There Is Only One Notion of Differentiation*

J. Robin B. Cockett[1] and Jean-Simon Lemay[2]

1 **Department of Computer Science, University of Calgary, Calgary, AB, Canada**
robin@ucalgary.ca

2 **Department of Mathematics and Statistics, University of Calgary, Calgary, AB, Canada**
jeansimon.lemay@ucalgary.ca

Abstract

Differential linear logic was introduced as a syntactic proof-theoretic approach to the analysis of differential calculus. Differential categories were subsequently introduce to provide a categorical model theory for differential linear logic. Differential categories used two different approaches for defining differentiation abstractly: a deriving transformation and a coderiliction. While it was thought that these notions could give rise to distinct notions of differentiation, we show here that these notions, in the presence of a monoidal coalgebra modality, are completely equivalent.

1998 ACM Subject Classification F.4.1 Mathematical Logic

Keywords and phrases Differential Categories, Linear Logic, Coalgebra Modalities, and Bialgebra Modalities

Digital Object Identifier 10.4230/LIPIcs.FSCD.2017.13

1 Introduction

Differential linear logic [5, 6] was introduced as a syntactic proof-theoretic approach to the analysis of differential calculus. Differential categories [2] were subsequently introduced to provide a categorical model theory for differential linear logic. In [2] two approaches to defining differentiation abstractly were introduced, these being based on a deriving transformation and a coderiliction. Fiore in [7] proposed an alternative approach to differentiation using what he called a *creation operator* – this was an operator having the same type as a deriving transformation but with a rather different axiomatization. He argued that, in the presence of a monoidal coalgebra modality and biproducts, his notion of differentiation provided a substantially better theory. However, as will be shown below, the notion of a creation operator is actually completely equivalent to that of a deriving transformation (with the interchange rule) as presented in [2]. Furthermore, these are, in that setting, equivalent to having a codereliction.

To show these notions are all equivalent it is necessary to explore, in some detail, the relation between a codereliction – essentially Fiore's "creation *map*" – and a deriving transformation (essentially Fiore's "creation *operator*"). It is straightforward to show that having a codereliction always implies the presence of a deriving transformation. The reverse, however, is more difficult and this is the main technical accomplishment of this paper. Its consequence is that contrary to the suggestion put forward in [7] there is only one notion of differentiation in linear logic.

* This work was partially supported by NSERC, Canada.

2nd International Conference on Formal Structures for Computation and Deduction (FSCD 2017).
Editor: Dale Miller; Article No. 13; pp. 13:1–13:21
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

First we should pause to consider Fiore's argument that one may as well work in a setting with biproducts. He argued that, as one can always complete an additive category to a category with biproducts, one may as well assume biproducts are present. Of course, when one considers differentiation, one also needs to be able to extend the coalgebra modality. It is standard that one can extend a *monoidal* coalgebra modality to the biproduct completion using the Seely isomorphisms. However, when the modality is *not* monoidal, there is apparently no reason why it should extend. Significantly, the deriving transformation assumed in [2] did not assume the coalgebra modality was monoidal. Thus, if one is to entertain these more basic modalities, one must be cautious about using Fiore's argument. Here we certainly take such modalities seriously.

The paper starts by reminding the reader of the "original" definition of a differential category. This relies on the idea of a coalgebra modality and a deriving transformation. Familiarity with linear logic may tempt one to think that this is the "exponential" modality of linear logic but it is not. It is a strictly weaker notion as the modality is not assumed monoidal. There are many familiar examples of differential categories based on a monoidal coalgebra modality [2] – such as (the opposite of) the free symmetric algebra monad on vector spaces. The reader may wonder, however, whether there are any significant examples of differential categories based a mere coalgebra modality. Two compelling examples – by no means the only ones – are given by smooth functions via the free $\mathcal{C}^\infty$-ring over vector spaces (as mentioned in [2]) and the free Rota-Baxter algebra over modules (for more details on this monad see [13]).

Here we refer to additive categories, with a monoidal coalgebra modality, as "additive linear categories": their biproduct completion is then an "additive monoidal storage category" [4]. This latter was the setting being considered by Fiore in [7]. An additive monoidal storage category (which has biproducts) is always an additive linear category (which may not have biproducts) and both always have a canonical monoidal bialgebra modality. Here, to facilitate the proofs, it is convenient to work with a further intermediate notion we called an "additive bialgebra modality": this is a bialgebra modality which has an additional coherence requirement between the additive and bialgebra structure. It is with respect to this structure that we prove that deriving transformations and coderelictions are equivalent.

Coderelictions always give deriving transformations, and it was known [2] that a deriving transformation, for a bialgebra modality, is equivalent to a codereliction if and only if the deriving transformation satisfies the ∇-rule. However, this latter rule was thought to be a completely independent requirement. The key new observation is that, for an additive bialgebra modality with a deriving transformation the ∇-rule is, in fact, implied. More specifically, while a deriving transformation is assumed to satisfy five rules – **[d.1]**–**[d.5]** below – which include the linear rule and the Leibniz rule, over an additive bialgebra modality, we prove that, for a deriving transformation which satisfies just the linear rule, the ∇-rule is equivalent to the Leibniz rule.

When an additive symmetric monoidal category has a monoidal coalgebra modality it is straightforward to show that the modality is an additive bialgebra modality. Thus, for additive linear categories deriving transformations and coderelictions are also equivalent.

Clearly additive bialgebra modalities and monoidal coalgebra modalities are closely related. In particular, additive bialgebra modalities can always be extended to the biproduct completion (see Appendix C) and, furthermore, this biproduct completion has Seely isomorphisms [11]. Thus, additive bialgebra modalities correspond to monoidal storage categories [4] (also called new Seely categories [1, 10]) having the Seely isomorphisms. However, it is well-known (as the name suggests) that monoidal storage categories have a monoidal coalgebra modality.

Thus, additive bialgebra modalities are, in fact, monoidal coalgebra modalities.

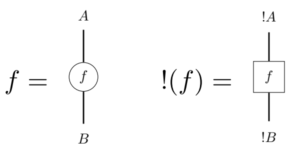
To complete the main story of the paper, we observe that in an additive linear category coderelictions and deriving transformations always satisfy the "strength laws": these were the subject of an addendum added to [7]. We provide a proof in the setting of additive linear categories. Putting all this together one has that – contrary to the suggestion in [7] – deriving transformations and creation operators are, in additive linear categories, completely equivalent.

2 Conventions and the Graphical Calculus

We shall use diagrammatic order for composition: explicitly, this means that the composite map fg is the map which first does f then g. Furthermore, to simplify working in symmetric monoidal categories, we will allow ourselves to work in strict symmetric monoidal categories and so will generally suppress the associator and unitor isomorphisms. For a symmetric monoidal category we will use $\otimes$ for the tensor product, K for the unit, and $\sigma : A \otimes B \to B \otimes A$ for the symmetry isomorphism.

We shall make extensive use of the graphical calculus [8] for symmetric monoidal categories as this makes proofs easier to follow. We refer the reader to [12] for an introduction to the graphical calculus in monoidal categories and its variations. Note, however, that our diagrams are to be read down the page – from top to bottom – and we shall often omit labelling wires with objects.

We will be working with coalgebra modalities: these are based on a comonad $(!, \delta, \varepsilon)$ where ! is the functor, δ is the comultiplicaton and ε is the counit. As in [2], we will use functor boxes when dealing with string diagrams involving the functor !: a mere map $f : A \to B$ will be encased in a circle while $!(f) : !A \to !B$ will be encased in a box:



3 Differential Categories

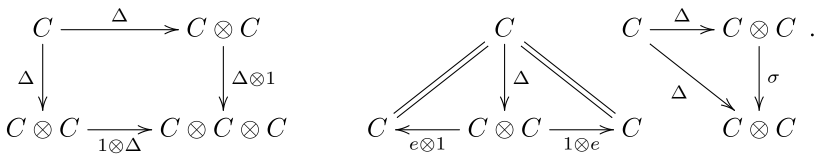
Tensor differential categories are structures over additive symmetric monoidal categories with a coalgebra modality. We begin by recalling the components of this structure starting with the notion of an additive category. Here we mean "additive" in the sense of being commutative monoid enriched: we do not assume negatives nor do we assume biproducts (this differs from the usage in [9] for example). This allows many important examples such as the category of sets and relation or the category of modules for a commutative rig[1].

▶ **Definition 1.** An **additive category** is a commutative monoid enriched category, that is, a category in which each hom-set is a commutative monoid with an addition operation $+$ and a zero 0, and such that composition preserves the additive structure, that is: $k(f+g)h = kfh+kgh$, $0f = 0$ and $f0 = 0$. An **additive symmetric monoidal category** is a symmetric monoidal category which is also an additive category in which the tensor product is compatible

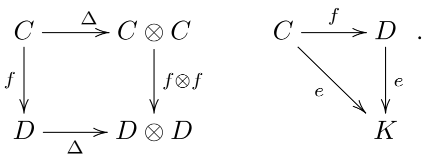
[1] Rigs are also known as a semirings: they are ri*n*gs without *n*egatives.

with the additive structure in the sense that $k \otimes (f+g) \otimes h = k \otimes f \otimes h + k \otimes g \otimes h$, $0 \otimes h = 0$, and $h \otimes 0 = 0$.

In a symmetric monoidal category, a cocommutative comonoid is a triple (C, Δ, e) consisting of an object C, a map $\Delta : C \to C \otimes C$ called the comultiplication and a map $e : C \to K$ called the counit such that the following diagrams commute:



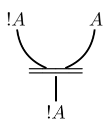
A morphism of comonoids $f : (C, \Delta, e) \to (D, \Delta', e')$ is a map $f : C \to D$ which preserves the comultiplication and counit, that is, the following diagrams commute:



▶ **Definition 2.** A **coalgebra modality** [2] on a symmetric monoidal category is a quintuple $(!, \delta, \varepsilon, \Delta, e)$ consisting of a comonad $(!, \delta, \varepsilon)$, a natural transformation Δ with components $\Delta_A : !A \to !A \otimes !A$, and a natural transformation e with components $e_A : !A \to K$ such that for each object A, $(!A, \Delta_A, e_A)$ is a cocommutative comonoid and δ preserves the comultiplication, that is, $\delta\Delta = \Delta(\delta \otimes \delta)$.

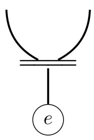
One can prove that δ also preserves the counit, $\delta e = e$, and so δ is actually a comonoid homomorphism. Furthermore, requiring that Δ and e be natural transformations is equivalent to asking that for each map $f : A \to B$, $!(f) : !A \to !B$ is a comonoid morphism. When combined with the additive structure, this ensures that $!A$ is a coalgebra in the classical algebraic sense. Note that, for now, we do not assume that the functor ! of a coalgebra modality is a monoidal functor (this will come later). The coKleisli maps for the comonad are important: these maps are of the form $f : !A \to B$, which amongst these are the **linear maps** $\varepsilon g : !A \to B$ where $g : A \to B$.

▶ **Definition 3.** A **differential category** is an additive symmetric monoidal category with a coalgebra modality which comes equipped with a **deriving transformation** [2], that is, a natural transformation d with components $\mathsf{d}_A : !A \otimes A \to !A$, which is represented in the graphical calculus as:

$\mathsf{d} :=$ 

such that d satisfies the following equations:

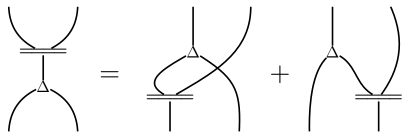
[d.1] Constant Rule: $\mathsf{d}e = 0$



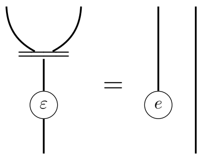
$= 0$

[d.2] Leibniz Rule:

$$\mathsf{d}\Delta = (\Delta \otimes 1)(1 \otimes \sigma)(\mathsf{d} \otimes 1) + (\Delta \otimes 1)(1 \otimes \mathsf{d})$$

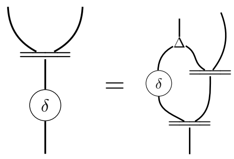


[d.3] Linear Rule: $\mathsf{d}\varepsilon = e \otimes 1$

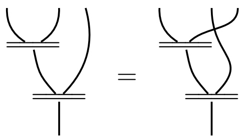


[d.4] Chain Rule:

$$\mathsf{d}\delta = (\Delta \otimes 1)(\delta \otimes 1 \otimes 1)(1 \otimes \mathsf{d})\mathsf{d}$$



[d.5] Interchange Rule: $(\mathsf{d} \otimes 1)\mathsf{d} = (1 \otimes \sigma)(\mathsf{d} \otimes 1)\mathsf{d}$



The first axiom **[d.1]** states that the derivative of a constant map is zero. The second axiom **[d.2]** is the Leibniz rule or the product rule for differentiation. The third axiom **[d.3]** says that the derivative of a linear map is constant. The fourth axiom **[d.4]** is the chain rule. The last axiom **[d.5]** is the interchange law, which naively states that differentiating with respect to x then y is the same as differentiation with respect to y then x. It should be noted that **[d.5]** was not originally a requirement in [2] but was later added to the definition to ensure that the coKleisli category of a differential category was a Cartesian differential category [3]. It should also be noted that by the naturality of e and d, the constant rule **[d.1]** is in fact derivable:

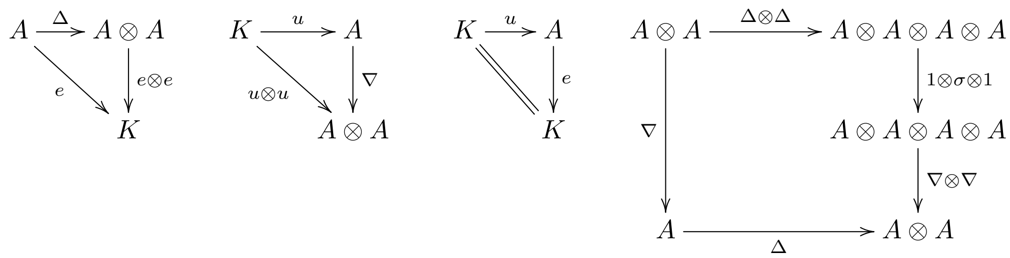
▶ **Lemma 4.** *For a coalgebra modality on additive symmetric monoidal category, any natural transformation $\mathsf{d}_A : !A \otimes A \to !A$ satisfies the constant rule **[d.1]**.*

Proof. By naturality and the additive structure: $\mathsf{d}e = \mathsf{d}!(0)e = (!(0) \otimes 0)\mathsf{d}e = 0\mathsf{d}e = 0$. ◀

Therefore a deriving transformation is simply a natural transformation which satisfies the Leibniz rule **[d.2]**, the linear rule **[d.3]**, the chain rule **[d.4]** and the interchange rule **[d.5]**.

4 Bialgebra Modalities, coderelictions and the ∇-Rule

In [2], Blute, Cockett and Seely observed that if the coalgebra modality came equipped with a bialgebra structure and a codereliction then one could obtain a deriving transformation. We recall these ideas, starting with the notion of a bialgebra modality. In a symmetric monoidal category, a (commutative) bialgebra is a quintuple $(A, \nabla, u, \Delta, e)$ such that (A, ∇, u) is a commutative monoid, (A, Δ, e) is a cocommutative comonoid, and the following diagrams commute:



▶ **Definition 5.** A **bialgebra modality** [2] on an additive symmetric monoidal category is a coalgebra modality $(!, \delta, \varepsilon, \Delta, e)$ equipped with a natural transformation $\nabla : !A \otimes !A \to !A$, and a natural transformation $u : K \to !A$ such that $(!A, \nabla, u, \Delta, e)$ is a bialgebra for each object A, and ε is compatible with ∇ in the following sense: $\nabla\varepsilon = \varepsilon \otimes e + e \otimes \varepsilon$.

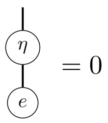
By the naturality of ∇, Δ, u and e we note that for every map f, $!(f)$ is both a monoid and comonoid morphism. Also, in the original definition of a bialgebra modality in [2], it was also required that ε be compatible with u, however this compatibility is provable by the naturality of u and ε. The proof is similar to that of Lemma 4.

▶ **Lemma 6.** *For a bialgebra modality* $u\varepsilon = 0$*.*

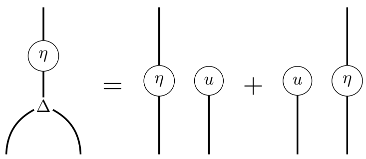
The key ingredient required to obtain a differential for a bialgebra modality is:

▶ **Definition 7.** A **codereliction** [2] for a bialgebra modality on an additive symmetric monoidal category is a natural transformation $\eta_A : A \to !A$, such that the following equalities hold:

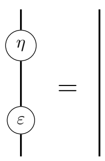
[dC.1] Constant Rule: $\eta e = 0$



[dC.2] Product Rule: $\eta\Delta = \eta \otimes u + u \otimes \eta$

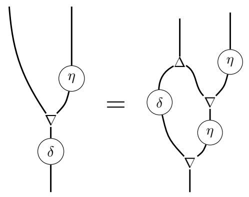


[dC.3] Linear Rule: $\eta\varepsilon = 1$



[dC.4] Chain Rule:

$$(1 \otimes \eta)\nabla\delta = (\Delta \otimes \eta)(1 \otimes \nabla)(\delta \otimes \eta)\nabla$$

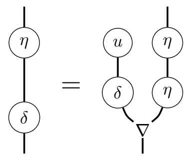


As for the constant rule for the deriving transformation, the constant rule [**dC.1**] for a codereliction can be derived (the proof is similar to that of Lemma 4):

▶ **Lemma 8.** *For a bialgebra modality, any natural transformation $\eta_A : A \to !A$ satisfies the constant rule [**dC.1**].*

Thus, a codereliction is simply a natural transformation which satisfies the product rule [**dC.2**], the linear rule [**dC.3**] and the chain rule [**dC.4**]. In [7], Fiore uses an alternative axiom for the chain rule [**dC.4**] which is stated as follows:

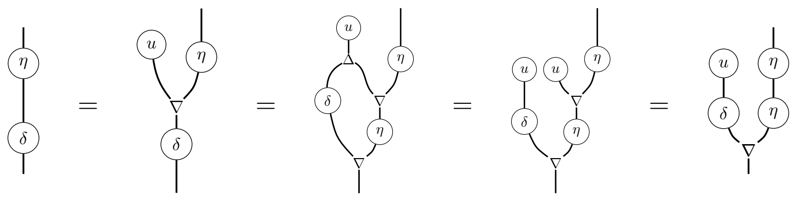
[dC.4'] Alternative Chain Rule: $\nabla\delta = (u \otimes \eta)(\delta \otimes \eta)\nabla$



Although it was not noted in [7], in that setting both [**dC.4**] and [**dC.4′**] are equivalent. In the setting of a mere bialgebra modality, it is clear that [**dC.4**] implies [**dC.4′**]: the reverse implication, however, does not appear to hold. Thus, at this stage we prove the implication in one direction:

▶ **Lemma 9.** *Any natural transformation which satisfies the chain rule [**dC.4**] also satisfies the alternative chain rule [**dC.4′**].*

Proof. The bialgebra structure gives the following chain of equalities:

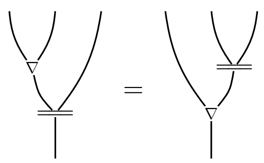


◀

Let us now consider the relation between deriving transformations and bialgebra modalities. This is captured by the ∇-rule [2]:

▶ **Definition 10.** For a bialgebra modality, a natural transformation $\mathsf{d}_A : !A \otimes A \to !A$ is said to satisfy the **∇-rule** if:

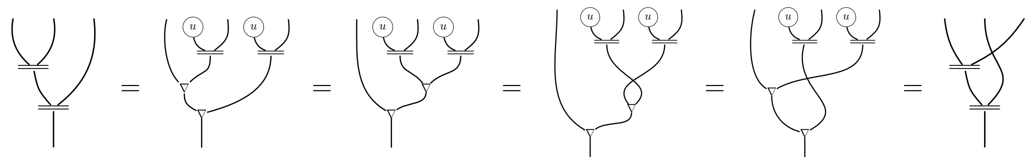
[d.∇] ∇-Rule: $(\nabla \otimes 1)\mathsf{d} = (1 \otimes \mathsf{d})\nabla$



Notice this implies that $\mathsf{d} = (1 \otimes u \otimes 1)(\nabla \otimes 1)\mathsf{d} = (1 \otimes u \otimes 1)(1 \otimes \mathsf{d})\nabla$. Thus, an immediate consequence of satisfying the ∇-rule is interchange rule:

▶ **Lemma 11.** *For a bialgebra modality, a natural transformation* d *which satisfies the* ∇*-rule,* **[d.∇]***, also satisfies the interchange rule,* **[d.5]***.*

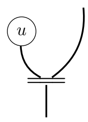
Proof. By **[d.∇]** and the commutative monoid structure, we have the following equality:



◀

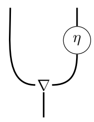
Therefore, for a bialgebra modality, a natural transformation which satisfies the Leibniz rule **[d.2]**, the linear rule **[d.3]**, the chain rule **[d.4]**, and **[d.∇]** is a deriving transformation. Furthermore, all deriving transformations which satisfy the ∇-rule **[d.∇]** induce a codereliction defined as:

$$eta := \; A \xrightarrow{u \otimes 1} !A \otimes A \xrightarrow{\mathsf{d}} !A \qquad \eta :=$$



Conversely, every codereliction induces a deriving transformation which satisfies the ∇-rule:

$$\mathsf{d} := \; !A \otimes A \xrightarrow{1 \otimes \eta} !A \otimes !A \xrightarrow{\nabla} !A \qquad \mathsf{d} :=$$



Using the same proof in [2] – which was for monoidal storage categories – it is easily seen that:

▶ **Theorem 12.** *For an additive symmetric monoidal category with a bialgebra modality, deriving transformations, which satisfy the* ∇*-rule* **[d.∇]***, are in bijective correspondence to coderelictions by:*

$$\mathsf{d} \longmapsto \eta := (u \otimes 1)\mathsf{d}\,, \qquad \eta \longmapsto \mathsf{d} := (1 \otimes \eta)\nabla\,.$$

5 Additive Bialgebra Modalities

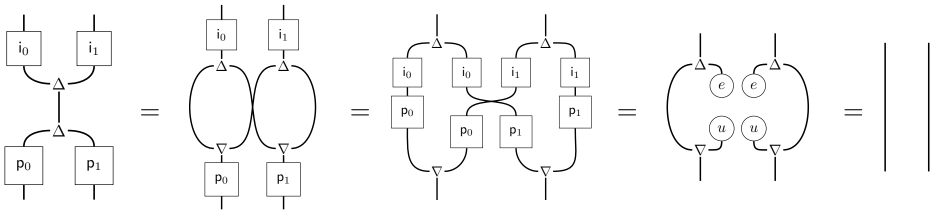
In this section we introduce the concept of an *additive* bialgebra modality. Our goal will be to show that a deriving transformation for an additive bialgebra modality induces a codereliction. Since a codereliction always implies a deriving transformation, this shows that, for an additive bialgebra modality, deriving transformations and coderiction maps are equivalent. A source of examples of additive bialgebra modalities will be derived in the next section. Additive bialgebra modalities require additional coherence with the additive structure:

▶ **Definition 13.** An **additive bialgebra modality** on an additive symmetric monoidal category is a bialgebra modality which is compatible with the additive structure in the sense that $!(0) = eu$ and $!(f+g) = \Delta(!(f) \otimes !(g))\nabla$.

First we examine coderelictions for additive bialgebra modalities. When a natural transformations η is a section of ε, that is, η satisfies [**dC.3**], we can define four natural transformations: $\mathsf{p}_0 = \varepsilon \otimes e$, $\mathsf{p}_1 = e \otimes \varepsilon$, $\mathsf{i}_0 = \eta \otimes u$, $\mathsf{i}_1 = u \otimes \eta$. Notice that as η satisfies the constant rule [**dC.1**] and the linear rule [**dC.3**] and from the properties of a bialgebra modality that we have: $\mathsf{i}_j\mathsf{p}_k = 0$ when $j \neq k$ and $\mathsf{i}_j\mathsf{p}_j = 1$, which is reminiscent of the identities satisfied by the projection and injection maps of a biproduct. These maps will be key to the proof of Lemma 15 below. For an additive bialgebra modality this means that we have: $!(\mathsf{i}_j)!(\mathsf{p}_k) = eu$ when $j \neq k$ and $!(\mathsf{i}_j)!(\mathsf{p}_j) = 1$. This allows the derivation of the following useful identity:

▶ **Lemma 14.** *For an additive bialgebra modality and a natural transformations η which satisfies the linear rule [**dC.3**]:* $(!(\mathsf{i}_0) \otimes !(\mathsf{i}_1))\nabla\Delta(!(\mathsf{p}_0) \otimes !(\mathsf{p}_1)) = 1$.

Proof. By the bialgebra modality, we have the following equality:



◀

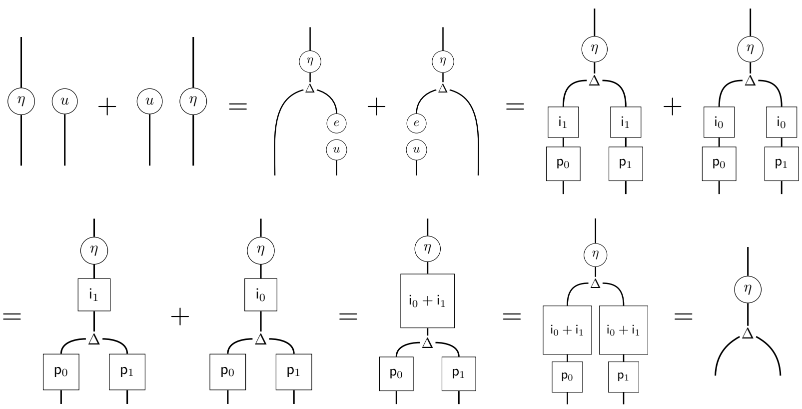
Furthermore, for an additive bialgebra modality, the linear rule [**dC.3**] implies the product rule [**dC.2**]:

▶ **Lemma 15.** *For an additive bialgebra modality, any natural transformation, η, which satisfies the linear rule, [**dC.3**], also satisfies the product rule, [**dC.2**].*

Proof. Notice that by naturality of η, we have that:

$$\eta!(f+g) = (f+g)\eta = f\eta + g\eta = \eta!(f) + \eta!(g) .$$

Then, using i_j and p_k, we have the following:



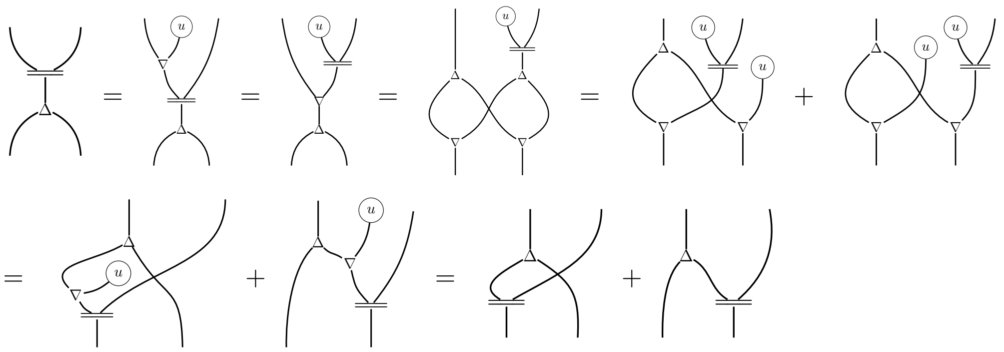
◀

Therefore, for an additive bialgebra modality, a natural transformation which simply satisfies the linear rule [**dC.3**] and the chain rule [**dC.4**] is a codereliction.

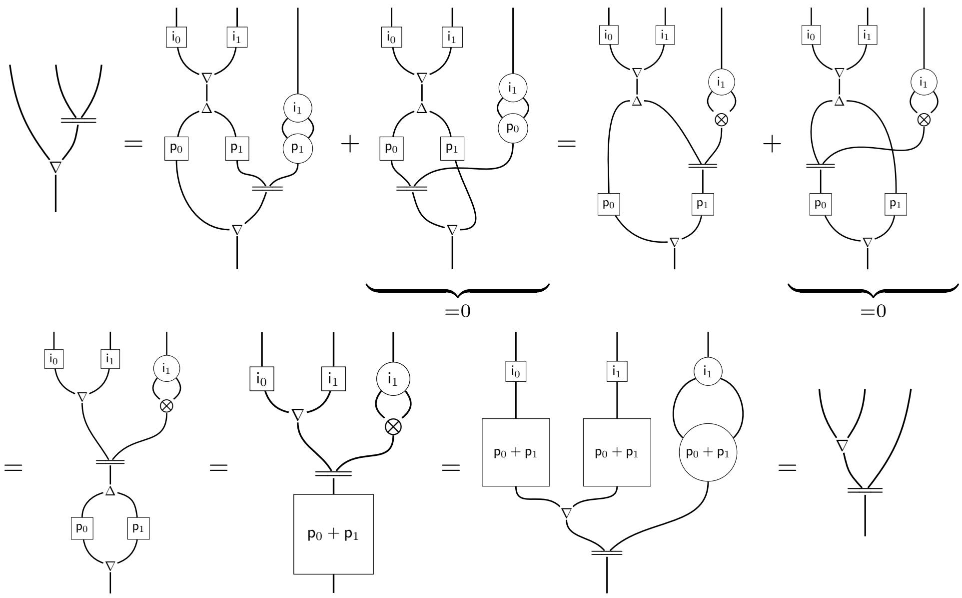
Turning our attention to deriving transformations for additive bialgebra modalities, we begin by noticing that satisfying the Leibniz rule is equivalent to satisfying the ∇-rule:

▶ **Theorem 16.** *For an additive bialgebra modality and a natural transformation* d *satisfying the linear rule* **[d.3]***, the Leibniz rule* **[d.2]** *is satisfied if and only if the* ∇*-rule* **[d.**∇**]** *is satisfied.*

Proof.
[d.∇**]** $\Rightarrow$ **[d.2]:** It is easy to see that since d satisfies [**d.3**] that $(u \otimes 1)\mathsf{d}$ satisfies [**dC.3**], the linear rule for coderelictions. However, by Lemma 15, this implies that $(u \otimes 1)\mathsf{d}$ satisfies [**dC.2**], the product rule for coderelictions. Therefore, we have:



[d.2] $\Rightarrow$ **[d.**∇**]:** By the properties of i_j and p_k, and the identity of Lemma 14, we have:



◀

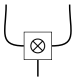
Therefore, we obtain the following theorem:

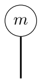
▶ **Theorem 17.** *For an additive bialgebra modality, every deriving transformation satisfies the* ∇*-rule* **[d.**∇**]***, thus is induced equivalently by a codereliction.*

6 Additive Linear Categories

We now turn our attention to the case when the coalgebra modality is monoidal. Recall that a symmetric monoidal functor [9] is a functor ! equipped with a natural transformation

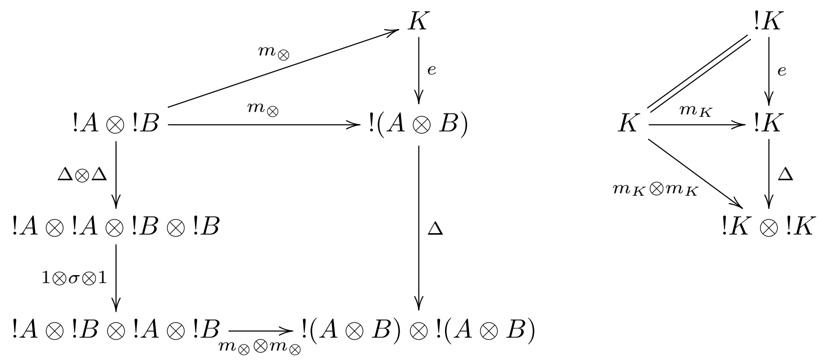
$m_\otimes : !A \otimes !B \to !(A \otimes B)$ and a map $m_K : K \to !K$ satisfying certain coherences. This can be extended to defining a symmetric monoidal comonad by asking that δ and ε be monoidal natural transformations. For a full detailed list of the coherences for symmetric monoidal comonads, see [1]. The string diagrams representations of $m_\otimes$ and m_K are as follows:

$m_\otimes =$ 

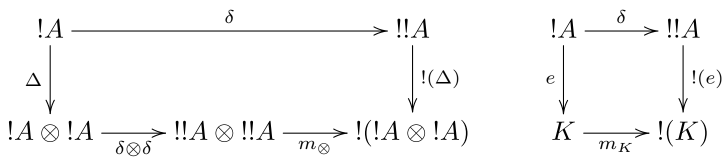
$m_K =$ 

▶ **Definition 18.** A **monoidal coalgebra modality** on a symmetric monoidal category is a symmetric monoidal comonad, $(!, \delta, \varepsilon, m_\otimes, m_K)$, and a coalgebra modality, $(!, \delta, \varepsilon, \Delta, e)$, satisfying

(i) Δ and e are monoidal transformations, that is, the following diagrams commute:



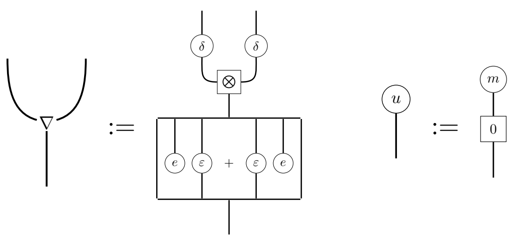
(ii) Δ and e are !-coalgebra morphisms, that is, the following diagrams commute:



A **linear category** [1, 4] is a symmetric monoidal category with a monoidal coalgebra modality; an **additive linear category** is a linear category which is also an additive symmetric monoidal category.

A monoidal coalgebra modality on an additive linear category induces an additive bialgebra modality. The multiplication and unit are respectively:

$$!A \otimes !A \xrightarrow{\delta \otimes \delta} !!A \otimes !!A \xrightarrow{m_\otimes} !(!A \otimes !A) \xrightarrow{!(\varepsilon \otimes e + e \otimes \varepsilon)} !A \qquad K \xrightarrow{m_K} !K \xrightarrow{!(0)} !A$$

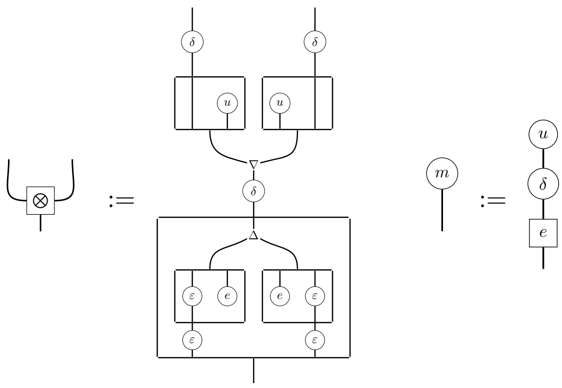


The converse statement is also, in fact, true: an additive bialgebra modality induces a monoidal coalgebra modality. The monoidal structure $m_\otimes$ and m_K are defined respectively as:

$$\begin{array}{ccccccc} !A\otimes !B & \xrightarrow{\delta\otimes\delta} & !A\otimes !B & \xrightarrow{!(1\otimes u)\otimes !(u\otimes 1)} & !(!A\otimes !B)\otimes !(!A\otimes !B) & \xrightarrow{\nabla} & !(!A\otimes !B) \\ {\scriptstyle m_\otimes}\downarrow & := & & & & & \downarrow{\scriptstyle \delta} \\ !(A\otimes B) & \xleftarrow[!(\varepsilon\otimes\varepsilon)]{} & !(!A\otimes !B) & \xleftarrow[!(!(\varepsilon\otimes e)\otimes !(e\otimes\varepsilon))]{} & !(!(!A\otimes !B)\otimes !(!A\otimes !B)) & \xleftarrow[!(\Delta)]{} & !!(!A\otimes !B) \end{array}$$

$$\begin{array}{ccc} K & \xrightarrow{u} & !K \\ {\scriptstyle m_K}\downarrow & := & \downarrow{\scriptstyle \delta} \\ !(K) & \xleftarrow[!(e)]{} & !!K \end{array}$$

Represented in string diagrams as:



▶ **Theorem 19.** *For an additive symmetric monoidal category, monoidal coalgebra modalities correspond bijectively to additive bialgebra modalities.*

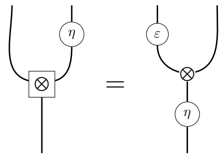
An indirect proof of this is discussed at the end of the introduction: a direct proof is long and quite technical. The identities expressing compatibility between the bialgebra modality and the monoidal coalgebra which were observed in [7] (Theorem 3.1) to hold in an additive monoidal storage category also hold in an additive linear category (see Appendix A for a proof):

▶ **Lemma 20.** *In an additive linear category, the following diagrams commute:*

$$\begin{array}{ccc} !A & \xrightarrow{1\otimes u} & !A\otimes !B \\ {\scriptstyle e}\downarrow & & \downarrow{\scriptstyle m_\otimes} \\ K & \xrightarrow[u]{} & !(A\otimes B) \end{array} \qquad \begin{array}{ccc} !A\otimes !A & \xrightarrow{\nabla} & !A \\ {\scriptstyle \delta\otimes\delta}\downarrow & & \downarrow{\scriptstyle \delta} \\ !!A\otimes !!A & & !!A \\ {\scriptstyle m_\otimes}\downarrow & \nearrow{\scriptstyle !(\nabla)} & \\ !(!A\otimes !A) & & \end{array} \qquad \begin{array}{ccc} !A\otimes !B\otimes !B & \xrightarrow{1\otimes\nabla} & !A\otimes B \\ {\scriptstyle \Delta\otimes 1\otimes 1}\downarrow & & \\ !A\otimes !A\otimes !B\otimes !B & & \downarrow{\scriptstyle m_\otimes} \\ {\scriptstyle 1\otimes\sigma\otimes 1}\downarrow & & \\ !A\otimes !B\otimes !A\otimes !B & & !(A\otimes B) \\ {\scriptstyle m_\otimes\otimes m_\otimes}\downarrow & \nearrow{\scriptstyle \nabla} & \\ !(A\otimes B)\otimes !(A\otimes B) & & \end{array}$$

We now turn our attention to the relation between the monoidal structure and the differential structure. In [7] Fiore introduced another axiom relating η to the monoidal structure:

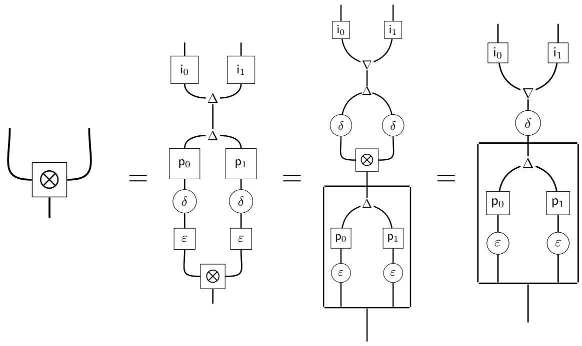
[dC.m] Monoidal Rule: $(1 \otimes \eta)m_\otimes = (\varepsilon \otimes 1)\eta$



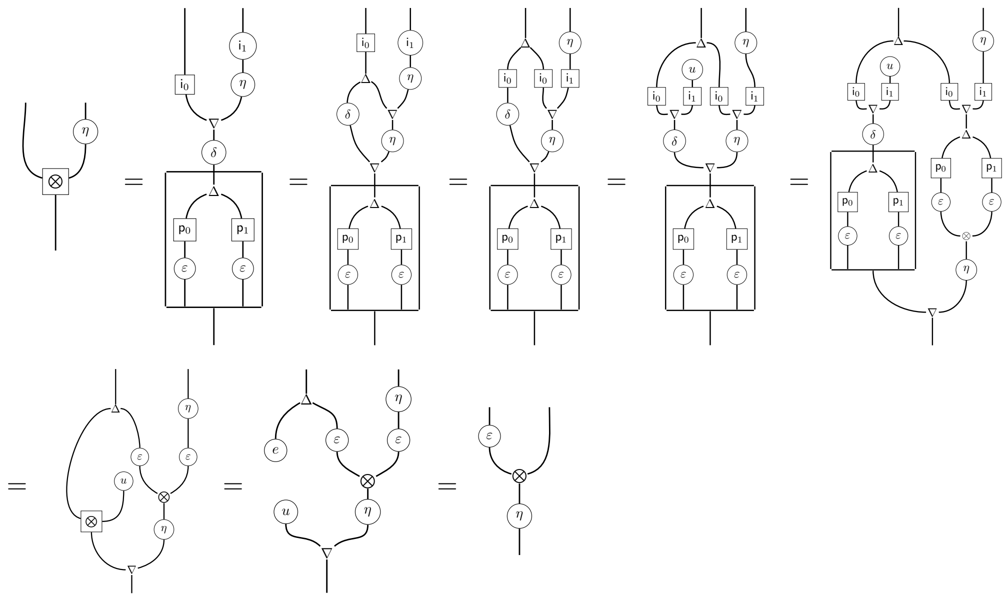
It turns out that coderelictions for the induced additive bialgebra modality always satisfy the monoidal rule [**dC.m**]:

▶ **Lemma 21.** *In an additive linear category, coderelictions on the induced additive bialgabra modality satisfy the monoidal rule [**dC.m**].*

Proof. By Lemma 14 and the fact that Δ is a !-coalgebra morphism, we first have that:



where i_j and p_k are defined as in the previous section. Expressing $m_\otimes$ as above, then by the linear rule [**dC.3**], chain rule [**dC.4**], the naturality of u and ∇, and the first diagram of Lemma 20 we have the following equality :

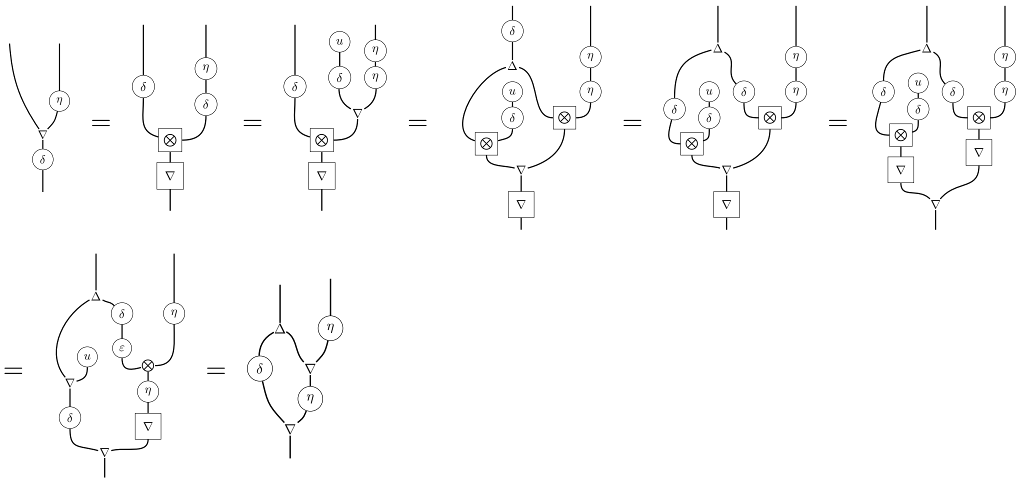


◀

Conversely, the alternative chain rule [**dC.4′**] and the monoidal rule [**dC.m**] imply the chain rule [**dC.4**].

▶ **Lemma 22.** *In an additive linear category, a natural transformation which satisfies the alternative chain rule* **[dC.4′]** *and the monoidal rule* **[dC.m]***, on the the induced additive bialgebra modality, also satisfies the chain rule* **[dC.4]***.*

Proof. By the compatiblity relations of Lemma 20 and the coalgebra modality requirements we have:



◀

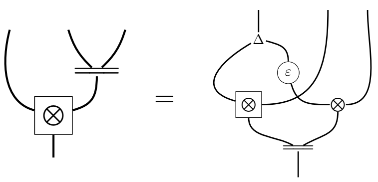
▶ **Corollary 23.** *For an additive linear category the following are equivalent for a natural transformation η and the induced additive bialgebra modality:*

(i) *η is a codereliction;*
(ii) *η satisfies the linear rule* **[dC.3]** *and the chain rule* **[dC.4]***;*
(iii) *η satisfies the linear rule* **[dC.3]***, the alternative chain rule* **[dC.4′]** *and the monoidal rule* **[dC.m]***.*

Part (iii) is the definition of Fiore's creation map: this shows that the original definition of a codereliction is equivalent to Fiore's creation map.

Finally we explore deriving transformations of additive linear categories. The compatibility between a deriving transformation and the monoidal structure is described by the monoidal rule – this is the strength rule which was the subject of Fiore's addendum:

[d.m] Monoidal Rule: $(1 \otimes \mathsf{d})m_\otimes = (\Delta \otimes 1 \otimes 1)(1 \otimes \varepsilon \otimes 1 \otimes 1)(1 \otimes \sigma \otimes 1)(m_\otimes \otimes 1 \otimes 1)\mathsf{d}$



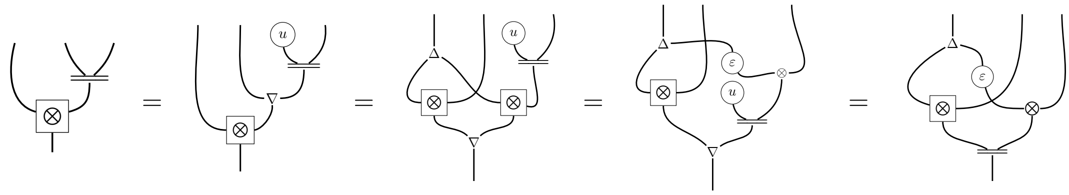
Fiore's creation operator [7] was defined to satisfy the linear rule **[d.3]**, the chain rule **[d.4]**, the ∇-rule **[d.∇]**, and the monoidal rule **[d.m]** – in his addendum, he pointed out the latter was redundant. It turns out that when a natural transformation satisfies both the linear rule **[d.3]** and the chain rule **[d.4]**, then the monoidal rule is equivalent to both the ∇-rule and the Leibniz rule:

▶ **Theorem 24.** *For the induced additive bialgebra modality of an additive linear category and a natural transformation satisfying the linear rule* **[d.3]** *and the chain rule* **[d.4]***, the following are equivalent:*

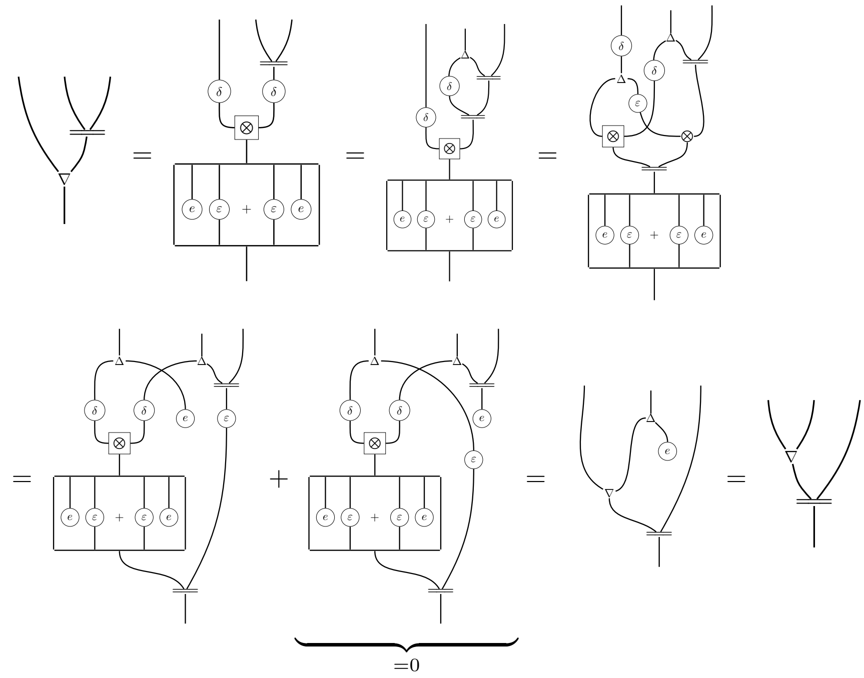
(i) *the Leibniz rule* ***[d.2]****;*
(ii) *the* ∇*-rule* ***[d.∇]****;*
(iii) *the monoidal rule* ***[d.m]***

Proof. Since this is an extension of Lemma 16, it suffices to show that the ∇-rule [**d.∇**] and the monoidal rule [**d.m**] are equivalent.

[d.∇] $\Rightarrow$ [d.m]: It is easy to see that since d satisfies the linear rule [**d.3**] and the chain rule [**d.4**], $(u \otimes 1)\mathsf{d}$ satisfies the codereliction linear rule [**dC.3**] and chain rule [**dC.4**], and therefore by Corollary 23 is a codereliction and which by Lemma 21 satisfies the codereliction monoidal rule [**dC.m**]. Therefore, by the compatibility relations of Lemma 20, we have:



[d.m] $\Rightarrow$ [d.∇]: By expanding ∇ using $m_\otimes$, we obtain the following equality:



◀

Finally, this gives the following theorem:

▶ **Theorem 25.** *For the induced additive bialgebra modality of an additive linear category, all deriving transformations satisfy the monoidal rule* ***[d.m]*** *and are induced by a codereliction.*

7 Conclusion

Of course it would have been very much simpler if all these observations had been made in [2], the original paper. Unfortunately, they were not. The suggestion in Marcelo Fiore's paper [7] that something new and different had been found, made us realize that the relation between deriving transformations and coderelictions had never been fully explored. Here we have revisited this relation. Significantly, it is not that the notion of differentiation varies but rather, as the setting varies, significantly different presentations of the differential become possible. Philosophically this is certainly how thing should be ... and apparently it is how they are!

References

1 Gavin Bierman. What is a categorical model of intuitionistic linear logic? *Typed Lambda Calculi and Applications*, pages 78–93, 1995.

2 Richard F. Blute, J. Robin B. Cockett, and Robert A. G. Seely. Differential categories. *Mathematical structures in computer science*, 16(06):1049–1083, 2006.

3 Richard F. Blute, J. Robin B. Cockett, and Robert A. G. Seely. Cartesian differential categories. *Theory and Applications of Categories*, 22(23):622–672, 2009.

4 Richard F. Blute, J. Robin B. Cockett, and Robert A. G. Seely. Cartesian differential storage categories. *Theory and Applications of Categories*, 30(18):620–686, 2015.

5 Thomas Ehrhard and Laurent Regnier. The differential lambda-calculus. *Theoretical Computer Science*, 309(1):1–41, 2003.

6 Thomas Ehrhard and Laurent Regnier. Differential interaction nets. *Theoretical Computer Science*, 364(2):166–195, 2006.

7 Marcelo P. Fiore. Differential structure in models of multiplicative biadditive intuitionistic linear logic. In *International Conference on Typed Lambda Calculi and Applications*, pages 163–177. Springer, 2007.

8 André Joyal and Ross Street. The geometry of tensor calculus, I. *Advances in Mathematics*, 88(1):55–112, 1991.

9 Saunders Mac Lane. *Categories for the working mathematician*, volume 5. Springer Science & Business Media, 2013.

10 Paul-André Mellies. Categorical models of linear logic revisited, 2003.

11 Robert A. G. Seely. *Linear logic, *-autonomous categories and cofree coalgebras.* Ste. Anne de Bellevue, Quebec: CEGEP John Abbott College, 1987.

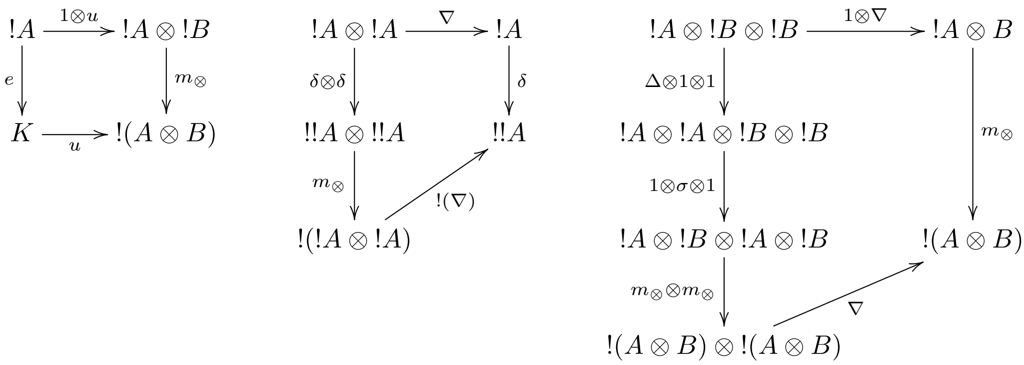
12 Peter Selinger. A survey of graphical languages for monoidal categories. In *New structures for physics*, pages 289–355. Springer, 2010.

13 Shilong Zhang, Li Guo, and William Keigher. Monads and distributive laws for rota–baxter and differential algebras. *Advances in Applied Mathematics*, 72:139–165, 2016.

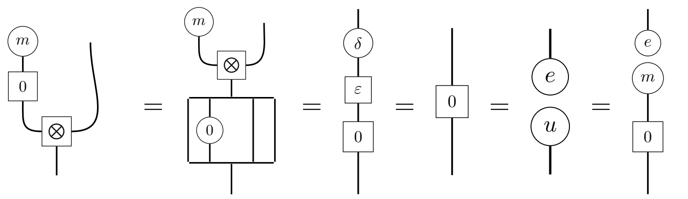
A Compatibilities of the Bialgebra Modality

We provide the proof of some compatibility relations between the bialgebra modality and the monoidal structure:

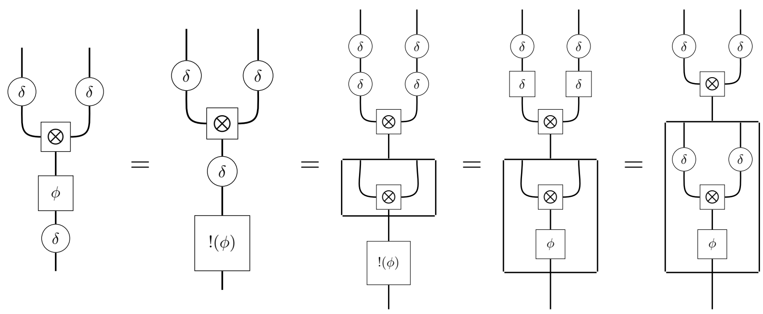
▶ **Lemma 26.** *In an additive linear category, the following diagrams commute:*



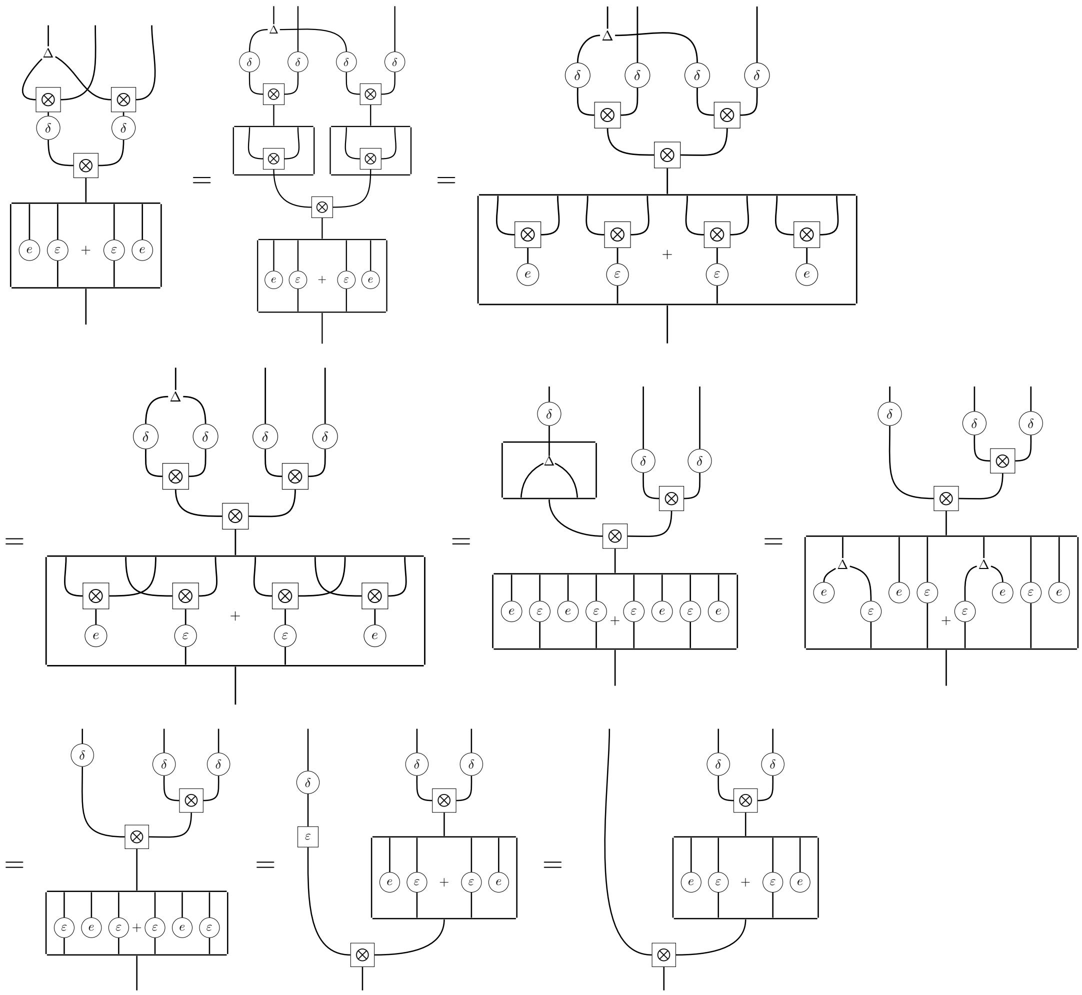
Proof. For the first square on the left we use the naturality of $m_\otimes$, the unit laws of the monoidal functor and that $!(0)$ splits:



Now, for space and simplification, define $\phi = e \otimes \varepsilon + \varepsilon \otimes e$. Then for the middle square, we use the naturality of $m_\otimes$, the comonad δ square and that δ is a monoidal transformation:



For the square on the right we use that δ is a monoidal transformation, naturality of $m_\otimes$, associativity and symmetry of $m_\otimes$, that e and ε are monoidal transformations and that Δ is a !-coalgebra morphism:



◀

B Seely Isomorphisms

In a symmetric monoidal category with finite products $\times$ and terminal object T, a coalgebra modality has **Seely isomorphisms** [1, 4] if the natural transformations χ and χ_{T} defined respectively as:

$$!(A \times B) \xrightarrow{\Delta} !(A \times B) \otimes !(A \times B) \xrightarrow{!(\pi_0)\otimes !(\pi_1)} !A \otimes !B \qquad !(\mathsf{T}) \xrightarrow{e} K$$

are isomorphisms, so $!(A \times B) \cong !A \otimes !B$ and $!(\mathsf{T}) \cong K$.

▶ **Definition 27.** A **monoidal storage category** [4] is a symmetric monoidal category with finite products and a coalgebra modality which has Seely isomorphisms.

Monoidal storage categories were called **new Seely categories** in [1, 10]. As explained in [4], every coalgebra modality which satisfies the Seely isomorphisms is a monoidal coalgebra modality, where $m_\otimes$ is

$$!A \otimes !B \xrightarrow{\chi^{-1}} !(A \times B) \xrightarrow{\delta} !!(A \times B) \xrightarrow{!(\chi)} !(!A \otimes !B) \xrightarrow{!(\varepsilon\otimes\varepsilon)} !(A \otimes B)$$

and m_K is defined as

$$K \xrightarrow{\chi_{\mathsf{T}}^{-1}} !(\mathsf{T}) \xrightarrow{\delta} !!(\mathsf{T}) \xrightarrow{!(\chi_{\mathsf{T}})} !(K)$$

Conversly, in the presence of finite products, every monoidal coalgebra modality satisfies the Seely isomorphisms [1] where the inverse of χ is

$$!A \otimes !B \xrightarrow{\delta \otimes \delta} !!A \otimes !!B \xrightarrow{m_\otimes} !(!A \otimes !B) \xrightarrow{!(\langle \varepsilon \otimes e, e \otimes \varepsilon \rangle)} !(A \times B)$$

while the inverse of χ_{T} is

$$K \xrightarrow{m_K} !(K) \xrightarrow{!(\mathsf{t})} !(\mathsf{T})$$

where $\mathsf{t} : K \to \mathsf{T}$ is the unique map to the terminal object. Therefore we obtain the following theorem (Theorem 3.1.6 [4]):

▶ **Theorem 28.** *Every monoidal storage category is a linear category and conversely, every linear category with finite products is a monoidal storage category.*

This together with Appendix C provides a rather indirect verification of Theorem 19.

We now turn our attention to monoidal storage categories with an additive structure:

▶ **Definition 29.** An **additive monoidal storage category** is a monoidal storage category which is also an additive symmetric monoidal category.

Notice, this implies that additive monoidal storage categories have finite biproducts $\times$ and a zero object 0. As noted in [2], the coalgebra modality of an additive monoidal storage category is an additive bialgebra modality where the multiplication and unit are defined respectively as:

$$!A \otimes !A \xrightarrow{\chi^{-1}} !(A \times A) \xrightarrow{!(\nabla_\times)} !(A) \qquad K \xrightarrow{\chi_0^{-1}} !0 \xrightarrow{!(0)} !A$$

Conversly, every additive bialgebra modality satisfies the Seely isomorphisms where χ^{-1} and χ_0^{-1} are defined respectively as:

$$!A \otimes !B \xrightarrow{!(\iota_0) \otimes !(\iota_1)} !(A \times B) \otimes !(A \times B) \xrightarrow{\nabla} !(A \times B) \qquad K \xrightarrow{u} !0$$

It is easy to check that this indeed gives the Seely isomorphisms, therefore we have:

▶ **Theorem 30.** *The following are equivalent:*
(i) *An additive monoidal storage category;*
(ii) *An additive linear category with finite biproducts;*
(iii) *An additive symmetric monoidal category with finite biproducts and an additive bialgebra modality.*

C Biproduct Completion

Every additive bialgebra modality induces an additive monoidal storage category the biproduct completion. We first recall the biproduct completion for an additive category [9].

Let $\mathbb{X}$ be an additive category. Define the biproduct completion of $\mathbb{X}$, $\mathsf{B}[\mathbb{X}]$, as the category whose objects are list of objects of $\mathbb{X}$: $(A_1, \ldots, A_n)$, including the empty list $()$, and whose maps are matrices of maps of $\mathbb{X}$, including the empty matrix:

$$(A_1, \ldots, A_n) \xrightarrow{[f_{i,j}]} (B_1, \ldots, B_m)$$

where $f_{i,j} : A_i \to B_j$. The composition in $\mathsf{B}[\mathbb{X}]$ is the standard matrix multiplication:

$$[f_{i,j}][g_{l,k}] = [\sum f_{i,k} g_{k,j}]$$

while the identity is the standard identity matrix:

$$(A_1, \ldots, A_n) \xrightarrow{[\delta_{i,j}]} (A_1, \ldots, A_n)$$

where $\delta_{i,j} = 0$ if $i \neq j$, and $\delta_{i,i} = 1$. It is easy to see that $\mathsf{B}[\mathbb{X}]$ does in fact have biproducts:

▶ **Lemma 31.** $\mathsf{B}[\mathbb{X}]$ *is a well-defined category with biproducts.*

If $\mathbb{X}$ is an additive symmetric monoidal category, then so is $\mathsf{B}[\mathbb{X}]$. The monoidal unit is the same as in $\mathbb{X}$, the tensor product of objects is:

$$(A_1, \ldots, A_n) \otimes (B_1, \ldots, B_m) = (A_1 \otimes B_1, \ldots, A_1 \otimes B_m, \ldots, A_n \otimes B_n)$$

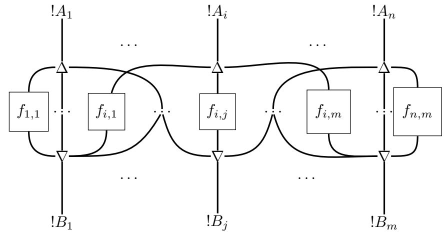
while the tensor product of maps is the standard Kronecker product of matrices.

▶ **Lemma 32.** *If* $\mathbb{X}$ *is an additive symmetric monoidal category, then so is* $\mathsf{B}[\mathbb{X}]$.

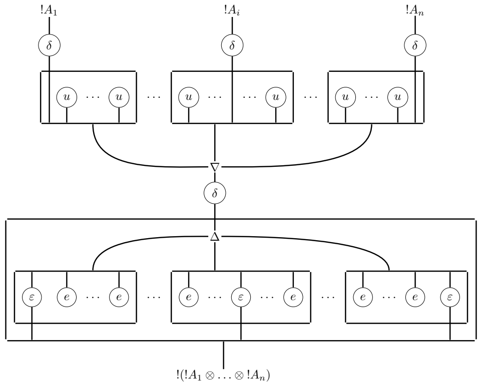
If $\mathbb{X}$ admits an additive bialgebra modality, then $\mathsf{B}[\mathbb{X}]$ is an additive monoidal storage category where the Seely isomorphisms are strict, i.e., equalities, and in particular is an additive linear category. We give the additive bialgebra modality of $\mathsf{B}[\mathbb{X}]$. The functor $! : \mathsf{B}[\mathbb{X}] \to \mathsf{B}[\mathbb{X}]$ is defined on objects as:

$$!(A_1, \ldots, A_n) = !A_1 \otimes \ldots \otimes !A_n$$

and on a map $[f_{i,j}] : (A_1, \ldots, A_n) \to (B_1, \ldots, B_m)$, $!([f_{i,j}])$ is represented in string diagrams below:



The bialgebra structure is given by the standard tensor product of bialgebras, the comonad comultiplication $!(A_1, \ldots, A_n) \to !!(A_1, \ldots, A_n)$ is represented in string diagrams as:



while the comonad counit is the following matrix:

$$!A_1 \otimes \ldots \otimes !A_n \xrightarrow{\begin{bmatrix} \varepsilon_{A_1} \otimes e \otimes \ldots \otimes e \\ \ldots \\ e \otimes \ldots \otimes \varepsilon_{A_i} \otimes \ldots \otimes e \\ \ldots \\ e \otimes e \otimes \ldots \otimes \varepsilon_{A_n} \end{bmatrix}} (A_1, \ldots, A_n)$$

▶ **Lemma 33.** *If* $\mathbb{X}$ *has an additive bialgebra modality, then* $\mathsf{B}[\mathbb{X}]$ *is an additive monoidal storage category.*