the digging would bet on a natural transformation of type $!!X \to !X$ making $(!, \overline{\mathbf{p}}, \overline{\mathbf{d}})$ a monad. Amazingly, as show by Kerjean and Lemay in [10], defining a "codigging" rule this way makes a lot of sense in some models.

First let's define the codigging rule and models of DiLL + codigging.

Definition 4.1.1 (Codigging)**.** *The codigging is the rule:*

$$\frac{\vdash \Gamma, ?A}{\vdash \Gamma, ??A}\ \overline{\mathbf{p}}$$

It's the dual of the digging rule of DiLL. An other way to present it, less logically correct but perhaps more intuitive, is:

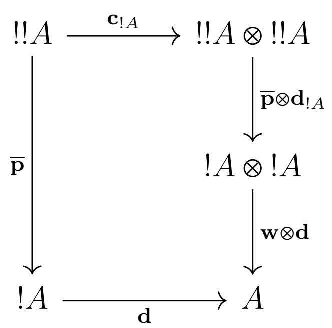
$$\frac{\vdash \Gamma, !!A}{\vdash \Gamma, !A}\ \overline{\mathbf{p}}$$

Remark 4.1.2. It's not yet known whether or not DiLL+$\overline{\mathbf{p}}$ enjoys the cut elimination property.

Definition 4.1.3 (Monadic differential category)**.** *A monadic differential category is a differential storage category equipped with a natural transformation $\overline{\mathbf{p}}_A : !!A \to !A$ called a codigging such that, $(!, \overline{\mathbf{p}}, \overline{\mathbf{d}})$ is a monad, $\overline{\mathbf{p}}$ is a monoid morphism, that is, the following diagrams commutes:*

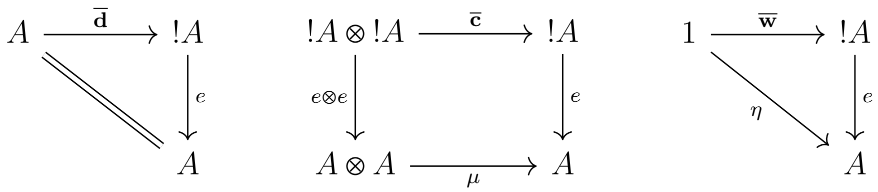
$$\begin{array}{ccc} !!A \otimes !!A & \xrightarrow{\overline{\mathbf{c}}_{!A}} & !!A \\ \downarrow{\scriptstyle \overline{\mathbf{p}}\otimes\overline{\mathbf{p}}} & & \downarrow{\scriptstyle \overline{\mathbf{p}}} \\ !A \otimes !A & \xrightarrow[\overline{\mathbf{c}}]{} & !A \end{array} \qquad \begin{array}{ccc} 1 & \xrightarrow{\overline{\mathbf{w}}_{!A}} & !!A \\ & \searrow{\scriptstyle \overline{\mathbf{w}}} & \downarrow{\scriptstyle \overline{\mathbf{p}}} \\ & & !A \end{array}$$

We also require that $\overline{\mathbf{p}}$ be compatible with $\mathbf{d}$, meaning that the following diagram commutes:



This definition is simply an extension of the notion of differential storage category by dualising the digging following the already established pattern. Notice that the conditions required for $\overline{\mathbf{p}}$ are simply the dual conditions required for $\mathbf{p}$. Intuitively, one can think about codigging as an generalised exponential map from $!A \Rightarrow !A$. The exponential function $x \mapsto e^x$ has three key properties: it's its own derivative, $e^{x+y} = e^x e^y$ and $e^0 = 1$. Those properties are capture in the categorical formalism by the following definition:

Definition 4.1.4 (!-differential exponential map)**.** *In a differential storage category, a commutative monoid (A, μ, η) is said to have a !-differential exponential map if there exists a morphism $e : !A \to A$ such that the following diagram commutes:*



A !-differential exponential algebra is a commutative monoid equipped with a !-differential exponential map.

Rewriting those diagrams with the interpretations in terms of functional analysis, we get $D_0(e)(v) = v$, $e^{\phi * \psi} = e^{\phi} \cdot e^{\phi}$ and $e^{\delta_0} = 1$. The following results makes more rigorous our intuition about codigging:

Theorem 4.1.5. *In a monadic differential category, all tuples $(!A, \overline{\mathbf{c}}_A, \overline{\mathbf{w}}_A, \overline{\mathbf{p}}_A)$ are !-differential exponential algebra.*

Proof. The first diagram is exactly one of the monad axioms $\overline{\mathbf{p}}_A \circ \overline{\mathbf{d}}_{!A} = \mathrm{id}_{!A}$. The last two diagrams are exactly the first two diagrams of definition 4.1.3. □

Codigging captures a generalised notion of exponentiation and also the notion of Taylor expansion. To understand why, we can first look at the monad axiom $\overline{\mathbf{p}}_A \circ !\overline{\mathbf{d}}_A = \mathrm{id}_{!A}$. Recall that coderelection is the function $x \mapsto D_0(-)(x)$ then rewriting the monad axiom we get:

$$\delta_x \mapsto \overline{\mathbf{p}}_A(D_0(-)(x)) = \mathrm{id}_{!A}$$

From this equation and the interpretation of codigging as an exponential function, one can guess that codigging is in fact the following function:

$$\overline{\mathbf{p}}_A = \delta_\phi \mapsto \sum_{n=0}^{\infty} \frac{\phi^{*n}}{n!}$$

Where $\phi^{*n} := \phi * \cdots * \phi$. This guess intuitively makes sense since $\delta_x \mapsto \sum_{n=0}^{\infty} \frac{D_0(-)(x)^{*n}}{n!}$ is the Taylor expansion (at point 0) of a function, this formula is indeed the identity on distributions and in models where every function is equal to its Taylor expansion (quantitative models) it's the identity. Making this intuition more rigorous means finding a categorical setting expressing the notion of "Taylor expansion" and restricting our function spaces such that each function is equal to its Taylor expansion at point 0.

To be able to talk about Taylor monomial, we need to able to multiply by $\frac{1}{n!}$. This can be done by requiring our model category to be enriched over $\mathbb{Q}_{\geq 0}$-modules. This means that each Hom-set has a $\mathbb{Q}_{0<}$-module structure and the composition operation is $\mathbb{Q}_{\geq 0}$-bilinear. A category satisfying the same axioms as a differential storage category and enriched on $\mathbb{Q}_{\geq 0}$-modules is called a $\mathbb{Q}_{\geq 0}$-differential storage category[11]. In the setting of $\mathbb{Q}_{\geq 0}$-differential storage categories, we can always construct morphisms $M_A^n : !A \to !A$ such that, when precomposed with a morphisms $f : !A \to B$, it corresponds to the n^{th} Taylor monomial of f. We can then define $T_A^n := \sum_{i=0}^{n} M_A^i$ the n^{th} Taylor polynomial morphism. This leads to the following definition:

Definition 4.1.6. *Taylor differential category A Taylor differential category is a $\mathbb{Q}_{\geq 0}$-differential storage category such that for any pair of parallel coKleisli maps $f : !A \to B$, $g : !A \to B$:*

$$(\forall n \in \mathbb{N},\ f \circ M_A^n = g \circ M_A^n) \Rightarrow f = g$$

The following two results make rigorous our intuition:

Proposition 4.1.7. *A Taylor differential category is a monadic differential category if and only if, for all object A, there exists a (necessarily unique) morphism $\overline{\mathbf{p}}_A : !!A \to !A$ such that,*

$$\forall n \in \mathbb{N},\ \overline{\mathbf{p}}_A \circ M_{!A}^n = \frac{1}{n!}(\overline{\mathbf{c}}_A^n \circ \mathbf{d}_{!A}^n)$$

Proposition 4.1.8. *In a Taylor differential category which is also a monadic differential category, the following series converges to the codigging $\overline{\mathbf{p}}$ with respect to ultrametric $\mathcal{D}$:*

$$\overline{\mathbf{p}}_A = \sum_{n=0}^{\infty} \frac{1}{n!}(\overline{\mathbf{c}}_A^n \circ \mathbf{d}_{!A}^{\otimes^n} \circ \mathbf{c}_{!A}^n)$$

[11] In particular, it's a differential storage category since $\mathbb{Q}_{\geq 0}$-modules are in particular commutatives monoids

Not all objects appearing in those two propositions were defined here but, we hope it's enough to convince the reader that the setting of Taylor differential categories provides a formal setting in which the codigging corresponds to the intuitions formulated. Moreover, notice that this setting forces every function to be equal to its Taylor series. It's a form of quantitative semantics: interpreting programs by functions equal to their Taylor series, each Taylor monomial containing some information on the program. This quantitative setting of differentiation is captured (at least in part) by a monad $(!, \overline{\mathbf{p}}, \overline{\mathbf{d}})$, which is a structure that can be integrated to a functional programming language (under some conditions). This well known fact leads to the hope of finding a calculus, here we are interested in a "resource-calculus", that capture in its semantics this "quantitative differentiation".

Before moving on to the next section we provide an example of a Taylor differential category which is also a monadic differential category: **Rel**. **Rel** is a $\mathbb{Q}_{\geq 0}$-differential storage category with the $\mathbb{Q}_{\geq 0}$-module enriched structure given by $\frac{q}{p} \cdot R = R$ if $\frac{q}{p} \neq 0$ and $0 \cdot R = 0$. Each M_A^n is defined as $M_A^n = \{(m,m) | \forall m \in !A, |m| = n\}$, where $|m|$ denotes the cardinality of the multiset. This makes **Rel** into a Taylor differential category. It's also a monadic differential category with codigging defined as

$$\overline{\mathbf{p}}_A = \{([m_1, \dots, m_n], m_1 + \cdots + m_n) \mid \forall n \in \mathbb{N}, \forall 0 < i \leq n, m_i \in !A\}$$

4.2 Monads in functional programming

In this section we explain how monads can be integrated to functional programming languages (= typed theories). For more details see [15]. First, we need some categorical definitions.

Definition 4.2.1 (Kleisli triple)**.** *Let $\mathcal{C}$ be a category then, a Kleisli triple is a tuple $(T, \eta, -^*)$ where T is a function from Obj$(\mathcal{C})$ to Obj$(\mathcal{C})$, η a natural transformation from $\mathbb{1}_{\mathcal{C}}$ to T and $-^*$ is a operation on morphisms such that, for all $f : A \to TB$, $f^* : TA \to TB$. We also require that the following equalities hold:*

- $\eta_A^* = \mathrm{id}_A$
- $f^* \circ \eta_A = f$
- $g^* \circ f^* = (g^* \circ f)^*$

Remark 4.2.2. For the reader familiar with some functional programming languages (like Haskel), this corresponds to the operations:

$$a : A \mapsto [a] : TA \qquad \frac{a : A \mapsto f(a) : TB}{c : TA \mapsto (\text{let } c \Rightarrow x \text{ in} f(x)) : TB}$$

Proposition 4.2.3. *There is a one to one correspondance between Kleisli triple and monads.*

Proof. Let $(T_K, \eta, -^*)$ be a Kleisli triple then, the associated monad is (T_m, η, μ) where T_m is the extension of T on morphisms by $T(f) := (\eta \circ f)^*$ and $\mu_A := \mathrm{id}^*_{TA}$. Let (T_m, η, μ) be a monad then, the associated Kleisli triple is $(T_K, \eta, -^*)$ where T_K is the restriction of T_m on objects and $f^* := \mu_B \circ Tf$ □

Kleisli triples provides an alternative formalism for monads, a formalism more suited for implementation in a type theory. For example, consider the following theory:

$$\frac{}{\vdash A : \text{Type}}\ \text{A} \qquad \frac{\vdash \tau : \text{Type}}{\vdash T\tau : \text{Type}}\ \text{T} \qquad \frac{\vdash \tau : \text{Type}}{x : \tau \vdash x : \tau}\ \text{var}$$

$$\frac{x : \tau \vdash e : \tau_1}{x : \tau \vdash f(e) : \tau_2}\ f : \tau_1 \to \tau_2 \qquad \frac{x : \tau \vdash e : \tau_1}{x : \tau \vdash [e]_T : T\tau_1}\ [-]_T$$

$$\frac{x : \tau \vdash e_1 : T\tau_1 \qquad x_1 : \tau_1 \vdash e_2 : T\tau_2}{x : \tau \vdash (\text{let } e_1 \Rightarrow x_1 \text{ in } e_2) : T\tau_2}\ \text{let} \qquad \frac{x : \tau \vdash e_1 : \tau_1 \qquad x : \tau \vdash e_2 : \tau_1}{x : \tau \vdash e_1 =_{\tau_1} e_2}\ \text{eq}$$

$$\frac{x : \tau \vdash e_1 =_{\tau_1} e_2}{x : \tau \vdash [e_1]_T =_{T\tau_1} [e_2]_T}\ [-].\xi \qquad \frac{x : \tau \vdash e_1 =_{T\tau_1} e_2 \qquad x' : \tau_1 \vdash e'_1 =_{T\tau_2} e'_2}{x : \tau \vdash (\text{let } e_1 \Rightarrow x' \text{ in } e'_1) =_{T\tau_2} (\text{let } e_2 \Rightarrow x' \text{ in } e'_2)}\ \text{let}.\xi$$

$$\frac{x : \tau \vdash e_1 : T\tau_1 \qquad x_1 : \tau_1 \vdash e_2 : T\tau_2 \qquad x_2 : \tau_2 \vdash e_3 : T\tau_3}{x : \tau \vdash (\text{let } (\text{let } x_1 \Rightarrow e_1 \text{ in } e_2) \Rightarrow x_2 \text{ in } e_3) =_{T\tau_3} (\text{let } e_1 \Rightarrow x_1 \text{ in } (\text{let } e_2 \Rightarrow x_2 \text{ in } e_3))}\ \text{ass}$$

$$\frac{x : \tau \vdash e_1 : \tau_1 \qquad x_1 : \tau_1 \vdash e_2 : T\tau_2}{x : \tau \vdash x : \tau \vdash (\text{let } [e_1]_T \Rightarrow x_1 \text{ in } x_2) =_{T\tau_2} e_2[e_1/x_1]}\ \text{T}.\beta$$

$$\frac{x : \tau \vdash e_1 : T\tau_1}{x : \tau \vdash (\text{let } e_1 \Rightarrow x_1 \text{ in } [x_1]_T) =_{T\tau_1} e_1}\ \text{T}.\eta$$

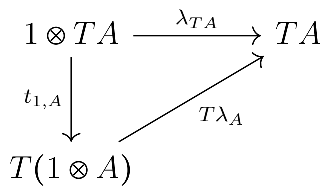
It's a very simple theory providing a framework to implement different monads. To implement a monad one must specify additional reduction rules capturing its behaviour. For example, lets consider the following monad

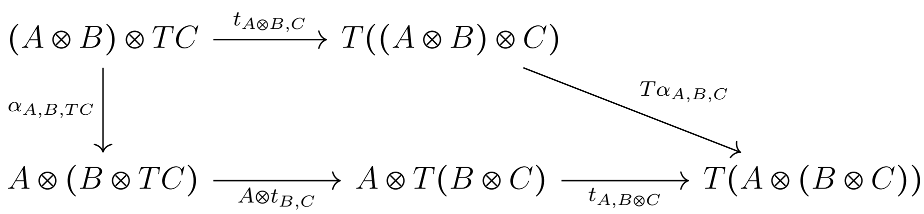
Definition 4.2.4 (Partiality monad)**.** *To a set A it associates the set $A + \{\bot\}$ denoted $A_\bot$ where $+$ is the disjoint union and $\bot$ is a fixed set. It defines a Kleisli triple where η_A is the inclusion from A to $A_\bot$ and f^* is defined as: $f^*\bot = \bot$ and $f^*(a) = f(a)$.*

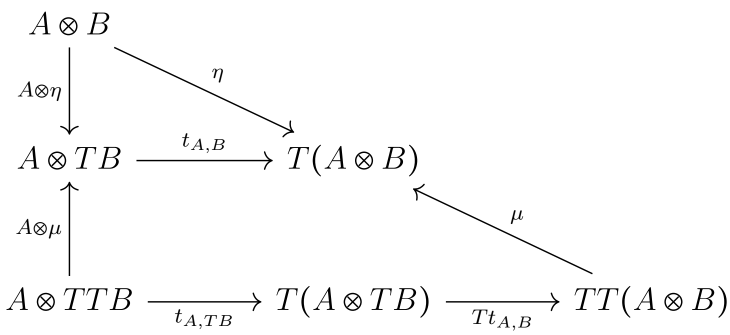
To implement it, we must add the rules of the form $(\text{let } \bot \Rightarrow x \text{ in } f(x)) = \bot$ and $(\text{let } a \Rightarrow x \text{ in } f(x)) = f(a)$.

When trying to generalise to a more complex theory, allowing multiple variable in the context, we run into a problem in the semantics. A sequent of the form $A, B, C \vdash D$, will be interpreted using the monoidal structure $A \otimes B \otimes C \to D$ hence, we need compatibility conditions between the monad structure and the monoidal structure of the category. The right compatibility conditions are given in the definition of Strong monads:

Definition 4.2.5 (Strong monad)**.** *A strong monad on a monoidal category $\mathcal{C}$ is a monad (T, η, μ), together with a natural transformation $t_{A,B} : A \otimes TB \to T(A \otimes B)$, called a strength, such that, the following diagrams commutes:*

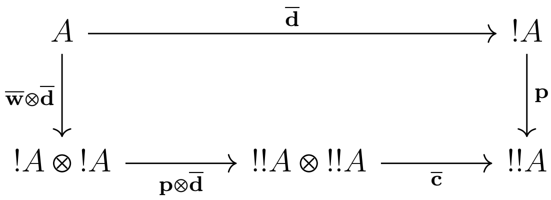




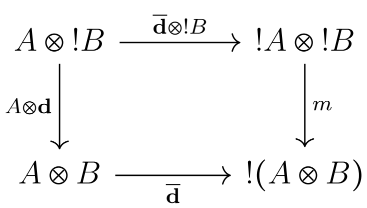


Is codigging a strong monad? From [8] and [1], we know that we can equivalently replace the chain rule by an alternative chain rule and a strength rule (also called a monoidal rule):

- Alternative chain rule

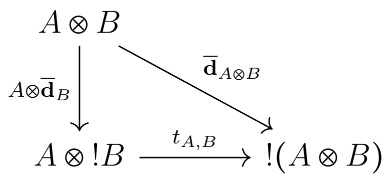


- Strength rule (monoidal rule)

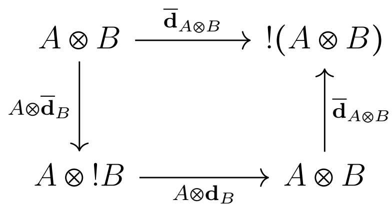


By defining $t_{A,B}$ as $m \circ (\overline{\mathbf{d}} \otimes !B) : A \otimes !B \to !(A \otimes B)$ and using the strength rule we can prove that $t_{A,B}$ satisfies all of the diagrams of definition 4.2.5 where the codigging dose not appear.

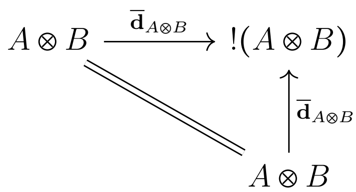
For example, we explicit the commutation in the top part of the last diagram:



Using the strength rule, we rewrite it as



Using the fact that $\otimes$ is a bifunctor and the linear rule (3.2.14) we obtain



Which is of course a commutative diagram.

4.3 A programming language for quantitative differentiation ?

The goal is to find a typed theory with a monad capturing in its semantics quantitative differentiation. To find it, we looked at what already existed in close proximity to LL and DiLL: ressource λ-calculus.

Ressource λ-calculus is a form of λ-calculus where redexes are formed from and abstraction $(\lambda x.t)$ and a bag $[u_1, \ldots, u_n]$. The key idea is that, much like in LL, we care about ressources: how many times "x" appears in the construction of "t". If "x" appears 3 times then, we need to provide 3 inputs, a bag of size 3: $[u_1, u_2, u_3]$, for the substitution to make sense. This kind of substitution is called (without surprises) linear substitution. This kind of calculus is in essence quantitative hence, my hope of being able to endow it with the codigging monad. An example of untyped ressource λ-calculus can be found in [17]. A problem I faced was in finding a typed system for ressource λ-calculus as the codigging monad acts on linear types. My attention was also grabbed by differential λ-calculus (introduced in 2004 by Ehrhard [6]) which allows to interpret ressource calculus in terms of differential nets. Much like differential λ-calculus, it's not improbable to be able to find a calculus that would allow to interpret ressource calculus and some monadic untyped structure in terms of DiLL + codigging.

In the future, I plan to investigate when the codigging monad is strong. I also plan to investigate models of ressource calculus in hope of finding intersections with models of codigging.

References

[1] Blute, R. F. ; Cockett, J. R. B. ; Lemay, J.-S. P. ; Seely, R. A. G.: *Differential Categories Revisited*. http://dx.doi.org/10.48550/arXiv.1806.04804. Version: Mai 2019. – arXiv:1806.04804 [math]

[2] Blute, R. F. ; Cockett, J. R. B. ; Seely, R. A. G.: Differential categories. In: *Mathematical Structures in Computer Science* 16 (2006), Dezember, Nr. 06, 1049. http://dx.doi.org/10.1017/S0960129506005676. – DOI 10.1017/S0960129506005676. – ISSN 0960–1295, 1469–8072

[3] Blute, Richard ; Ehrhard, Thomas ; Tasson, Christine: A convenient differential category.

[4] Dabrowski, Yoann ; Kerjean, Marie: Models of linear logic based on the Schwartz epsilon-product. In: *Theory and Applications of Categories* (2020)

[5] Ehrhard, Thomas: An introduction to differential linear logic: proof-nets, models and antiderivatives. In: *Mathematical Structures in Computer Science* 28 (2018), August, Nr. 7, 995–1060. http://dx.doi.org/10.1017/S0960129516000372. – DOI 10.1017/S0960129516000372. – ISSN 0960–1295, 1469–8072

[6] Ehrhard, Thomas ; Regnier, Laurent: The differential lambda-calculus. In: *Theoretical Computer Science* 309 (2003), Dezember, Nr. 1-3, 1–41. http://dx.doi.org/10.1016/S0304-3975(03)00392-X. – DOI 10.1016/S0304–3975(03)00392–X. – ISSN 03043975

[7] Ehrhard, Thomas ; Regnier, Laurent: Differential interaction nets. In: *Theoretical Computer Science* 364 (2006), November, Nr. 2, 166–195. http://dx.doi.org/10.1016/j.tcs.2006.08.003. – DOI 10.1016/j.tcs.2006.08.003. – Publisher: Elsevier

[8] Fiore, Marcelo P.: Differential Structure in Models of Multiplicative Biadditive Intuitionistic Linear Logic: (Extended Abstract). Version: 2007. http://link.springer.com/10.1007/978-3-540-73228-0_13. In: *Typed Lambda Calculi and Applications* Bd. 4583. Berlin, Heidelberg : Springer Berlin Heidelberg, 2007. – ISBN 978–3–540–73227–3 978–3–540–73228–0, 163–177. – Series Title: Lecture Notes in Computer Science

[9] Girard, Jean-Yves: Linear logic. In: *Theoretical Computer Science* 50 (1987), Januar, Nr. 1, 1–101. http://dx.doi.org/10.1016/0304-3975(87)90045-4. – DOI 10.1016/0304–3975(87)90045–4. – ISSN 0304–3975

[10] Kerjean, Marie ; Lemay, Jean-Simon P.: Taylor Expansion as a Monad in Models of DiLL. In: *2023 38th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*. Boston, United States : IEEE, Juni 2023, 1–13

[11] Kerjean, Marie ; Tasson, Christine: Mackey-complete spaces and power series – a topological model of differential linear logic. In: *Mathematical Structures in Computer Science* 28 (2018), April, Nr. 4, 472–507. http://dx.doi.org/10.1017/S0960129516000281. – DOI 10.1017/S0960129516000281. – ISSN 0960–1295, 1469–8072

[12] Laurent, Olivier: Théorie de la démonstration. https://perso.ens-lyon.fr/olivier.laurent/thdem.pdf

[13] Mac Lane, Saunders: *Categories for the working mathematician*. 2nd ed. New York : Springer, 1998 (Graduate texts in mathematics 5). – ISBN 978–0–387–98403–2

[14] Melliès, Paul-André: Categorical Semantics of Linear Logic. https://www.irif.fr/~mellies/papers/panorama.pdf

[15] Moggi, Eugenio: Notions of computation and monads. In: *Information and Computation* 93 (1991), Juli, Nr. 1, 55–92. http://dx.doi.org/10.1016/0890-5401(91)90052-4. – DOI 10.1016/0890–5401(91)90052–4. – ISSN 08905401

[16] Pagani, Michele: The Cut-Elimination Thereom for Differential Nets with Promotion. In: *9th International Conference on Typed Lambda Calculi and Applications (TLCA 2009)* Bd. 5608. Brasilia, Brazil : Springer, Juli 2009 (Lectures Notes in Computer Science), 219–233

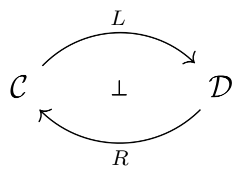
[17] Pagani, Michele ; Tranquilli, Paolo: Parallel Reduction in Resource Lambda-Calculus. Version: 2009. http://link.springer.com/10.1007/978-3-642-10672-9_17. In: *Programming Languages and Systems* Bd. 5904. Berlin, Heidelberg : Springer Berlin Heidelberg, 2009. – ISBN 978–3–642–10671–2 978–3–642–10672–9, 226–242. – Series Title: Lecture Notes in Computer Science

A Notions of category theory

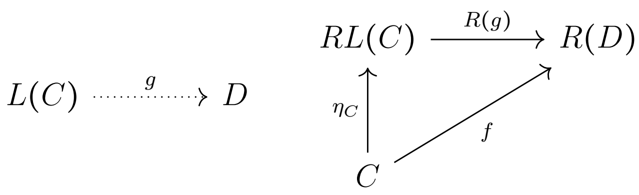
In this appendix we recall some well know notions of category theory and others the reader might not be familiar with. Often, theorems will be given without proofs as they are easily found in the literature, see nLab, [13] or [14]. Moreover, concepts wont be defined in the outmost generality, they will be defined in the context that is of interest for this memoir.

A.1 Adjunction, monad and comonad

Definition A.1.1 (Adjunction). *Let $\mathcal{C}$ and $\mathcal{D}$ be two categories, $L : \mathcal{C} \to \mathcal{D}$ and $R : \mathcal{D} \to \mathcal{C}$ two functors. We say that L is left adjoint to R (or equivalently that R is a right adjoint to L) denoted $L \dashv R$ and diagrammatically written*

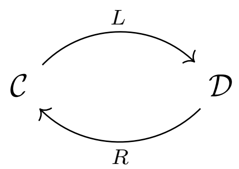


if there exists a natural transformation $\eta : \mathbb{1}_{\mathcal{C}} \Rightarrow R \circ L$ such that for any $C \in \mathcal{C}$, $D \in \mathcal{D}$ and $f : C \to R(D)$, there exists a unique $g : L(C) \to D$ such that $f = R(g) \circ \eta_C$



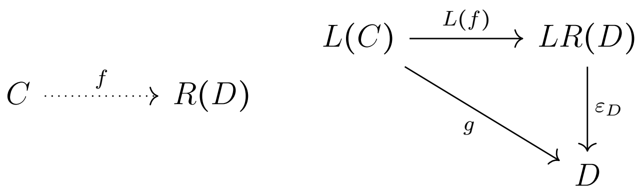
η is called the unit of the adjunction.

Theorem A.1.2. *Let $\mathcal{C}$ and $\mathcal{D}$ be two categories, $L : \mathcal{C} \to \mathcal{D}$ and $R : \mathcal{D} \to \mathcal{C}$ two functors.*



The following are equivalent:

- $L \dashv R$
- *There exists a natural transformation $\varepsilon : L \circ R \to \mathbb{1}_{\mathcal{D}}$, called a counit, such that for any $C \in \mathcal{C}$, $D \in \mathcal{D}$ and $G : L(C) \to D$, there exists a unique $f : C \to R(D)$ such that $g = \varepsilon_D \circ L(f)$*



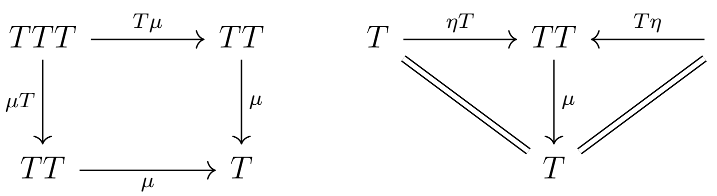
- *For all $C \in \mathcal{C}$ and $D \in \mathcal{D}$, there exists an isomorphism*

$$\phi : \mathrm{Hom}_{\mathcal{D}}(L(C), D) \simeq \mathrm{Hom}_{\mathcal{C}}(C, R(D))$$

natural in both C and D.

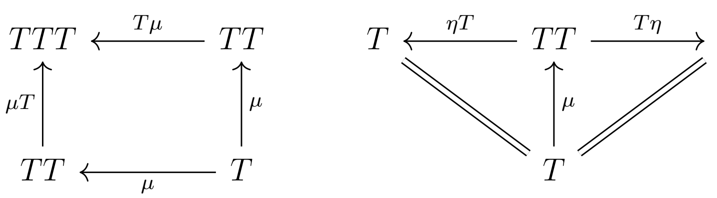
Definition A.1.3 (Monad)**.** *Let $\mathcal{C}$ be a category then, a monad on $\mathcal{C}$ is a triplet (T, μ, η) such that:*

- *T is a $\mathcal{C}$ endofunctor.*
- *μ is a natural transformation from $TT \Rightarrow T$ called the multiplication.*
- *η is a natural transformation from $\mathbb{1}_{\mathcal{C}} \Rightarrow T$ called the unit.*
- *The following associativity and unit law diagrams commutes:*



Definition A.1.4 (Comonad)**.** *Let $\mathcal{C}$ be a category then, a comonad on $\mathcal{C}$ is a triplet (T, μ, η) such that:*

- *T is a $\mathcal{C}$ endofunctor.*
- *μ is a natural transformation from $T \Rightarrow TT$ called the comultiplication.*
- *η is a natural transformation from $T \Rightarrow \mathbb{1}_{\mathcal{C}}$ called the counit.*
- *The following coassociativity and counit law diagrams commutes:*



Theorem A.1.5. *Let $\mathcal{C}$ and $\mathcal{D}$ be two categories, $L : \mathcal{C} \to \mathcal{D}$ and $R : \mathcal{D} \to \mathcal{C}$ two adjoint functors:*

$$\mathcal{C} \underset{R}{\overset{L}{\underset{\bot}{\rightleftarrows}}} \mathcal{D}$$

Then, $(R \circ L, R\varepsilon_L, \eta)$ is a monad on $\mathcal{C}$ and $(L \circ R, L\eta_R, \varepsilon)$ is a comonad on $\mathcal{D}$.

Definition A.1.6 (Kleisli category)**.** *Let (T, μ, η) be a monad on a category $\mathcal{C}$ then the Kleisli category of this monad is the category $\mathcal{C}_T$ defined as:*

- *The objects of $\mathcal{C}_T$ are the objects of $\mathcal{C}$.*
- *A morphism $A \to B$ in $\mathcal{C}_T$ is a morphism from $A \to TB$ in $\mathcal{C}$. If $f : A \to B$ and $g : B \to C$ are two morphisms in $\mathcal{C}_T$ then the composition is defined as $g \circ f := \mu_C \circ T(g) \circ f$.*

Theorem A.1.7. *Let (T, μ, η) be a monad on a category $\mathcal{C}$. Then, the functors*

- *$F_* : \mathcal{C} \to \mathcal{C}_T$ that maps A to A and $f : A \to B$ to $\eta_B \circ f : A \to T(B)$.*
- *$F^* : \mathcal{C}_T \to \mathcal{C}$ that maps A to TA and $f : A \to TB$ to $\mu_B \circ Tf : T(A) \to T(B)$.*

are adjoint functors $F_* \dashv F^*$ and the monad (T, μ, η) is the monad generated by this adjonction.

Definition A.1.8 (coKleisli category)**.** *Let (T, μ, η) be a comonad on a category $\mathcal{C}$ then the coKleisli category of this monad is the category $\mathcal{C}_T$ defined as:*

- *The objects of $\mathcal{C}_T$ are the objects of $\mathcal{C}$.*
- *A morphism $A \to B$ in $\mathcal{C}_T$ is a morphism from $TA \to B$ in $\mathcal{C}$. If $f : A \to B$ and $g : B \to C$ are two morphisms in $\mathcal{C}_T$ then the composition is defined as $g \circ f := g \circ Tf \circ \mu_A$.*

Theorem A.1.9. *Let (T, μ, η) be a comonad on a category $\mathcal{C}$. Then, the functors*

- *$F^* : \mathcal{C} \to \mathcal{C}_T$ that maps A to A and $f : A \to B$ to $f \circ \eta_A : TA \to B$.*
- *$F_* : \mathcal{C}_T \to \mathcal{C}$ that maps A to TA and $f : TA \to B$ to $Tf \circ \mu_A : T(A) \to T(B)$.*

are adjoint functors $F_* \dashv F^*$ and the comonad (T, μ, η) is the comonad generated by this adjonction.

A.2 Monoidal categories

Definition A.2.1 (Monoidal category)**.** *A monoidal category is a category* $\mathbb{C}$ *equipped with a bifunctor* $\otimes : \mathbb{C} \times \mathbb{C} \to \mathbb{C}$*, associative up to natural transformation and with unit* 1*. More formally, it is the data of* $\otimes$ *a bifunctor,* 1 *a distinguished object and tree natural isomorphism:*

$$\alpha_{A,B,C} : (A \otimes B) \otimes C \to A \otimes (B \otimes C) \qquad \lambda_A : 1 \otimes A \to A \qquad \rho_A : A \otimes 1 \to A$$

Such that, for all A, B, C *and* D*, the pentagram and the triangular diagrams commutes:*

$$\begin{array}{ccccc} & & (A\otimes B)\otimes(C\otimes D) & & \\ & \nearrow^{\alpha} & & \searrow^{\alpha} & \\ ((A\otimes B)\otimes C)\otimes D & & & & A\otimes(B\otimes(C\otimes D)) \\ {\scriptstyle \alpha\otimes D}\downarrow & & & & \uparrow{\scriptstyle A\otimes\alpha} \\ (A\otimes(B\otimes C))\otimes D & & \xrightarrow{\qquad\alpha\qquad} & & A\otimes((B\otimes C)\otimes D)\end{array}$$

and the triangular diagram commutes for all A *and* B

$$\begin{array}{ccc} (A\otimes 1)\otimes B & \xrightarrow{\alpha} & A\otimes(1\otimes B) \\ & {\scriptstyle \rho\otimes B}\searrow \quad \swarrow{\scriptstyle A\otimes\lambda} & \\ & A\otimes B & \end{array}$$

A monoidal category is said to be strict if α, λ *and* ρ *are the identity morphism.*

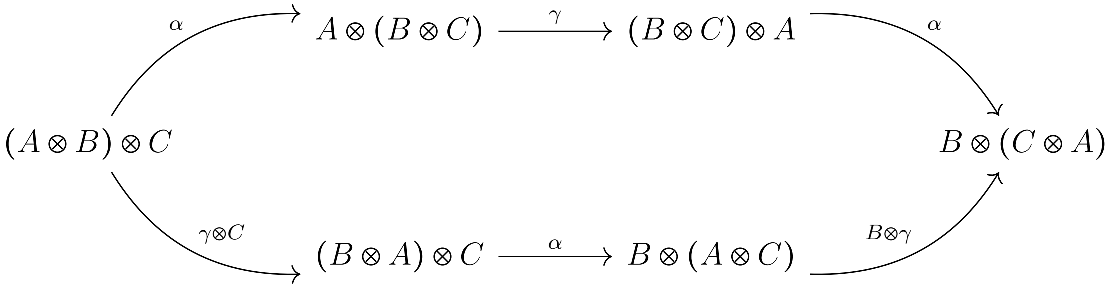
Proposition A.2.2. *In any monoidal category, the following triangles commutes:*

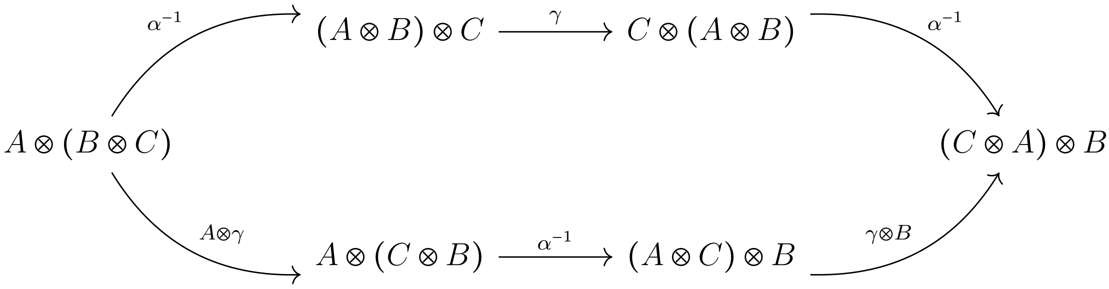
$$\begin{array}{ccc} (1\otimes A)\otimes B & \xrightarrow{\alpha} & 1\otimes(A\otimes B) \\ & {\scriptstyle \lambda\otimes B}\searrow \quad \swarrow{\scriptstyle \lambda} & \\ & A\otimes B & \end{array}$$

$$\begin{array}{ccc} (A\otimes B)\otimes 1 & \xrightarrow{\alpha} & A\otimes(B\otimes 1) \\ & {\scriptstyle \rho}\searrow \quad \swarrow{\scriptstyle A\otimes\rho} & \\ & A\otimes B & \end{array}$$

Corollary A.2.3. *In any monoidal category the morphisms* λ_1 *and* ρ_1 *are equal.*

Definition A.2.4 (Braided monoidal category)**.** *A braided monoidal category is a monoidal category* $(\mathbb{C}, \otimes, 1)$ *equipped with a braiding, that is a natural isomorphism* $\gamma_{A,B} : A \otimes B \to B \otimes A$*. Such that, for all* A, B *and* C*, the hexagonal diagrams commutes:*



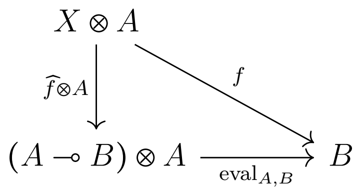


Definition A.2.5 (Symmetric monoidal category)**.** *A symmetric monoidal category is a braided monoidal category* $(\mathbb{C}, \otimes, 1, \gamma)$ *such that for all* A *and* B, $\gamma^{-1}_{A,B} = \gamma_{B,A}$*. A symmetric monoidal category is said to be strict if* γ *is the identity morphism.*

Definition A.2.6 (Monoidal closed category)**.** *A monoidal (right) closed category is a monoidal category equipped with a bifunctor* $\multimap : \mathbb{C}^{op} \times \mathbb{C} \to \mathbb{C}$ *such that, for all* A, $- \otimes A$ *is left adjoint to* $A \multimap -$*. In other words:*

$$\mathrm{Hom}_{\mathbb{C}}(B \otimes A, C) \simeq \mathrm{Hom}_{\mathbb{C}}(B, A \multimap C)$$

Equivalently, we can define a monoidal closed category as a monoidal category such that for every pair of objects A *and* B *there exists an object denoted* $A \multimap B$ *and a morphism* $\mathrm{eval}_{A,B} : (A \multimap B) \otimes A \to B$*, the (right) evaluation morphism, that satisfy the following condition: for every morphism* $f : X \otimes A \to B$*, there exists a morphism* $\widehat{f} : X \to A \multimap B$ *called the (right) transpose of* f*, such that the following diagram commutes*



Dualy, for every morphism of the shape $f : X \to A \multimap B$*, there exists a morphism* $\widetilde{f} : X \otimes A \to B$ *such that* $\widehat{\widetilde{f}} = f$*. The definition of a left closed monoidal category is obtained in the same way by requiring that the functor* $A \otimes -$ *admits a right adjoint. This is again equivalent to requiring the existence of a left transpose and left evaluation morphism that satisfy a similar triangle diagram.*

Proposition A.2.7 (Symmetric monoidal closed category)**.** *If a symmetric monoidal category is closed then, by symmetry the functor $A \multimap -$ (given by the closed structure) is a right adjoint to the functor $A \otimes -$ hence, it is left and right closed. In those cases, we will often not distinguish between the left and right transpose and the left and right evaluation morphism, calling them simply transpose and evaluation morphism.*

Definition A.2.8. *Let $\mathbb{C}$ be a symmetric closed monoidal category, then for every pair of objects A and B let us define $\partial_{A,B} : A \to (A \multimap B) \multimap B$ as :*

$$\begin{array}{ccc} B & & (A \multimap B) \multimap B \\ {\scriptstyle \mathrm{eval}_{A,B}}\uparrow & \rightsquigarrow & \uparrow {\scriptstyle \widehat{\mathrm{eval}_{A,B}} \;=:\; \partial_{A,B}} \\ (A \multimap B) \otimes A & & A \end{array}$$

A.3 Monoidal functors

Definition A.3.1 (Lax monoidal functor)**.** *Let $(\mathbb{C}, \otimes, e)$ and $(\mathbb{D}, \bullet, u)$ be two monoidal categories. A lax monoidal functor from $(\mathbb{C}, \otimes, e)$ to $(\mathbb{D}, \bullet, u)$ is the data of a functor $F : \mathbb{C} \to \mathbb{D}$ and two natural transformations*

$$m^2_{A,B} : FA \bullet FB \to F(A \otimes B) \qquad m^0 : u \to Fe$$

Such that the following three diagrams commutes

$$\begin{array}{ccc} (FA \bullet FB) \bullet FC & \xrightarrow{\alpha^\bullet} & FA \bullet (FB \bullet FC) \\ \downarrow {\scriptstyle m \bullet FC} & & \downarrow {\scriptstyle FA \bullet m} \\ F(A \otimes B) \bullet FC & & FA \bullet F(B \otimes C) \\ \downarrow {\scriptstyle m} & & \downarrow {\scriptstyle m} \\ F((A \otimes B) \otimes C) & \xrightarrow{F\alpha^\otimes} & F(A \otimes (B \otimes C)) \end{array}$$

$$\begin{array}{ccc} FA \bullet u & \xrightarrow{\rho^\bullet} & FA \\ \downarrow {\scriptstyle FA \bullet m} & & \uparrow {\scriptstyle F\rho^\otimes} \\ FA \bullet Fe & \xrightarrow{m} & F(A \otimes e) \end{array} \qquad \begin{array}{ccc} u \bullet FA & \xrightarrow{\lambda^\bullet} & FA \\ \downarrow {\scriptstyle m \bullet FA} & & \uparrow {\scriptstyle F\lambda^\otimes} \\ Fe \bullet FA & \xrightarrow{m} & F(e \otimes A) \end{array}$$

m^2 and m^0 are called the mediating maps of F.

Definition A.3.2 (Colax monoidal functor)**.** *Let $(\mathbb{C}, \otimes, e)$ and $(\mathbb{D}, \bullet, u)$ be two monoidal categories. A colax monoidal functor from $(\mathbb{C}, \otimes, e)$ to $(\mathbb{D}, \bullet, u)$ is the data of a functor $F : \mathbb{C} \to \mathbb{D}$ and two natural transformations*

$$n^2_{A,B} : F(A \otimes B) \to FA \bullet FB \qquad n^0 : Fe \to u$$

Such that the following three diagrams commutes

$$
\begin{array}{ccc}
F((A \otimes B) \otimes C) & \xrightarrow{F\alpha^{\otimes}} & F(A \otimes (B \otimes C)) \\
\downarrow{\scriptstyle n} & & \downarrow{\scriptstyle n} \\
F(A \otimes B) \bullet FC & & FA \bullet F(B \otimes C) \\
\downarrow{\scriptstyle n \bullet FC} & & \downarrow{\scriptstyle FA \bullet n} \\
(FA \bullet FB) \bullet FC & \xrightarrow{\alpha^{\bullet}} & FA \bullet (FB \bullet FC)
\end{array}
$$

$$
\begin{array}{ccc}
F(A \otimes e) & \xrightarrow{F\rho^{\otimes}} & FA \\
\downarrow{\scriptstyle n} & & \uparrow{\scriptstyle \rho^{\bullet}} \\
FA \bullet Fe & \xrightarrow{FA \bullet n} & FA \bullet u
\end{array}
\qquad
\begin{array}{ccc}
F(e \otimes A) & \xrightarrow{F\lambda^{\otimes}} & FA \\
\downarrow{\scriptstyle n} & & \uparrow{\scriptstyle \lambda^{\bullet}} \\
Fe \bullet FA & \xrightarrow{n \bullet FA} & u \bullet FA
\end{array}
$$

n^2 and n^0 are called the mediating maps of F.

Definition A.3.3 (Strong monoidal functor)**.** *A strong monoidal functor or simply monoidal functor, is a lax monoidal functor whose mediating maps are isomorphisms. Equivalently, it's also a colax monoidal functor whose mediating maps are isomorphisms. A strict monoidal functor is a strong monoidal functor whose mediating maps are the intentity morphisms.*

Definition A.3.4 (Symmetric (co)lax monoidal functors)**.** *A lax monoidal functor $(F, m) : (\mathbb{C}, \otimes, e,) \to (\mathbb{D}, \bullet, u)$ between two symmetric monoidal categories is called symmetric when the following diagram commutes:*

$$
\begin{array}{ccc}
FA \bullet FB & \xrightarrow{\gamma^{\bullet}} & FB \bullet FA \\
\downarrow{\scriptstyle m} & & \downarrow{\scriptstyle m} \\
F(A \otimes B) & \xrightarrow{\gamma^{\otimes}} & F(B \otimes A)
\end{array}
$$

A colax monoidal functor $(F, n) : (\mathbb{C}, \otimes, e,) \to (\mathbb{D}, \bullet, u)$ between two symmetric monoidal categories is called symmetric when the following diagram commutes:

$$
\begin{array}{ccc}
F(A \otimes B) & \xrightarrow{\gamma^{\otimes}} & F(B \otimes A) \\
\downarrow{\scriptstyle n} & & \downarrow{\scriptstyle n} \\
FA \bullet FB & \xrightarrow{\gamma^{\bullet}} & FB \bullet FA
\end{array}
$$

Proposition A.3.5. *The composition of (co)lax monoidal functors is (co)lax monoidal. The composition of strong monoidal functors is strong monoidal. The composition of symmetric (co)lax monoidal functors is symmetric (co)lax monoidal.*

Proposition A.3.6. *Let $\mathbb{C}$ and $\mathbb{D}$ be cartesian categories and $F : \mathbb{C} \to \mathbb{D}$ a functor. Then, F lifts in a unique way to a symmetric colax monoidal functor $(F, n) : (\mathbb{C}, \times_{\mathbb{C}}, 1_{\mathbb{C}}) \to (\mathbb{D}, \times_{\mathbb{D}}, 1_{\mathbb{D}})$.*

A.4 Monoidal adjunctions

Definition A.4.1 (Monoidal natural transformation)**.** *A monoidal natural transformation between two lax monoidal functors $(F, m), (G, n) : (\mathbb{C}, \otimes, e,) \to (\mathbb{D}, \bullet, u)$ is a natural transformation $\eta : F \Rightarrow G$ such that the following diagrams commutes:*

$$\begin{array}{ccc} FA \bullet FB & \xrightarrow{\eta_A \bullet \eta_B} & GA \bullet GB \\ \downarrow{\scriptstyle m} & & \downarrow{\scriptstyle n} \\ F(A \otimes B) & \xrightarrow{\eta_{A \otimes B}} & G(A \otimes B) \end{array} \qquad \begin{array}{ccc} & u & \\ {\scriptstyle m}\swarrow & & \searrow{\scriptstyle n} \\ Fe & \xrightarrow{\eta_e} & Ge \end{array}$$

Similarly, a monoidal natural transformation between two colax monoidal functors $(F, m), (G, n) : (\mathbb{C}, \otimes, e,) \to (\mathbb{D}, \bullet, u)$ is a natural transformation $\eta : F \Rightarrow G$ such that the following diagrams commutes:

$$\begin{array}{ccc} F(A \otimes B) & \xrightarrow{\eta_{A \otimes B}} & G(A \otimes B) \\ \downarrow{\scriptstyle m} & & \downarrow{\scriptstyle n} \\ FA \bullet FB & \xrightarrow{\eta_A \bullet \eta_B} & GA \bullet GB \end{array} \qquad \begin{array}{ccc} Fe & \xrightarrow{\eta_e} & Ge \\ {\scriptstyle m}\searrow & & \swarrow{\scriptstyle n} \\ & u & \end{array}$$

Proposition A.4.2. *Let $\mathbb{C}$ and $\mathbb{D}$ be cartesian categories $F, G : \mathbb{C} \to \mathbb{D}$ two functors and $\eta : F \Rightarrow G$ a natural transformation. Then, η is a monoidal natural transformation between the colax monoidal functors F and G (see A.3.6).*

Definition A.4.3 (Monoidal adjunction)**.** *Let $(\mathbb{C}, \otimes, e)$ and $(\mathbb{D}, \bullet, u)$ be two monoidal categories and $(F_*, m), (F^*, n)$ be an adjoint pair of (co)lax monoidal functors*

$$(\mathbb{C}, \otimes, e) \underset{(F^*, n)}{\overset{(F_*, m)}{\rightleftarrows}} (\mathbb{D}, \bullet, u) \qquad \bot$$

This adjunction is said to be monoidal if the unit and counit are monoidal in the sense of definition A.4.1

Proposition A.4.4 (lax-colax duality)**.** *Let $(\mathbb{C}, \otimes, e)$ and $(\mathbb{D}, \bullet, u)$ be two monoidal categories and F_*, F^* be an adjoint pair of functors*

$$(\mathbb{C}, \otimes, e) \underset{F^*}{\overset{F_*}{\rightleftarrows}} (\mathbb{D}, \bullet, u) \qquad \bot$$

with unit $\eta : \mathbb{1}_{\mathbb{C}} \Rightarrow F^* F_*$ *and counit* $\varepsilon : F_* F^* \Rightarrow \mathbb{1}_{\mathbb{D}}$. *Then, every lax monoidal structure* p *on* F^* *gives rise to a colax monoidal structure* n *on* F_* *defined as follows:*

$$n^2 : F_*(A \otimes B) \xrightarrow{F_*\eta\otimes\eta} F_*(F^*F_*A \otimes F^*F_*B) \xrightarrow{F_*p} F_*F^*(F_*A \bullet F_*B) \xrightarrow{\varepsilon} F_*A \bullet F_*B$$

$$n^0 : F_*e \xrightarrow{F_*p} F_*F^*u \xrightarrow{\varepsilon} u$$

Dualy, every colax monoidal structure n *on* F_* *gives rise to a lax monoidal structure* p *on* F^* *defined as follows:*

$$p^2 : F^*A \otimes F^*B \xrightarrow{\eta} F^*F_*(F^*A \otimes F^*B) \xrightarrow{F^*n} F^*(F_*F^*A \bullet F_*F^*B) \xrightarrow{F^*\varepsilon\bullet\varepsilon} F^*(A \bullet B)$$

$$p^0 : e \xrightarrow{\eta} F^*F_*e \xrightarrow{F^*n} F^*u$$

The first transformation defines a function ϕ *from the lax monoidal structures on* F^* *to the colax monoidal structures on* F_*. *The second transformation defines a function as well, which we can prove to be the inverse of* ϕ.

Proposition A.4.5. *Let* $(\mathbb{C}, \otimes, e)$ *and* $(\mathbb{D}, \bullet, u)$ *be two monoidal categories and* F_*, F^* *be an adjoint pair of functors*

$$(\mathbb{C}, \otimes, e) \underset{F^*}{\overset{F_*}{\rightleftarrows}} (\mathbb{D}, \bullet, u) \qquad \perp$$

with unit $\eta : \mathbb{1}_{\mathbb{C}} \Rightarrow F^* F_*$ *and counit* $\varepsilon : F_* F^* \Rightarrow \mathbb{1}_{\mathbb{D}}$. *If* (F_*, m) *and* (F^*, p) *are lax monoidal functors then:*

- *the colax structure* $n := \phi^{-1}(p)$ *is a right inverse of* m *(meaning* $m^2 \circ n^2 = \mathrm{id}$ *and* $m^0 \circ n^0 = \mathrm{id}$*) if and only if* η *is monoidal.*
- *the colax structure* $n := \phi^{-1}(p)$ *is a left inverse of* m *(meaning* $n^2 \circ m^2 = \mathrm{id}$ *and* $n^0 \circ m^0 = \mathrm{id}$*) if and only if* ε *is monoidal.*

Theorem A.4.6. *Let* $(\mathbb{C}, \otimes, e)$ *and* $(\mathbb{D}, \bullet, u)$ *be two monoidal categories,* F_*, F^* *be an adjoint pair of functors*

$$(\mathbb{C}, \otimes, e) \underset{F^*}{\overset{F_*}{\rightleftarrows}} (\mathbb{D}, \bullet, u) \qquad \perp$$

and (F_*, m) *a lax monoidal functor then, the adjuction* $F_* \dashv F^*$ *lifts to a monoidal adjuction* $(F_*, m) \dashv (F^*, p)$ *if and only if* (F_*, m) *is strong monoidal. In that case, the lax monoidal structure* p *on* F^* *is given by* $\phi(m^{-1})$*. In particular, if* $(F_*, m) \dashv (F^*, p)$ *is a monoidal adjunction between lax functors then,* (F_*, m) *is strong monoidal.*

Definition A.4.7 (Symmetric monoidal adjunction)**.** *A Symmetric monoidal adjunction is a monoidal adjunction in which the two categories* $(\mathbb{C}, \otimes, e)$ *and* $(\mathbb{D}, \bullet, u)$ *are symmetric monoidal and the functors* (F_*, m) *and* (F^*, p) *are symmetric monoidal.*

Proposition A.4.8. *Every adjunction between two cartesian categories lifts in a unique way to a monoidal symmetric adjunction between colax symmetric monoidal functors. This result is a corollary of A.4.2 and A.3.6*

Theorem A.4.9. *Let* $(\mathbb{C}, \otimes, e)$ *and* $(\mathbb{D}, \bullet, u)$ *be two symmetric monoidal categories,* F_*, F^* *be an adjoint pair of functors*

$$(\mathbb{C}, \otimes, e) \underset{F^*}{\overset{F_*}{\rightleftarrows}} (\mathbb{D}, \bullet, u) \qquad (\bot)$$

and (F_*, m) *a lax symmetric monoidal functor then, the adjuction* $F_* \dashv F^*$ *lifts to a symmetric monoidal adjuction* $(F_*, m) \dashv (F^*, p)$ *if and only if* (F_*, m) *is strong symmetric monoidal. In that case, the lax symmetric monoidal structure* p *on* F^* *is given by* $\phi(m^{-1})$*. In particular, if* $(F_*, m) \dashv (F^*, p)$ *is a symmetric monoidal adjunction between lax symmetric functors then,* (F_*, m) *is strong symmetric monoidal.*

The theorems A.4.6 and A.4.9 can be adapted to the dual case where a colax structure p is given on F^*.