Math. Struct. in Comp. Science (2006), *vol.* 16, *pp.* 1049–1083.
doi:10.1017/S0960129506005676 Printed in the United Kingdom

Differential categories

R. F. BLUTE[†], J. R. B. COCKETT[‡] and R. A. G. SEELY[§]

[†]*Department of Mathematics, University of Ottawa, 585 King Edward St., Ottawa, ON, K1N 6N5, Canada*
Email: rblute@mathstat.uottawa.ca

[‡]*Department of Computer Science, University of Calgary, 2500 University Drive, Calgary, AL, T2N 1N4, Canada*
Email: robin@cpsc.ucalgary.ca

[§]*Department of Mathematics, McGill University, 805 Sherbrooke St., Montréal, PQ, H3A 2K6, Canada*
Email: rags@math.mcgill.ca

Received 3 July 2005; revised 31 May 2006

Following work of Ehrhard and Regnier, we introduce the notion of a *differential category*: an additive symmetric monoidal category with a comonad (a 'coalgebra modality') and a differential combinator satisfying a number of coherence conditions. In such a category one should imagine the morphisms in the base category as being linear maps and the morphisms in the coKleisli category as being smooth (infinitely differentiable). Although such categories do not necessarily arise from models of linear logic, one should think of this as replacing the usual dichotomy of linear *vs.* stable maps established for coherence spaces.
After establishing the basic axioms, we give a number of examples. The most important example arises from a general construction, a comonad S_∞ on the category of vector spaces. This comonad and associated differential operators fully capture the usual notion of derivatives of smooth maps. Finally, we derive additional properties of differential categories in certain special cases, especially when the comonad is a storage modality, as in linear logic. In particular, we introduce the notion of a *categorical model of the differential calculus*, and show that it captures the not-necessarily-closed fragment of Ehrhard–Regnier differential λ-calculus.

1. Introduction

Linear logic (Girard 1987) originated with Girard's observation that the internal hom in the category of stable domains decomposed into a linear implication and an endofunctor:

$$A \Rightarrow B = \,!A \multimap B.$$

The categorical content of this observation, *viz.*, the interpretation of $!$ as a comonad and, given appropriate coherence conditions, the fact that the coKleisli category was cartesian closed, was subsequently described in Seely (1989). Thus, the category of stable domains came to be viewed in a rather different light as the coKleisli category for the comonad $!$ on the category of coherence spaces. Coherence spaces, of course, provided, for Girard, the principal model underlying the development of linear logic.

More recently, in a series of papers, Ehrhard and Regnier introduced the *differential λ-calculus* and *differential proof nets* (Ehrhard 2001; Ehrhard and Regnier 2005; Ehrhard and Regnier 2003; Ehrhard 2005). Their work began with Ehrhard's construction of models of linear logic in the categories of *Köthe spaces* and *finiteness spaces*. They noted that these models had a natural notion of *differential operator* and made the key observation that the logical notion of 'linear' (using arguments exactly once) coincided with the mathematical notion of linear transformation (which is essential to the notion of a derivative, as the best linear approximation of a function). This observation is central to the decision to situate a categorical semantics for differential structure in appropriately endowed monoidal categories.

Our aim in this paper is to provide a categorical reconstruction of the Ehrhard–Regnier differential structure. In order to achieve this we introduce the notion of *differential category*, which captures the key structural components required for a basic theory of differentiation. As with Ehrhard–Regnier models, the objects of a differential category should be thought of as spaces that possess a modality (a comonad)†; the maps should be thought of as linear, while the coKleisli maps for the modality should be interpreted as being smooth.

It is important to note that differential categories are, essentially, a more general notion than that introduced by Ehrhard and Regnier in two important respects. First, differential categories are monoidal, rather than monoidal closed or $*$-autonomous, additive categories. This is crucial, as it allows us to capture various 'standard models' of differentiation that are notably not closed. Second, we do not require the $!$ comonad to be a 'storage' modality in the usual sense of linear logic (as described by Bierman (1995), for example). Specifically, we do not require the comonad to be monoidal, although we do require that the cofree coalgebras carry the structure of a commutative comonoid: these we call coalgebra modalities. Again, this seems necessary, as the standard models we consider do not necessarily give rise to a full storage modality. That said, we do agree that the special case of storage modalities has an important role in this theory.

It is natural to ask what the form of a differential combinator should be in a monoidal category with a modality $!$. A smooth map from A to B is just a linear map $f : \,!A \longrightarrow B$. To see what the type of its differential should be, consider a simple example from multivariable calculus: $f(x, y, z) = \langle x^2 + xyz, z^3 - xy \rangle$. This is a smooth function from $\mathbb{R}^3$ to $\mathbb{R}^2$. Its Jacobian is $\begin{pmatrix} 2x+yz & xz & xy \\ -y & -x & 3z^2 \end{pmatrix}$. Given a choice of x, y and z, that is, a point of $\mathbb{R}^3$, we obtain a *linear* map from $\mathbb{R}^3$ to $\mathbb{R}^2$. But the assignment of the linear map for a point is smooth. So, given a map $f : A \subseteq \mathbb{R}^n \longrightarrow \mathbb{R}^m$, one gets a smooth map $D(f) : A \longrightarrow L(\mathbb{R}^n, \mathbb{R}^m)$: for a point $x \in A$, $D(f)$ is given by the Jacobian of f at x. In general, the type of the differential should be $D(f) : \,!A \longrightarrow A \multimap B$. As we are working in not-necessarily-closed categories, we simply transpose this map, and obtain a differential combinator of the form:

$$\frac{f : \,!A \longrightarrow B}{D[f] : A \otimes \,!A \longrightarrow B} .$$

So, from our perspective, a differential category will be an additive symmetric monoidal category with coalgebra modality and a differential combinator, as above, which must satisfy various equations that are familiar from first year calculus.

† In fact, Ehrhard and Regnier (2005) does not require a comonad, since they do not have the promotion rule in their system. But they indicate that their system extends easily; this is not the significant difference between the two approaches. In fact, Ehrhard and Regnier (2003) does have promotion, *via* the λ calculus.

Note also that this example suggests

$$\text{Smooth}(\mathbb{R}^3, \mathbb{R}^2) = \text{Lin}(\mathbb{R}^2, S(\mathbb{R}^3))$$

where $S(V)$ = smooth functions from V to R. Consider f as above, a smooth map $\mathbb{R}^3 \longrightarrow \mathbb{R}^2$, which may be thought of as a pair of smooth maps $\mathbb{R}^3 \longrightarrow \mathbb{R}$ and hence as a linear map $\mathbb{R}^2 \longrightarrow S(\mathbb{R}^3)$. We shall see this again in Proposition 3.5.

In Section 2 we introduce these notions and, in particular, we note that it suffices to differentiate the identity on $!A$, as all other differentials can be obtained from this by composition. This gives the notion of a deriving transformation, which was introduced in Ehrhard (2001). Given the appropriate coherence conditions, we show that having a deriving transformation is equivalent to having a differential combinator. The remainder of Section 2 is devoted to examples. We show that the category of sets and relations and the category of sup-lattices have very simple differential structures. For a more significant example, we take (the opposite of) the category of vector spaces. The free commutative algebra construction here provides us with a comonad, and when elements of that algebra are interpreted as polynomials, the usual notion of the derivative of polynomials provides a differential combinator.

In Section 3, we extend this idea to general smooth functions by introducing a new construction, which we call S_∞. This is a general construction, which, given a *polynomial theory* over a rig R, allows one to produce a coalgebra modality on the opposite of the category of R-modules. If, furthermore, the polynomial theory has partial derivatives, and is thus a differential theory, this can be translated into a differential combinator associated with the modality. The construction shows how to associate a differential combinator with any reasonable notion of smoothness.

In Section 4, we explore certain special cases of the notion of a differential category. In particular, we consider the case where the comonad actually satisfies the additional requirements of being a storage modality, that is, a model of the exponential modality of linear logic. In the case where the category additionally has biproducts, we define the notion of a *categorical model of the differential calculus*, and show that this structure characterises the not-necessarily-closed version of the Ehrhard–Regnier differential λ-calculus.

2. Differential categories

Throughout this paper we will be working with **additive**† symmetric monoidal categories, by which we mean that the homsets are enriched in commutative monoids so that we may 'add' maps $f + g$, and there is a family of zero maps, 0. Recall that there are important examples of categories that are additive in this sense but are not enriched in Abelian groups: sets and relations (with tensor given by cartesian product), suplattices, and commutative monoids are all examples. To be explicit, the composition in additive

† We should emphasise that our 'additive categories' are commutative monoid enriched categories, rather than Abelian group enriched; in fact, some people might prefer to call them 'semi-additive'. Furthermore, we do not require biproducts as part of the structure at this stage. In particular, our definition is not the same as the one in Mac Lane (1971).

categories, which we write in diagrammatic order, is 'biadditive' in the sense that $h(f+g) = hf + hg$, $(f+g)k = fk + gk$, $h0 = 0$ and $0k = 0$. The tensor $\otimes$ is assumed to be enriched so that $(f+g)\otimes h = f \otimes h + g \otimes h$ and $0 \otimes h = 0$.

A differential category is an additive symmetric monoidal category with a coalgebra modality and a differential combinator. A coalgebra modality often arises as a 'storage modality', and a monoidal category with such a modality is a model of linear logic. However, we have deliberately avoided that nomenclature here because the modalities we consider are not restricted to commutative coalgebras, nor do they necessarily satisfy the coherences expected of storage. Recall that, for a storage modality, the coKleisli category is a cartesian category that is canonically linked to the starting category by a monoidal adjunction. This adjunction turns the tensor in the original category into a product and produces the storage isomorphism (sometimes called the Seely isomorphism): $!(A \times B) \cong\ !A \otimes\ !B$.

It is because the computational intuition of Girard's 'storage' modality does not have significant resonance with the developments in this paper that we have chosen to use the nomenclature derived from a more traditional source, though storage modalities are an important basis for some of the examples. When a category is additive or, more precisely, commutative monoid enriched, the comonoid associated with the modality is precisely what the majority of algebraists would simply call a coalgebra, and it seems natural to emphasise, in this context, these connections. We shall use the term 'storage modality' when we wish to impose the extra coherence conditions usual in categorical models of linear logic.

The notion of a differential combinator is the new ingredient of this work and is described below. Before introducing this notion it is worth emphasising the peculiar role the modality plays in this work. Here, as in Ehrhard's original work, the modality is a comonad for which the coKleisli category is regarded as a category of differentiable functions: the maps of the parent category are the linear maps. The idea of a differential combinator is that it should mediate the interaction between these two settings.

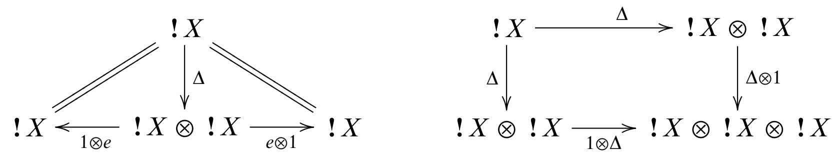
2.1. *Coalgebra modalities*

Definition 2.1. A comonad $(\,!, \delta, \epsilon)$ on an additive symmetric monoidal category, $\mathbb{X}$, is a **coalgebra modality** when each object $!X$ comes equipped with a natural coalgebra structure given by

$$\Delta\colon\ !X \longrightarrow\ !X \otimes\ !X \qquad e\colon\ !X \longrightarrow \top$$

where $\top$ is the tensor unit. This data must satisfy the following basic coherences:

1. $(\,!X, \Delta, e)$ is a **comonoid**:



2. δ is a morphism of these comonoids:

$$
\begin{array}{ccc}
!X & \xrightarrow{\ \delta\ } & !!X \\
 & {\scriptstyle e}\searrow \quad \swarrow{\scriptstyle e} & \\
 & \top &
\end{array}
\qquad\qquad
\begin{array}{ccc}
!X & \xrightarrow{\ \delta\ } & !!X \\
{\scriptstyle \Delta}\downarrow & & \downarrow{\scriptstyle \Delta} \\
!X \otimes !X & \xrightarrow[\delta\otimes\delta]{} & !!X \otimes !!X
\end{array}
$$

Note that we have *not* assumed that $!$ is monoidal or that any of the transformations are monoidal. This may occasionally be the case, but, in general, it need not be so.

A coKleisli map $!A \longrightarrow B$ will be viewed as an abstract differentiable map from $A \longrightarrow B$ so that the coKleisli category $\mathbb{X}_!$ is the category of abstract differentiable maps for the setting. Of course, for this to make sense we shall need more structure, which will be introduced in the next subsection. Meanwhile, we can give some examples of coalgebra modalities on additive categories.

Example 2.2.

1. For any cartesian category the identity monad is a coalgebra modality where the coalgebra structure is given by the diagonal and final map on the product.
2. A storage modality (the 'bang' from linear logic) on a monoidal category is a rather special example. These are discussed further in Section 4.
3. One way to obtain a coalgebra modality is to take the dual of an algebra modality. There are a number of such examples from commutative algebra (see Lang (2002)):
   (a) the free algebra $T(X) = \bigoplus_{r=0}^{\infty} X^{\otimes^r}$, where $\oplus$ denotes the biproduct
   (b) the free symmetric algebra $\mathrm{Sym}(X) = \bigoplus_{r=0}^{\infty} X^{\otimes^r}/\mathscr{S}_r$
   (c) the 'exterior algebra' $\Lambda(X) = \bigoplus_{r=0}^{\infty} X^{\otimes^r}/\mathscr{A}$ is the free algebra generated by the module X subject to the relation that monomials $v_1 v_2 \ldots v_n = 0$ whenever $v_i = v_j$ where $i \neq j$. This makes the algebra anti-commute in the sense that $xy = -yx$.

   We will use this source of examples in Section 3 and provide a general way of constructing such monads that will allow us to capture all the classical notions of differentiation.

In addition, there are a number of other, less standard, examples, which we shall briefly describe in the course of developing the general theory.

2.2. *Differential combinators*

Definition 2.3. For an additive symmetric monoidal category $\mathscr{C}$ with a coalgebra modality $!$, a (left) **differential combinator** $D_{AB}\colon \mathscr{C}(\,!A, B) \longrightarrow \mathscr{C}(A \otimes \,!A, B)$ produces for each coKleisli map $f\colon\, !A \longrightarrow B$ a (left) **derivative** $D_{AB}[f]\colon A \otimes \,!A \longrightarrow B$:

$$
\frac{!A \xrightarrow{\ f\ } B}{A \otimes \,!A \xrightarrow[D[f]]{} B}\,,
$$

which must satisfy the coherence requirements (**[D.1]** to **[D.4]** below), the principal one of which is the chain rule.

It should be mentioned that if the monoidal category is closed, a differential combinator can be re-expressed as

$$\frac{A \otimes\, !A \xrightarrow{D[f]} B}{\overline{\;!A \xrightarrow[\hat{D}[f]]{} A \multimap B\;}} \;.$$

In other words, from the original differentiable map, one obtains a new differentiable map into the space of linear transformations. Intuitively, this associates with each point of the domain the linear map that approximates the original map at that point.

A differential combinator must satisfy the usual property of a functorial combinator: namely, that it is additive, in other words $D[0] = 0$ and $D[f+g] = D[f] + D[g]$, and it carries commuting diagrams to commuting diagrams, so D_{AB} is natural in A and B:

$$\frac{\begin{array}{ccc} !A & \xrightarrow{f} & B \\ {\scriptstyle !u}\downarrow & & \downarrow{\scriptstyle v} \\ !C & \xrightarrow[g]{} & D \end{array}}{\begin{array}{ccc} A \otimes\, !A & \xrightarrow{D[f]} & B \\ {\scriptstyle u\otimes !u}\downarrow & & \downarrow{\scriptstyle v} \\ C \otimes\, !C & \xrightarrow[D[g]]{} & D \end{array}}$$

In addition, a differential combinator must satisfy the following four identities†:

[D.1] Constant maps:

$$D[e_A] = 0\,.$$

[D.2] Product rule:

$$D[\Delta(f \otimes g)] = (1 \otimes \Delta)a_{\otimes}^{-1}(D[f] \otimes g) + (1 \otimes \Delta)a_{\otimes}^{-1}(c_{\otimes} \otimes 1)a_{\otimes}(f \otimes D[g])\,.$$

where $f\colon !A \longrightarrow B$, $g\colon !A \longrightarrow C$, and $a_{\otimes}, c_{\otimes}$ are the associativity and commutativity isomorphisms.

[D.3] Linear maps:

$$D[\epsilon_A f] = (1 \otimes e_A)u_{\otimes}^{R} f$$

where $f\colon A \longrightarrow B$ and $u_{\otimes}$ is the unit isomorphism.

[D.4] The chain rule:

$$D[\delta\; !f\; g] = (1 \otimes \Delta)a_{\otimes}^{-1}(D[f] \;\otimes\; \delta\; !f)D[g]\,,$$

that is

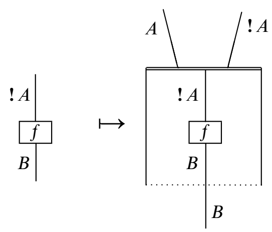
$$\frac{!A \xrightarrow{\delta} \;!!A \xrightarrow{!f} \;!B \xrightarrow{g} C}{A \otimes\, !A \xrightarrow[1\otimes\Delta]{} A \otimes (\,!A \otimes\, !A) \xrightarrow[a_{\otimes}^{-1}]{} (A \otimes\, !A) \otimes\, !A \xrightarrow[D[f]\otimes(\delta\, !f)]{} B \otimes\, !B \xrightarrow[D[g]]{} C} \;.$$

† Recall that we use ‘diagrammatic notation’: fg means ‘first f, then g’.

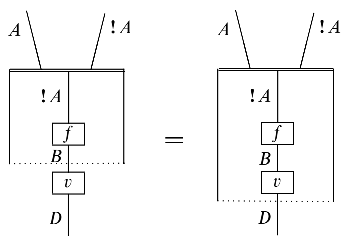
Each of these identities should accord immediately with the intuition of a derivative, as they are quite literally simply a re-expression in categorical terminology of the standard requirements of a derivative. Constant functions have derivative 0. The tensor of two functions on the same arguments is morally the product of two functions (on the unit $\top$ this is literally true), thus the second rule is just the familiar product rule from calculus. The derivative of a map that is linear is, of course, constant. The derivative of the composite of two functions is the derivative of the first function composed with the derivative of the second function at the value produced by the first function: in other words, the chain rule holds.

2.3. *Circuits for differential combinators*

Readers of previous papers by the present authors will be familiar with our use of circuits (or proof nets adapted to our context); a good introduction to our circuits, relevant to their use here, are Blute *et al.* (1997) and Blute *et al.* (1996). It is no surprise that a similar technique will work in the present situation: we may represent the differential operator using circuits, using a 'differential box'

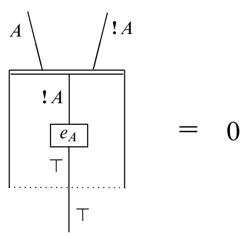


Note that the naturality of D means (by taking $u = 1$) that one can move a component in and/or out of the bottom of a differential box (and so, in a sense, the box is not really necessary – we shall return to this point soon).

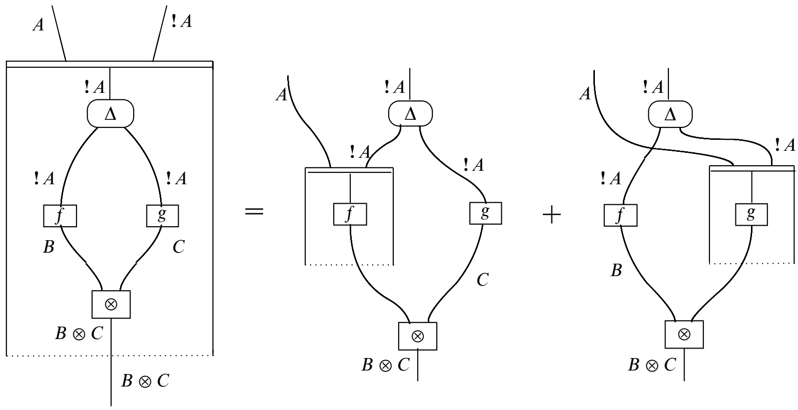


The rules can also be represented as additive circuits:

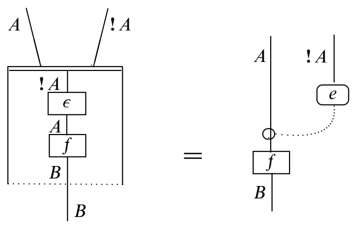
[D.1] Constant maps:



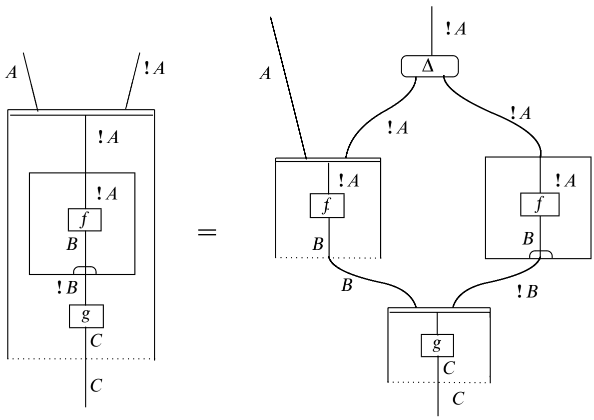
[D.2] Product rule:



[D.3] Linear maps:



[D.4] Chain rule:



Notice that in the chain rule we use two sorts of boxes: the differential box and the comonad box (Blute *et al.* 1996). This latter box embodies following inference

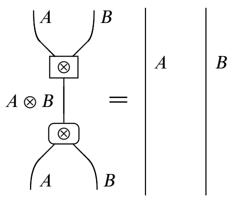
$$\frac{!A \xrightarrow{f} B}{!A \xrightarrow[f^\flat = \delta\, !f]{} !B},$$

which allows an alternative presentation of a monad, which was originally given in Manes (1976) and was used to describe storage modalities in Blute *et al.* (1996), following the usage introduced in Girard (1987).

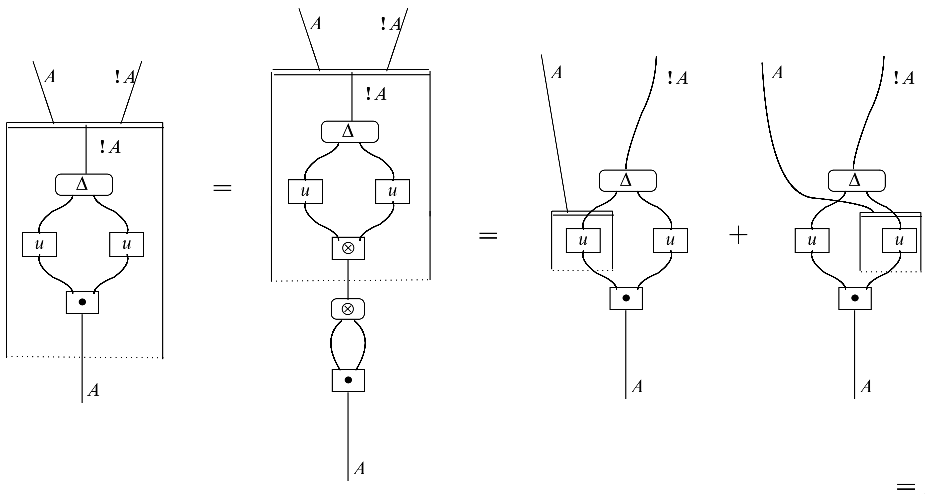
So we can restate our fundamental definition as follows.

Definition 2.4. A **differential category** is an additive symmetric monoidal category with a coalgebra modality and a differential combinator.

As an example of a simple derivative calculation using circuits, we can calculate the derivative of u^2 (which the reader may not be surprised to discover is $2u \cdot u'$). We suppose there is a commutative multiplication $A \otimes A \xrightarrow{\bullet} A$, so u^2 means $u \bullet u$. We make use of some simple graph rewrites introduced in Blute *et al.* (1997); in particular, one can join and then split two wires with tensor nodes without altering the identity of the circuit.



Then, using the other rewrites for a differential combinator, we obtain $D(u^2) =$



$$= 2u \bullet u'$$

2.4. *Deriving transformations*

It is convenient for the calculations we will perform to re-express the notion of a differential combinator in a more primitive form. A special case of the functorial property of a differential combinator is the action on identity maps

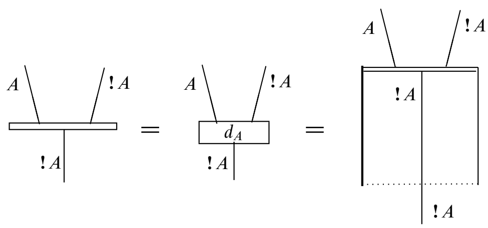
$$\dfrac{\begin{array}{ccc} !A & \xrightarrow[1_{!A}]{} & !A \\ {\scriptstyle !u}\downarrow & & \downarrow{\scriptstyle !u} \\ !B & \xrightarrow[1_{!B}]{} & !B \end{array}}{\begin{array}{ccc} A \otimes\ !A & \xrightarrow{D[1_{!A}]} & !A \\ {\scriptstyle u\otimes !u}\downarrow & & \downarrow{\scriptstyle !u} \\ B \otimes\ !B & \xrightarrow[D[1_{!B}]]{} & !B \end{array}}$$

which produces a natural transformation below the line. The map $D[1_{!A}]$ produced in this manner will be denoted d_A for both simplicity and to remind us that it is natural in A.

This map occurs in another revealing instance of functoriality for the differential combinator:

$$\dfrac{\begin{array}{ccc} !A & \xrightarrow{1} & !A \\ {\scriptstyle !1}\downarrow & & \downarrow{\scriptstyle f} \\ !A & \xrightarrow[f]{} & B \end{array}}{\begin{array}{ccc} A \otimes\ !A & \xrightarrow{d_A} & !A \\ {\scriptstyle 1\otimes !1}\downarrow & & \downarrow{\scriptstyle f} \\ A \otimes\ !A & \xrightarrow[D[f]]{} & B \end{array}}$$

Consequently, the natural transformation d_A actually generates the whole differential structure. In terms of circuits, this says that the boxes are 'bottomless', which justifies our circuit notation for the differential combinator, and motivates the following circuit notation for the differential operator, with a component box representing the combinator box 'shrunk' to having only an identity wire inside. We shall normally use the first notation, which represents a differential box 'pulled back' past the identity; if we wish to emphasise the component d_A, we shall use the second notation. The third notation is the equivalent presentation using a differential box.



The properties of a differential combinator may be re-expressed succinctly in terms of the following transformation.

Definition 2.5. For an additive category with a coalgebra modality, a natural transformation $d_X : X \otimes\ !X \longrightarrow\ !X$ is a (left) **deriving transformation** when it satisfies the following conditions:

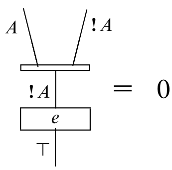
[d.1] Constant maps: $d_A e_A = 0$
[d.2] Copying: $d_A \Delta = (1 \otimes \Delta)a_\otimes^{-1}(d_A \otimes 1) + (1 \otimes \Delta)a_\otimes^{-1}(c_\otimes \otimes 1)a_\otimes(1 \otimes d_A)$
[d.3] Linearity: $d_A \epsilon_A = (1 \otimes e)u_\otimes^R$
[d.4] Chaining: $d_A \delta = (1 \otimes \Delta)a_\otimes^{-1}(d_A \otimes \delta)d_{!A}$.

For completeness, we have included all the coherence transformations in this definition: in subsequent calculations we shall omit them (this will be without loss of generality in view of Mac Lane's coherence theorem), assuming that the setting is strictly monoidal. Although we cannot drop the symmetry transformation $c_\otimes : A \otimes B \longrightarrow B \otimes A$, this does allow, for example, **[d.2]** to be stated a little more succinctly as

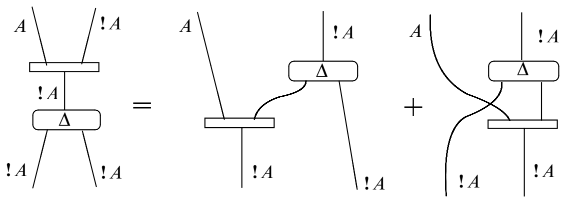
$$d_A \Delta = (1 \otimes \Delta)(d_A \otimes 1) + (1 \otimes \Delta)(c_\otimes \otimes 1)(1 \otimes d_A).$$

Of course, the circuit representation has the advantage of handling *all* the coherence issues painlessly. These rules may be presented as circuits as follows.

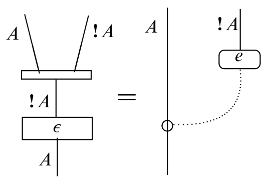
[d.1] Constant maps:



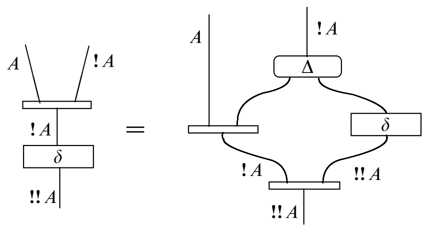
[d.2] Copying:



[d.3] Linearity:



[d.4] Chaining:



The main observation is then given by the following proposition.

Proposition 2.6. The following are equivalent:

(i) an additive symmetric monoidal category with a deriving transformation for its coalgebra modality

(ii) a differential category.

Proof. It is easy to check that a differential category satisfies all these identities. Conversely, the interpretation of the derivative using the natural transformation is, as indicated above, $D[f] = d_A f$. When d_A is natural, this immediately provides a functorial combinator $D[f] = d_A f$. Checking that this combinator satisfies the requirements of a derivative is straightforward with the possible exception of the chain rule:

$$\begin{aligned} D[\delta\ !f\, g] &= d_A\delta\ !f\, g \\ &= (1 \otimes \Delta)(d_A \otimes \delta)d_{!A}\ !f\, g \\ &= (1 \otimes \Delta)(d_A f \otimes (\delta\ !f))d_B g \\ &= (1 \otimes \Delta)(D[f] \otimes (\delta\ !f))D[g]. \end{aligned}$$ □

This means that in order to check that we have a differential category, we may check **[d.1]**–**[d.4]**, which are considerably easier than our starting point.

2.5. *Examples of differential categories*

In the following subsections we give some basic examples of differential categories.

2.5.1. *Sets and relations* On sets and relations (where the additive enrichment is given by unions and the tensor is given by cartesian product), the converse of the free commutative monoid monad (commonly known as the 'bag' functor) is a storage modality with respect to the tensor provided by the product in sets. There is an obvious natural transformation

$$d_X : X \otimes\ !X \longrightarrow\ !X\ :\ x_0, \{\![x_1, \ldots, x_n]\!\} \mapsto \{\![x_0, x_1, \ldots, x_n]\!\}$$

given by adding the extra element to the bag.

Proposition 2.7. The category of sets and relations with the bag functor and the above differential transformation is a differential category.

Proof (sketch). We proceed by checking the identities:

[d.1] d_X produces only non-empty bags; e is the partial function whose domain is the empty bag, which is sent to the point of $\top$. So the composite of d with e is 0.

[d.2] The copying map relates a bag to all the pairs of bags whose union it is. If one adds an element and then takes all the decompositions, it is the same as taking all the decompositions before adding the element and then taking the union of the possibilities provided by adding the element to each component of each decomposition.

[d.3] ϵ: $!X \longrightarrow X$ is the relation that is the converse of the map that picks out the singleton bag corresponding to $x \in X$, so ϵ is the partial function whose domain is the singletons, which are mapped to themselves. Hence the only pairs that survive $d_X\epsilon_X$ are those that were paired with the empty bag.

[d.4] The relation $\delta\colon {!X} \longrightarrow {!!X}$ associates to a bag all bags of bags whose 'union' is the bag. If one adds an extra element to a bag, when the bag is decomposed in this manner, the added element must occur in at least one component. This means this decomposition can be obtained by doing a binary decomposition that first extracts the component to which the element is added on the left while the right component contains what remains and can be decomposed to give the original decomposition when the left component (with extra element) is added. □

Exactly the same reasoning can be used to show that the power set monad, which is also a coalgebra modality for relations, has a differential combinator obtained by adding an element to each subset.

2.5.2. *Suplattices* The category of suplattices, sLat, that is, the category of lattices with arbitrary joins and maps that preserve these joins, is a well-known $*$-autonomous category (Barr 1979). It contains as a subcategory the category of sets and relations. It has a storage modality, which can be described in various ways. It is the de Morgan dual of the free $\oplus$-algebra functor (see Hyland and Schalk (2003)), but, more explicitly, it has the underlying object ${!X} = \bigoplus_{r=0}^{\infty} X^{\otimes^r}/\mathscr{S}_r$ and comultiplication $\Delta\colon {!X} \longrightarrow {!X} \otimes {!X}$, which, because sums and product coincide in this category, is determined by maps

$$X^{\otimes^{i+j}}/\mathscr{S}_{i+j} \longrightarrow X^{\otimes^i}/\mathscr{S}_i \otimes X^{\otimes^j}/\mathscr{S}_j \ : \ \prod_{i=1}^{m} x_i^{k_1} \mapsto \bigvee_{k_i=k_i'+k_i''} \prod_{i=1}^{m} x_i^{k_i'} \otimes x_i^{k_i''} .$$

Intuitively, this maps a monomial to the join of all the pairs that when multiplied give the element. The fact that we are taking the joins over all possibilities makes the map invariant under the symmetric group.

Clearly, ${!X}$ is also the free commutative algebra with the usual commutative multiplication of monomials. This actually makes ${!X}$ a bialgebra (we will develop these ideas further in Section 4). Clearly, ${!X}$ not only has a comonad structure but also the monad structure that goes with being the free symmetric algebra. The comonad comultiplication is certainly a coalgebra morphism, but it is not an algebra morphism and so fails to be a bialgebra morphism.

There is an obvious map $d\colon X \otimes {!X} \longrightarrow {!X}$, which simply adds an element by multiplying by that element (using the symmetric algebra structure). It is now straightforward to prove the following proposition (the proof is, in fact, essentially the same as for sets and relations, Proposition 2.7).

Proposition 2.8. sLat with respect to the above structure is a differential category.

2.5.3. *Commutative polynomials and derivatives* The category of modules, Mod_R (over any commutative ring R) has a free non-commutative algebra monad $\mathbb{T} = (T, \eta, \mu)$. On $\mathsf{Mod}_R^{\mathrm{op}}$ the free non-commutative algebra functor gives a comonad for which each $T(X)$ has a natural coalgebra structure. There is an 'obvious' differential structure on these non-commutative polynomials, which is determined by where it takes the monomials:

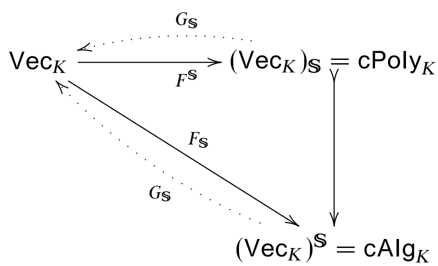
$$d(x_1^{m_1} x_2^{m_2} \ldots x_n^{m_n}) = \sum_{\substack{x = x_i \\ m = m_i}} m x^{m-1} \otimes m_1 x_1^{m_1 - 1} \ldots m_{i-1} x_{i-1}^{m_{i-1}-1} m_{i+1} x_{i+1}^{m_{i+1}-1} \ldots m_n x_n^{m_n - 1}.$$

Written in a more traditional form, this is just

$$d(f) = \sum x \otimes \frac{\partial f}{\partial x}.$$

This satisfies **[d.1]**, **[d.2]**, **[d.3]**, but, significantly, fails **[d.4]**. However, if one examines what goes wrong, it becomes clear that the free commutative algebra monad $\mathbb{S} = (S, \eta, \mu)$ should have been used.

The Eilenberg–Moore category for the commutative algebra monad is just the category of commutative R-algebras. While the Kleisli category is the subcategory of polynomial algebras. This gives, for a field K, the following diagram of adjoints (where the right adjoints are dotted):



Here cAlg_K is the category of commutative K-algebras, the Eilenberg–Moore category of the monad $\mathbb{S}$, and cPoly_K is the Kleisli category of the monad $\mathbb{S}$, which consists of the polynomial algebras over K (Mac Lane 1971). If we consider the effect of $\mathbb{S}$ on the opposite category $\mathsf{Vec}_K^{\mathrm{op}}$, then $\mathbb{S}$ becomes a comonad and, also, a coalgebra modality that has coKleisli category $\mathsf{cPoly}_K^{\mathrm{op}}$. It is well known that both $\mathsf{cPoly}_K^{\mathrm{op}}$ and $\mathsf{cAlg}_K^{\mathrm{op}}$ are distributive categories (Cockett 1993). In particular, this coKleisli category is the category of polynomial functions, since a map $f : V \longrightarrow W$ in $\mathsf{cPoly}_K^{\mathrm{op}}$ is a map $f : W \longrightarrow \mathscr{S}(V)$ in Vec_K and as such is determined by its basis: if $W = \langle w_1, w_2, \ldots \rangle$, then f is determined by its image on these elements. But $f(w_i) = \sum_{j=1,\ldots,m} a_{ij} \prod_{k=1,\ldots,l} v_k^{\otimes s_{kij}}$ where this is a finite sum and v_k are basis elements of V. Thus, our original function f may be viewed as a collection, one for each w_i, of polynomial functions in the basis of V. Composition in $\mathsf{cAlg}_K^{\mathrm{op}}$ is by substitution of these polynomial functions.

Our aim is now to provide a very concrete demonstration of the following proposition.

Proposition 2.9. $\mathsf{Vec}_K^{\mathrm{op}}$ with the opposite of the free commutative algebra monad is a differential category.

Furthermore, this is the standard notion of differentiation for these polynomial functions, so we have exactly captured the most basic notion of differentiation for polynomial functions taught in every freshman calculus class. The proof will occupy the remainder of this subsection.

Proof. Observe that if X is a basis for V, then $\mathrm{Sym}(V) \cong K[X]$, the polynomial ring over the field K (Lang 2002). We will only verify axioms **[d2]** and **[d4]**. Recall that we are working in the opposite of the category of vector spaces, so the maps are in the opposite direction.

They take on the following simple form:

- Δ becomes the ring multiplication.
- e is the inclusion of the constant polynomials.
- ϵ is the inclusion $V \longrightarrow K[X] : v \mapsto \sum_{i=1}^{n} r_i x_i$, where $v = \sum_{i=1}^{n} r_i x_i$, where this time the x_i are regarded as basis elements of V.
- As already noted, $d(f(x_1, x_2, \ldots, x_n)) = \sum_{i=1}^{n} x_i \otimes \frac{\partial f}{\partial x_i}$.
- Remembering that a basis for a polynomial ring is given by monomials, a typical basis element for $!!\, V$ is given by $[w_1]^{k_1}[w_2]^{k_2} \ldots [w_m]^{k_m}$. Then the map δ simply erases brackets.

We shall now do an elementwise argument on basis elements. To verify **[d2]** we have, for the left-hand side:

$$
\begin{aligned}
f \otimes g &\stackrel{\Delta}{\longmapsto} fg \\
&\stackrel{d_V}{\longmapsto} \sum_{i=1}^{n} x_i \otimes \frac{\partial (fg)}{\partial x_i} \\
&= \sum_{i=1}^{n} x_i \otimes \left[\frac{\partial f}{\partial x_i} g + \frac{\partial g}{\partial x_i} f \right] \\
&= \sum_{i=1}^{n} x_i \otimes \frac{\partial f}{\partial x_i} g + \sum_{i=1}^{n} x_i \otimes \frac{\partial g}{\partial x_i} f \,.
\end{aligned}
$$

For the right-hand side, we get

$$
\begin{aligned}
f \otimes g &\xmapsto{d_V \otimes 1 + (c_\otimes \otimes 1)(1 \otimes d_V)} \sum_{i=1}^{n} x_i \otimes \frac{\partial f}{\partial x_i} \otimes g + \sum_{i=1}^{n} x_i \otimes \frac{\partial g}{\partial x_i} \otimes f \\
&\xmapsto{1 \otimes \Delta} \sum_{i=1}^{n} x_i \otimes \frac{\partial f}{\partial x_i} g + \sum_{i=1}^{n} x_i \otimes \frac{\partial g}{\partial x_i} f \,.
\end{aligned}
$$

To verify **[d4]**, for the left-hand side,

$$
[w_1]^{k_1}[w_2]^{k_2} \ldots [w_m]^{k_m} \stackrel{\delta}{\longmapsto} w_1^{k_1} w_2^{k_2} \ldots w_m^{k_m} \stackrel{d_V}{\longmapsto} \sum_{i=1}^{n} x_i \otimes \frac{\partial (w_1^{k_1} w_2^{k_2} \ldots w_m^{k_m})}{\partial x_i} \,.
$$

The right-hand side yields

$$
\begin{aligned}
[w_1]^{k_1}[w_2]^{k_2}\dots[w_m]^{k_m} &\xmapsto{d_{!V}} \sum_{j=1}^{m} k_j w_j \otimes [w_1]^{k_1}\dots[w_j]^{k_j-1}\dots[w_m]^{k_m}\\
&\xmapsto{d_V\otimes\delta} \sum_{j=1}^{m}\sum_{i=1}^{n} k_j x_i \otimes \frac{\partial w_j}{\partial x_i} \otimes w_1^{k_1}\dots w_j^{k_j-1}\dots w_m^{k_m}\\
&\xmapsto{1\otimes\Delta} \sum_{j=1}^{m}\sum_{i=1}^{n} x_i \otimes w_1^{k_1}\dots k_j w_j^{k_j-1}\frac{\partial w_j}{\partial x_i}\dots w_m^{k_m}.
\end{aligned}
$$

The result then follows immediately from the product rule. □

It is worth noting that a direct proof, calculating the terms explicitly from an explicit definition of d_V,

$$
d_V : {!V} \longrightarrow V \otimes {!V} : \begin{cases} e & \mapsto\ 0 \\ \prod_{i=1}^{m} v_{r_i}^{s_i} & \mapsto\ \sum_{j=1}^{m} v_{r_j} \otimes s_j \cdot \prod_{i=1}^{m} v_{r_i}^{s_i-\delta_{ij}} \end{cases}
$$

where $\delta_{i,j}$ is the Kronecker delta ($\delta_{ij} = 1$ when $i = j$ and is zero otherwise), is also possible, but the calculations are quite appalling!

3. The S_∞ construction

One might well wonder whether there is not a better approach to understanding this sort of differential operator on $\mathsf{Vec}_K^{\mathrm{op}}$. After all, this calculation provides a theory that only covers the polynomial functions: even at high school one is expected to understand more, for example, the trigonometric functions!

Our aim in this section is therefore to show that, no matter what one cares to take as a (standard) basis for differentiable functions, one can construct an algebra modality on Vec_K for which there is a deriving transformation on $\mathsf{Vec}_K^{\mathrm{op}}$ that recaptures this notion of differentiation. We call this the S_∞ construction, as it allows one to realise various notions of infinite differentiability as differential combinators on $\mathsf{Vec}_K^{\mathrm{op}}$.

We shall break this programme down into stages. First we shall give a general method of constructing monads on a module category, Mod_R, from an algebraic structure on a rig R. A rig is a commutative monoid enriched over any additive system, and the algebraic structure is what we shall call a polynomial theory. Next we will show that if this algebraic structure supports partial derivatives, there is a corresponding (co)deriving transformation on the module category so that the dual category with this structure becomes a differential category.

3.1. *Polynomial theories to monads*

Let R be a commutative rig (that is a commutative monoid enriched over any additive structure). Then Mod_R is a symmetric monoidal closed category with (monoidal) unit R. Furthermore, there is an underlying functor $U\colon \mathsf{Mod}_R \longrightarrow \mathsf{Sets}$. We shall suppose that $U(R)$ is the initial algebra for an algebraic theory, $\mathbb{T}$, which includes the theory of commutative polynomials over R. In other words, the constants of $\mathbb{T}$ are exactly the elements of R, that is $r \in R$ if and only if $r \in \mathbb{T}(0,1)$ (where $\mathbb{T}(n,m)$ denotes the hom-set of the algebraic theory). The multiplication and addition are binary operations of $\mathbb{T}$, so that $\cdot, + \in \mathbb{T}(2,1)$, which on constants are defined as for R and otherwise satisfy the equations of being a commutative algebra over R. Note that $\mathbb{T}(n,1)$ includes $R[x_1,\ldots,x_n]$: for instance, $\cdot$ and $+$ correspond to x_1x_2 and x_1+x_2. We call such a theory $\mathbb{T}$ a **polynomial theory** over R.

An example of such a theory, which is central to this paper, for the field $\mathbb{R}$, is the 'smooth theory' of infinitely differentiable continuous real functions (and the same can be done for the complex numbers). The smooth theory then has $\mathbb{T}(n,1) = C^\infty(\mathbb{R}^n,\mathbb{R})$ with the constants being exactly the points in $\mathbb{R}$. Substitution determined by the usual substitution of functions gives the theory its categorical structure. This clearly introduces many more maps between the powers of reals than are present in $\mathsf{Vec}_{\mathbb{R}}$. We shall now show how to construct a monad on this category to represent these enlarged function spaces.

We shall use the following Kleisli triple form of a monad.

Proposition 3.1 (Manes 1976). The following data defines a monad $S:\mathbb{X} \longrightarrow \mathbb{X}$. An object function S together with assignments

$$(X \xrightarrow{f} S(Y)) \mapsto (S(X) \xrightarrow{f^\sharp} S(Y)) \qquad (X \in \mathbb{X}) \mapsto (X \xrightarrow{\eta_X} S(X))$$

satisfying three equalities: $\eta_X^\sharp = 1_{S(X)}$, $\eta f^\sharp = f$ and $f^\sharp g^\sharp = (fg^\sharp)^\sharp$.

Note that these ensure that S is a functor and η and $\mu = 1^\sharp_{S(X)}$ are natural transformations that form a monad in the usual sense.

The object part of the monad, which we shall call $S_{\mathbb{T}}$, is defined as follows:

$$S_{\mathbb{T}}(V) = \{h: V^* \longrightarrow R \;\mid\; \exists v_1,\ldots,v_n \in V, \alpha \in \mathbb{T}(n,1) \;\text{ so }\; h(u) = \alpha(u(v_1),\ldots,u(v_n))\}$$

where $V^* = V \multimap R$ where R is the monoidal unit in Mod_R. Note that h is really a map between the underlying sets of V^* and R, and so is not generally going to be linear.

We may think of h as a 'V-instantiation' of $\alpha \in \mathbb{T}(n,1)$. The choice of $v_1,\ldots,v_n$ determines scalars, so h may be viewed as $\alpha \in \mathbb{T}(n,1)$ operating on these scalars. But note that if V is finite dimensional over a field R, one can choose a basis once-and-for-all, making the choice of $v_1,\ldots,v_n$ unnecessary, so h may be identified with α (although different choices of the v_i may produce different h's, the set of h's is invariant). So in this case, $S_{\mathbb{T}}(V)$ is, essentially, the theory $\mathbb{T}$: for instance, if $\mathbb{T}$ is the 'pure' theory of polynomial functions, $S_{\mathbb{T}}(V)$ (as a set) is the symmetric algebra $\mathrm{Sym}(V)$, since $\mathrm{Sym}(V) \cong R[X]$, for X a basis for V (see Lang (2002), for example). Once we know $S_{\mathbb{T}}(V)$ is a monad, this will give the symmetric algebra monad (Proposition 3.5). When V is infinite dimensional over a field, $v_1,\ldots,v_n$ determines a finite dimensional subspace on which h can be viewed in this finite dimensional way, and so again, for the pure polynomial function theory, we get the symmetric algebra.

To show this is well defined, we must show that this set forms an R-module, in fact, a commutative R-algebra.

Lemma 3.2. $S_{\mathbb{T}}(V)$ as defined above is a commutative R-algebra.

Proof. We define $h_1 + h_2$ pointwise, where $h_i(u) = \alpha_i(u(v_{i1}), \ldots, u(v_{in_i}))$, as

$$(h_1 + h_2)(u) = \alpha_1(u(v_{11}), \ldots, u(v_{1n_1})) + \alpha_2(u(v_{21}), \ldots, u(v_{2n_2})),$$

which may be put into the required form with a suitable use of dummy variables, using the additivity of the theory $\mathbb{T}$. We define multiplication and multiplication by scalars similarly, so, for example,

$$(r \cdot h)(u) = r \cdot \alpha(u(v_1), \ldots, u(v_n)).$$

The requirement that scalar multiplication, addition, and multiplication satisfy the equations expected of a commutative algebra now imply that this is an R-algebra. □

Of course, this is still just a mapping on the objects. To obtain the monad, we need to define the $(\_)^\sharp$ operation and the η. Suppose $f: V \longrightarrow S_{\mathbb{T}}(W)$. We define $f^\sharp: S_{\mathbb{T}}(V) \longrightarrow S_{\mathbb{T}}(W)$ as

$$[h: u \mapsto \alpha(u(v_1), \ldots, u(v_n))] \mapsto [h': u' \mapsto \alpha(f(v_1)(u'), \ldots, f(v_n)(u'))]$$

where $f(v_i) = [u' \mapsto \beta_i(u'(v_{1i}), \ldots, u'(v_{n_i i}))]$, and η is evaluation:

$$\eta: V \longrightarrow S_{\mathbb{T}}(V) : v \mapsto [u \mapsto u(v)],$$

taking α to 1.

We must start by checking that both $f^\sharp$ and η are R-module maps. For η this is almost immediate, so we shall focus on $f^\sharp$. We have

$$\begin{aligned} r \cdot f^\sharp([u \mapsto \alpha(u(v_1), \ldots, u(v_n))]) &= r \cdot [u' \mapsto \alpha(f(v_1)(u'), \ldots, f(v_n)(u'))] \\ &= [u' \mapsto r \cdot \alpha(f(v_1)(u'), \ldots, f(v_n)(u'))] \\ &= f^\sharp(r \cdot [u \mapsto \alpha(u(v_1), \ldots, u(v_n))]) \end{aligned}$$

$$\begin{aligned} f^\sharp(h_1 + h_2) &= f^\sharp([u \mapsto \alpha_1(u(v_{11}), \ldots, u(v_{1n_1})) + \alpha_2(u(v_{21}), \ldots, u(v_{2n_2}))]) \\ &= [u' \mapsto \alpha_1(f(v_{11})(u'), \ldots, f(v_{1n_1})(u')) + \alpha_2(f(v_{21})(u'), \ldots, f(v_{2n_2})(u'))] \\ &= f^\sharp([u \mapsto \alpha_1(u(v_{11}), \ldots, u(v_{1n_1}))]) + f^\sharp([u \mapsto \alpha_2(u(v_{21}), \ldots, u(v_{2n_2}))]) \\ &= f^\sharp(h_1) + f^\sharp(h_2). \end{aligned}$$

Proposition 3.3. $S_{\mathbb{T}}$ is a commutative coalgebra modality on $\mathsf{Mod}_R^{\mathrm{op}}$.

Proof. We first check the monad requirements and that $f^\sharp$ is a homomorphism of algebras. The monad requirements are given by:

$$\begin{aligned} (\eta)^\sharp([u \mapsto \alpha(u(v_1), \ldots, u(v_n))]) &= [u \mapsto \alpha(\eta(u)(v_1), \ldots, \eta(u)(v_n))] \\ &= [u \mapsto \alpha(u(v_1), \ldots, u(v_n))] \end{aligned}$$

$$\begin{aligned} f^\sharp(\eta(v)) &= f^\sharp([u \mapsto u(v)]) \\ &= [u \mapsto f(v)(u)] \\ &= f(v) \end{aligned}$$

$$
\begin{aligned}
g^\sharp(f^\sharp([u \mapsto \alpha(u(v1),\ldots,u(v_n))])) &= g^\sharp([u' \mapsto \alpha(f(v_1)(u'),\ldots,f(v_n)(u'))]) \\
&= g^\sharp([u' \mapsto \alpha(\beta_1(u'(v_{11}),\ldots,u'(v_{1m_1})),\ldots, \\
&\qquad \beta_m(u'(v_{m1}),\ldots,u'(v_{mn_m})))]) \\
&= [u'' \mapsto \alpha(\beta_1(g(v_{11})(u''),\ldots,g(v_{1m_1})(u'')),\ldots, \\
&\qquad \beta_m(g(v_{m1})(u''),\ldots,g(v_{mn_m})(u'')))] \\
&= [u'' \mapsto \alpha(g^\sharp([u' \mapsto \beta_1(u'(v_{11}),\ldots, \\
&\qquad u'(v_{1m_1}))](u'')),\ldots \\
&\qquad g^\sharp([u' \mapsto \beta_n(u'(v_{n1}),\ldots, \\
&\qquad u'(v_{nm_n}))](u''))] \\
&= [u'' \mapsto \alpha(g^\sharp(f(v_1))(u''),\ldots,g^\sharp(f(v_n))(u''))] \\
&= [(fg^\sharp)^\sharp]([u \mapsto \alpha(u(v_1),\ldots,u(v_n))]).
\end{aligned}
$$

The fact that $f^\sharp$ is an algebra homomorphism is given by checking the multiplication and the unit is preserved. The unit is the constant map $[u \mapsto e]$ and $f^\sharp$ applied to any constant map returns the same constant map (but with a different domain). Thus, the unit is preserved. For the multiplication we have

$$
\begin{aligned}
&f^\sharp([u \mapsto \alpha(u(v_1),\ldots,u(v_n))] \cdot [u \mapsto \beta(u(v_1'),\ldots,u(v_m'))]) \\
&\quad= f^\sharp([u \mapsto \alpha(u(v_1),\ldots,u(v_n)) \cdot \beta(u(v_1'),\ldots,u(v_m'))]) \\
&\quad= [u' \mapsto \alpha(f(v_1)(u'),\ldots,f(v_n)(u')) \cdot \beta(f(v_1')(u'),\ldots,f(v_m')(u'))] \\
&\quad= [u' \mapsto \alpha(f(v_1)(u'),\ldots,f(v_n)(u'))] \cdot [u' \mapsto \beta(f(v_1')(u'),\ldots,f(v_m')(u'))] \\
&\quad= f^\sharp([u \mapsto \alpha(u(v_1),\ldots,u(v_n))]) \cdot f^\sharp([u \mapsto \beta(u(v_1'),\ldots,u(v_m'))]).
\end{aligned}
$$

At this point, note that a modality requires only that the (co)multiplication of the (co)monad is a homomorphism, but when one combines this with naturality one gets that $f^\sharp := S_{\mathbb{T}}(f)\,\mu$ is a homomorphism. Conversely, if each $f^\sharp$ is a homomorphism, then $(f\eta)^\sharp = S_{\mathbb{T}}(f)$ is a homomorphism, showing that each free algebra is naturally a (co)algebra. Also, as the multiplication of the (co)monad is given by $(1)^\sharp$, it must be a homomorphism. In other words, $f^\sharp$ being a homomorphism is equivalent to the (co)multiplication (and unit) being natural and the (co)multiplication being a homomorphism.

An equivalent way to state the proposition, then, is to say that $(S_{\mathbb{T}}, (\_)^\sharp, \eta)$ is a monad on Mod_R for which each free object is naturally a commutative algebra and for which each $f^\sharp$ is an algebra homomorphism. □

There are various well-known options for the algebraic theory $\mathbb{T}$ over the field of real (or complex) numbers. For example, a fundamental example is the following.

Corollary 3.4. If $\mathbb{T}$ is the 'pure' theory of polynomial functions, then $S_{\mathbb{T}}(V)$ is the symmetric algebra monad $\mathrm{Sym}(V)$.

Another example suggested above is to take for real vector spaces *all* the infinitely differentiable functions $C^\infty(\mathbb{R}^n,\mathbb{R})$. There are many important subtheories of this: for example, one can take the subtheory of everywhere convergent power series (or of everywhere analytic functions).

Finally, we should connect these examples with our fundamental intuition that linear maps $!A \longrightarrow B$ are smooth maps $A \longrightarrow B$.

Proposition 3.5. If $\mathbb{T} := \mathbb{P}\mathrm{oly}$ is the 'pure' theory of polynomials:

$$\mathrm{Lin}(\mathbb{R}^m, S_{\mathbb{T}}(\mathbb{R}^n)) \cong \mathrm{Lin}(\mathbb{R}^m, \mathrm{Sym}(\mathbb{R}^n)) \cong \mathbb{P}\mathrm{oly}(n, m).$$

If $\mathbb{T} := \mathbb{S}\mathrm{mooth}$ is the smooth theory $C^\infty(\mathbb{R}^n, \mathbb{R})$,

$$\mathrm{Lin}(\mathbb{R}^m, S_{\mathbb{T}}(\mathbb{R}^n)) \cong \mathbb{S}\mathrm{mooth}(n, m).$$

Proof (sketch). The basic idea is that this really just reduces to the case $m = 1$ (in both cases), for which the result is obvious. □

Remember that if the maps seem to be 'backwards', we are working in the dual categories in these examples.

3.2. *From differential theory to deriving transformation*

In order to ensure there is a deriving transformation, one needs to require that the algebraic theory $\mathbb{T}$ has some further structure. We shall present this structure as the ability to take **partial derivatives**. Such a theory will allow us to extend the proof of Proposition 2.9 to a much more general setting. It is convenient for the development of these ideas to view the maps in $\mathbb{T}(n, 1)$ as terms $x_1, \ldots, x_n \vdash t$, and this allows us to suppose that there are 'partial differential' combinators:

$$\frac{x_1, \ldots, x_n \vdash t}{x_1, \ldots, x_n \vdash \partial_{x_i} t}.$$

We shall frequently just write $\partial_i t$ for the partial derivative with respect to the i^{th} coordinate. We then require the following properties of these combinators:

[pd.1] Identity: $\partial_x x = 1$
[pd.2] Constants: $\partial_x t = 0$ when $x \notin t$
[pd.3] Addition: $\partial_x(t_1 + t_2) = \partial_x t_1 + \partial_x t_2$
[pd.4] Multiplication: $\partial_x(t_1 \cdot t_2) = (\partial_x t_1) \cdot t_2 + t_1 \cdot (\partial_x t_2)$
[pd.5] Substitution: $\partial_z t[s/x] = (\partial_x t)[s/x] \cdot \partial_z s + (\partial_z t)[s/x]$.

A polynomial theory over a rig R with differential combinators is called a **differential theory** over R. Almost all the rules should be self-explanatory, except perhaps for **[pd.5]**, which is a combination of the chain rule and the copying rule natural for terms.

Given a differential theory $\mathbb{T}$ over R, we may define an induced co-deriving transformation $d: S_{\mathbb{T}}(V) \longrightarrow V \otimes S_{\mathbb{T}}(V)$ in Mod_R by

$$[u \mapsto \alpha(u(v_1), ., u(v_n))] \mapsto \sum_{i=1}^{n} v_i \otimes [u \mapsto \partial_i(\alpha)(u(v_1), \ldots, u(v_n))].$$

(Note the nullary case $[u \mapsto r] \mapsto 0$.) Now it is not immediately clear that this is even well defined, since, if $\alpha(u(v_1), ., u(v_n)) = \beta(u(v'_1), ., u(v'_m))$ for all u, we must show that (for all u)

$$\sum_{i=1}^{n} v_i \otimes [u \mapsto \partial_i(\alpha)(u(v_1), \ldots, u(v_n))] = \sum_{j=1}^{m} v'_j \otimes [u \mapsto \partial_j(\beta)(u(v'_1), \ldots, u(v'_m))].$$

We shall say V is **separated by functionals**† if whenever v' is not dependent on $v_1, \ldots, v_n$ in an R-module V, then for any functional u there is for each $r \in R$ a functional u_r such that $u(v_i) = u_r(v_i)$ but $u_r(v') = r$. When we are working enriched over Abelian groups it is necessary and sufficient to find a functional u_0 with $u_0(v_i) = 0$ and $u_0(v') = 1$. Given this condition, to obtain the u_r for u, one may set $u_r = r \cdot u_0 + u$; conversely, one may set $u_0 = u_{u(v')+1} - u$. We shall say Mod_R is separated by functionals if each $V \in \mathsf{Mod}_R$ is separated by functionals.

This is clearly a rather special property. It implies, in particular, that each finitely generated algebra has a well-defined dimension that is determined by the cardinality of the minimal spanning set. This certainly holds for all categories of vector spaces over fields. Thus, the reader may now essentially start thinking of modules over fields. This property is also sufficient to ensure the well-definedness of this transformation.

Lemma 3.6. If Mod_R is separated by functionals and $\mathbb{T}$ is a differential theory on R, then the co-deriving transformation is a well-defined natural transformation.

Proof. We first observe that if $[u \mapsto \alpha(u(v_1), \ldots, u(v_n))]$, we may assume that the set $\{v_1, \ldots, v_n\}$ is independent. For, if $v_1 = \sum_{j=2}^{n} r_j \cdot v_j$, we can adjust α to be

$$\alpha'(u(v_2), \ldots, u(v_n)) = \alpha\left(\sum_{j=2}^{n} r_j \cdot u(v_j), u(v_2), \ldots, u(v_n)\right).$$

Note that this adjustment does not change the definition of d since

$$\begin{aligned}
d([u \mapsto \alpha(u(v_1), ., u(v_n))]) &= \sum_{i=1}^{n} v_i \otimes [u \mapsto \partial_i \alpha(x)[u(v)/x]] \\
&= v_1 \otimes [u \mapsto \partial_1 \alpha(x)[u(v)/x]] \\
&\quad + \sum_{i=2}^{n} v_i \otimes [u \mapsto \partial_i \alpha(x)[u(v)/x]] \\
&= \sum_{j=2}^{n} r_j \cdot v_j \otimes [u \mapsto \partial_1 \alpha(x)[u(v)/x]] \\
&\quad + \sum_{j=2}^{n} v_j \otimes [u \mapsto \partial_j \alpha(x)[u(v)/x]] \\
&= \sum_{j=2}^{n} v_j \otimes ([u \mapsto (\partial_1 \alpha(x) r_j \\
&\quad + \partial_j \alpha(x)[\textstyle\sum r_j x_j / x_1])[u(v_j)/x_j] \\
&= \sum_{j=2}^{n} v_j \otimes [u \mapsto \partial_j \alpha'(x)[u(v_j)/x_j]].
\end{aligned}$$

Thus we may assume that in both α and β the elements $v_1, \ldots, v_n$ and $v'_1, \ldots, v'_m$ are independent, as otherwise we can do a replacement. Furthermore, by the same reasoning,

† By 'functionals' we mean 'linear functionals'.

we may replace the arguments of β by an expression in $v_1, \ldots, v_n$ whenever they are dependent on these elements. This gives a minimal independent set, which may have some extra points not in $v_1, \ldots, v_n$. However, using the separation property, we now know that β cannot depend on these points! Thus, β can be completely expressed in term of the points $v_1, \ldots, v_n$, and this shows that the map is well defined.

We also need to show that d_V is a map of R-modules. However, this is immediate from the properties of the partial derivatives. Finally, we need to establish naturality. For this we have

$$\begin{aligned} d(S_{\mathbb{T}}(f)([u \mapsto \alpha(u(v_1), \ldots, u(v_n))])) &= d([u \mapsto \alpha(u(f(v_1)), \ldots, u(f(v_n)))]) \\ &= \sum_{i=1}^{n} f(v_i) \otimes [u \mapsto \partial_i \alpha(u(f(v_1)), \ldots, u(f(v_n)))] \\ &= [f \otimes S_{\mathbb{T}}(f)] \sum_{i=1}^{n} v_i \otimes [u \mapsto \partial_i \alpha(u(v_1), \ldots, u(v_n))] \\ &= [f \otimes S_{\mathbb{T}}(f)](d([u \mapsto \alpha(u(v_1), \ldots, u(v_n))])). \qquad \square \end{aligned}$$

Proposition 3.7. If $\mathbb{T}$ is a differential theory over R, and Mod_R is separated by functionals, then $\mathsf{Mod}_R^{\mathrm{op}}$ becomes a differential category with respect to the algebra modality $(S_{\mathbb{T}}, (\_)^{\#}, \eta)$ on Mod_R and the induced co-deriving transformation.

Proof. It just remains to establish the properties of a differential combinator. The argument is the same as that used in Proposition 2.9, where we implicitly used familiar properties of partial derivatives. Here we do an explicit calculation to mimic that calculation based on the axiomatic structure of a differential theory. The point, of course, is that the S_∞ construction provides the formal support required to make this argument. One should think of $S_{\mathbb{T}}(R^n)$ as smooth real-valued functions. The various contortions in the definition occur for two reasons. First, one has to make this covariant; and second, it has to be defined on infinite-dimensional spaces. Once that is sorted out, the simpler proof goes through verbatim. In retrospect, the point of introducing polynomial theories and differential theories is to present a general abstract framework in which the proof can be carried out.

[d.1] For scalars we have $d(e(r)) = d([u \mapsto r]) = 0$.
[d.2] The copying rule gives

$$\begin{aligned} &[d\Delta]([u \mapsto \alpha(u(v_1), \ldots, u(v_n))], [u \mapsto \beta(u(v_1'), \ldots, u(v_m'))]) \\ &\quad = d([u \mapsto \alpha(u(v_1), \ldots, u(v_n)) \cdot \beta(u(v_1'), \ldots, u(v_m'))])) \\ &\quad = \sum_{i=1}^{n} v_i \otimes [u \mapsto \partial_{u(v_i)} \alpha(u(v_1), \ldots, u(v_n)) \cdot \beta(u(v_1'), \ldots, u(v_m'))] \\ &\qquad + \sum_{j=1}^{m} v_j' \otimes [u \mapsto \alpha(u(v_1), \ldots, u(v_n)) \cdot \partial_{u(v_j')} \alpha(u(v_1'), \ldots, u(v_m'))] \\ &\quad = [1 \otimes \Delta(d \otimes 1) + (1 \otimes \Delta)(1 \otimes c \otimes 1)(1 \otimes d)] \\ &\qquad\qquad ([u \mapsto \alpha(u(v_1), \ldots, u(v_n))], [u \mapsto \beta(u(v_1'), \ldots, u(v_m'))]). \end{aligned}$$

[d.3] Linearity is $d(\eta(x)) = d([u \mapsto u(x)]) = x \otimes 1$.
[d.4] Chaining requires the following calculation:

$$
\begin{aligned}
& d(1^\sharp([u \mapsto \alpha(u([v \mapsto \beta_1(v(x_{11}), \ldots, v(x_{1m_1}))]), \ldots, u([v \mapsto \beta_n(v(x_{n1}), \ldots, v(x_{nm_n}))]))])) \\
&= d([v \mapsto \alpha(\beta_1(v(x_{11}), \ldots, v(x_{1m_1})), \ldots, \beta_n(v(x_{n1}), \ldots, v(x_{nm_n}))]) \\
&= \sum_{i=1}^{n} \sum_{j=1}^{m_i} x_{ij} \otimes [v \mapsto \partial_{ij}\alpha(\beta_1(v(x_{11}), \ldots, v(x_{1m_1})), \ldots, \beta_n(v(x_{n1}), \ldots, v(x_{nm_n}))] \\
&= \sum_{i=1}^{n} \sum_{j=1}^{m_i} x_{ij} \otimes [v \mapsto \partial_j(\beta_i)(v(x_{i1}), \ldots, v(x_{im_i})) \\
&\qquad \cdot \partial_i(\alpha)(\beta_1(v(x_{11}), \ldots, v(x_{1m_1})), \ldots, \beta_n(v(x_{n1}), \ldots, v(x_{nm_n})))] \\
&= (1 \otimes \Delta)(\sum_{i=1}^{n} \sum_{j=1}^{m_i} x_{ij} \otimes [v \mapsto \partial_j(\beta_i)(v(x_{i1}), \ldots, v(x_{im_i}))] \\
&\qquad \otimes [u \mapsto \partial_i(\alpha)(\beta_1(u(x_{11}), \ldots, u(x_{1m_1})), \ldots, \beta_n(u(x_{n1}), \ldots, u(x_{nm_n})))]) \\
&= (1 \otimes \Delta)((d \otimes 1^\sharp)(\sum_{i=1}^{n} [v \mapsto \beta_1(v(x_{11}), \ldots, v(x_{1m_1}))] \\
&\qquad \otimes [u \mapsto \partial_i(\alpha)(u([v \mapsto \beta_1(v(x_{11}), \ldots, v(x_{1m_1}))]), \ldots \\
&\qquad\qquad \ldots, u([v \mapsto \beta_n(v(x_{n1}), \ldots, v(x_{nm_n}))]))]))) \\
&= (1 \otimes \Delta)((d \otimes 1^\sharp)(d([u \mapsto \alpha(u([v \mapsto \beta_1(v(x_{11}), \ldots, v(x_{1m_1}))]), \\
&\qquad\qquad \ldots, u([v \mapsto \beta_n(v(x_{n1}), \ldots, v(x_{nm_n}))]))])))\,.
\end{aligned}
$$

□

At this stage we have shown how to incorporate notions of differentiability into a category of vector spaces. Applying these results to other settings does require that one can prove that all objects are separable by functionals. There is a further subtle aspect to these settings that must be remembered: the finite-dimensional support of the elements of $S_{\mathbb{T}}(V)$ builds in a certain finite dimensionality to the notion of differentiability.

4. Differential storage categories

A storage modality on a symmetric monoidal category is a comonad that is symmetric monoidal and has each cofree object symmetrical monoidally naturally a commutative comonoid so that the comultiplication and elimination map are also morphisms of the coalgebras of the comonad. When the category also has products, these rather technical conditions give what we shall call a **storage category**. In this case the category has the **storage** (or Seely) **isomorphisms**, and it is this fact that we wish to exploit below. The storage isomorphisms are natural isomorphisms $s_\times : \,!A \otimes \,!B \longrightarrow \,!(A \times B)$ that also, importantly, hold in the nullary case $s_1 : \top \longrightarrow \,!1$.

Regarding terminology, storage categories are exactly the same as Bierman's notion of a 'linear category' (Bierman 1995). We have chosen not to follow his terminology here as the notion of a linear map (in the context of maps between vector spaces) has a different connotation in the theory of differentiation. This paper involves a number of modalities and we have chosen nomenclature that corresponds to the appropriate modality involved: a 'storage category' has a storage modality. These have appeared frequently in the literature, especially when the category is closed, and often with different names. We called them 'bang's in (Blute *et al.* 1996). Recently, they have been called 'linear exponential monads' in Hyland and Schalk (2003).

4.1. *Basics for storage categories*

Definition 4.1. A storage modality on a symmetric monoidal category is a comonad $(\,!\,, \delta, \epsilon)$ that is symmetric monoidal and has each cofree object naturally a commutative comonoid $(\,!\,A, \Delta, e)$. In addition, the comonoid structure must be a morphism for the coalgebras for the comonad.

Recall that a coalgebra (A, v) for the comonad is an object together with a map $v : A \longrightarrow\, !\,A$ such that $v\epsilon = 1$ and $v\delta = v\ !\,v$. This means that given coalgebras (A, v) and (A', v'), the tensor product of these is formed as $(A \otimes A', (v \otimes v')m_\otimes)$. For any symmetric monoidal comonad this makes the (Eilenberg–Moore) category of coalgebras a symmetric monoidal category.

We first recall the following proposition (see Schalk (2004)).

Proposition 4.2. A symmetric monoidal category has a storage modality if and only if the induced symmetric tensor on the category of coalgebras for the comonad is a product.

In particular, this means that we have coalgebra morphisms $\Delta : (\,!\,A, \delta) \longrightarrow (\,!\,A \otimes\, !\,A, (\delta \otimes \delta)m_\otimes)$ that must be an associative multiplication with counit $e : (\,!\,A, \delta) \longrightarrow (\top, m_\top)$. These give rise to the rather technical requirements above.

This is a useful result, as the symmetric algebra monad on Mod_R is always symmetric comonoidal and has the induced tensor a coproduct on its algebras. Therefore we have the following corollary.

Corollary 4.3. For any commutative rig R, the opposite of its category of modules, $\mathsf{Mod}_R^{\mathrm{op}}$, has a storage modality given by the symmetric algebra monad on Mod_R.

A primary example of this (in addition to the ever-present symmetric algebra functor on vector spaces) is the storage modality on suplattices described earlier. The duality twist required to get this example is explained by this observation.

Definition 4.4. A **storage category** is a symmetric monoidal category possessing products and a storage modality.

When products are present a crucial observation is given by the following proposition from Bierman (1995).

Proposition 4.5. A storage category possesses the storage isomorphisms

$$s_\times : \,!\,A \otimes\, !\,B \longrightarrow\, !\,(A \times B) \qquad s_1 : \top \longrightarrow\, !\,1$$

and, furthermore,

$$\begin{array}{ccc} & !X & \\ {\scriptstyle\Delta}\swarrow & & \searrow{\scriptstyle !(\Delta_\times)} \\ !X\otimes !X & \xrightarrow[s_\times]{} & !(X\times X) \end{array} \qquad \begin{array}{ccc} & !X & \\ {\scriptstyle e}\swarrow & & \searrow{\scriptstyle !(\langle\rangle)} \\ \top & \xrightarrow[s_1]{} & !1 \end{array} ,$$

commute.

The storage isomorphisms are not arbitrary maps; they are given in a canonical way by the structure of the setting.

$$s_\times \;=\; !X\otimes !X \xrightarrow{\delta\otimes\delta} !!X\otimes !!X \xrightarrow{m_\otimes} !(!X\otimes !X) \xrightarrow{!(\langle\epsilon\otimes e, e\otimes\epsilon\rangle)} !(X\times X)$$

$$s_1 \;=\; \top \xrightarrow{m_\top} !\top \xrightarrow{!(\langle\rangle)} !1$$

where the inverses are

$$s_\times^{-1} \;=\; !(X\times X) \xrightarrow{\Delta} !(X\times X)\otimes !(X\times X) \xrightarrow{!\pi_0\otimes !\pi_1} !X\otimes !X$$

$$s_1^{-1} \;=\; !1 \xrightarrow{e} \top .$$

The Kleisli category of a storage modality is the subcategory of cofree coalgebras in the Eilenberg–Moore category. From Proposition 4.2 we know that the Eilenberg–Moore category has products given by the tensor. In general, the tensor of two cofree objects is not itself a cofree object, but the storage isomorphism ensures it is equivalent to a cofree object. This gives the following corollary.

Corollary 4.6. The coKleisli category $\mathbb{X}_!$ of the modality of a storage category $\mathbb{X}$, viewed as a subcategory of the Eilenberg–Moore category, is closed under the induced tensor of the latter. Moreover, if $\mathbb{X}$ has products, they give products in $\mathbb{X}_!$ as well.

We can also record the following observation.

Proposition 4.7. For any storage category $\mathbb{X}$ the adjunction between $\mathbb{X}$ and the coKleisli category $\mathbb{X}_!$ is a monoidal adjunction.

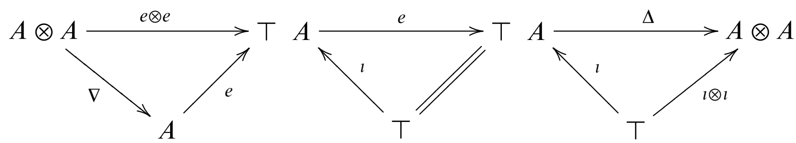
The monoidal structure of $\mathbb{X}_!$ is the product. Recall that in a monoidal adjunction the left adjoint is necessarily iso-monoidal (that is, strong). In this case the left adjoint is the underlying category and the iso-monoidal transformation is given by the storage isomorphism. The monoidal map for the right adjoint amounts to a coKleisli map $X\times Y \xrightarrow{m} X\otimes Y$, which in $\mathbb{X}$ is the composite map

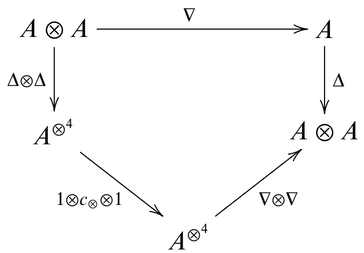
$$!(X\times Y) \xrightarrow{s^{-1}} !(X)\otimes !(Y) \xrightarrow{\epsilon\otimes\epsilon} X\otimes Y .$$

4.2. *Bialgebra modalities*

Our next step toward considering differential categories with storage is to consider the effect of requiring a storage category to be additive. It is well known that in any additive category if there are either products or coproducts, they must coincide and be biproducts. One way to describe biproducts is as a natural commutative bialgebra structure on a symmetric tensor.

Recall (for example, from Kassel (1995)) that an object A in a symmetric monoidal category is a (commutative) **bialgebra** when it has both a (cocommutative) comonoid (A, Δ, e) and a (commutative) monoid (A, ∇, ι) structure such that all the triangles and the pentagon in





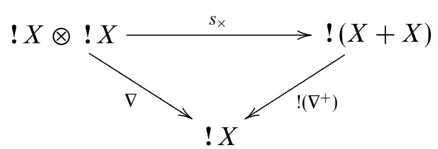
commute.

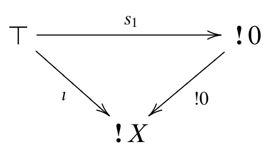
Definition 4.8. A **bialgebra modality** is a comonad $(\,!\,, \delta, \epsilon)$ such that each $!A$ is in fact naturally a bialgebra, $(\,!A, \nabla, \iota, \Delta, e)$, such that δ is a homomorphism of coalgebra structures (but not necessarily of the algebra structures), and such that ϵ satisfies the following equations $\iota\epsilon = 0$ and $\nabla\epsilon = \epsilon \otimes e + e \otimes \epsilon$.

The following proposition is immediate.

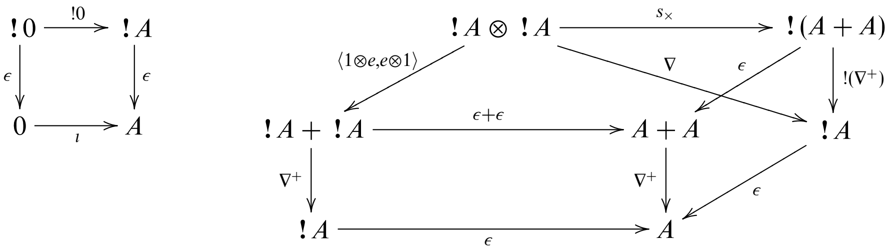
Proposition 4.9. In any additive storage category, each cofree object is naturally a commutative bialgebra where the canonical bialgebra structure on the biproduct is given by transporting the bialgebra structure onto the tensor using the storage isomorphism. Furthermore, the storage modality is in fact a bialgebra modality.

Proof (sketch). As stated, $\nabla\colon !A \otimes !A \longrightarrow !A$ and $\iota\colon \top \longrightarrow !A$ are defined by the following commutative diagrams (note that $+$ is $\times$ and 0 is 1):





To see the ϵ equations, note that they essentially lift from the biproduct structure *via* the storage isomorphisms:



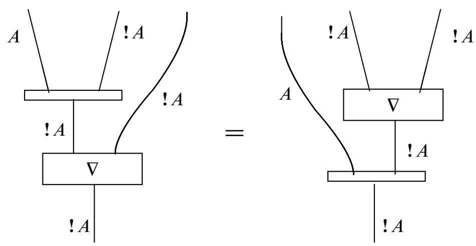
And note that $\nabla^+ = \pi_1 + \pi_2$, so $\epsilon(\pi_1 + \pi_2) = \epsilon\pi_1 + \epsilon\pi_2 = \,!\pi_1\epsilon + \,!\pi_2\epsilon$, so we get $\nabla\epsilon = \epsilon \otimes e + e \otimes \epsilon$. □

4.3. *Differential storage categories and the differential calculus*

If an additive storage category has a differential combinator, it is natural to expect it to interact with the multiplication $\nabla\colon !A \otimes !A \longrightarrow !A$ in a well-defined manner.

[∇–rule] $(d \otimes 1)\nabla = (1 \otimes \nabla)d$:

$$\begin{array}{ccc} A \otimes !A \otimes !A & \xrightarrow{1\otimes\nabla} & A \otimes !A \\ {\scriptstyle d_A\otimes 1}\downarrow & & \downarrow{\scriptstyle d_A} \\ !A \otimes !A & \xrightarrow[\nabla]{} & !A \end{array}$$



Definition 4.10. A **differential storage category** is an additive storage category with a deriving transformation such that the ∇-rule is satisfied.

We observe that in this setting, whenever we have a deriving transformation, we obtain a natural transformation

$$\eta\colon A \longrightarrow !A \;=\; A \xrightarrow{1\otimes\iota} A \otimes !A \xrightarrow{d_A} !A\,.$$

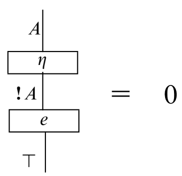
Thomas Ehrhard and Laurent Regnier have introduced a syntax they refer to as ‘differential interaction nets’ (Ehrhard and Regnier 2005). Their formalism makes explicit the fact that $!X$ has a bialgebra structure and presents differentiation as a map $X \multimap !X$, indeed, as the η map above. They also have rewriting rules similar to the equations on circuits presented here, apart from those involving 'promotion', which their system did not include. That additional structure on ! had been considered in Ehrhard (2001). However, their formalism demands the presence of considerably more structure, which includes the requirement of being monoidal closed (actually $*$-autonomous). Our basic example of polynomial functions is not even closed. To make a comparison of the two approaches easier, we shall now reformulate the ideas of Ehrhard and Regnier into a first-order setting; we shall call the resulting notion a 'categorical model of the differential calculus'.

Definition 4.11. A categorical model of the **differential calculus** is an additive category with biproducts with a bialgebra modality consisting of a comonad $(\,!, \delta, \epsilon)$ such that each object $!X$ has a natural bialgebra structure $(\,!X, \nabla_X, \iota_X, \Delta_X, e_X)$, and a natural map $\eta_X : X \longrightarrow\, !X$ satisfying the following coherences:

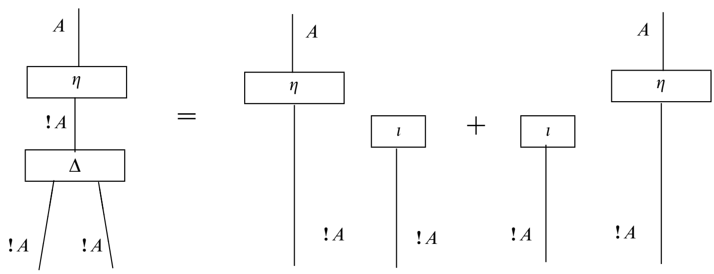
[dC.1] $\eta e = 0$
[dC.2] $\eta\Delta = \eta \otimes \iota + \iota \otimes \eta$
[dC.3] $\eta\epsilon = 1$
[dC.4] $(\eta \otimes 1)\nabla\delta = (\eta \otimes \Delta)((\nabla\eta) \otimes \delta)\nabla$.

We may present these as circuit equations by:

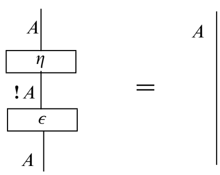
[dC.1]



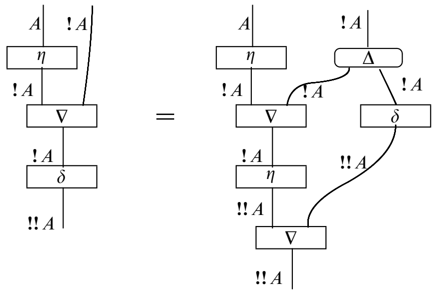
[dC.2]



[dC.3]



[dC.4]



An additive storage category could provide a variety of models for the differential calculus: each corresponds to specifying a deriving transformation satisfying the ∇-rule. We first observe a more general result: that a model of the differential calculus always gives rise to deriving transformation (whether it is on a storage modality or not).

Theorem 4.12. A model of the differential calculus is equivalent to a differential category with biproducts whose coalgebra modality is a bialgebra modality satisfying the ∇-rule.

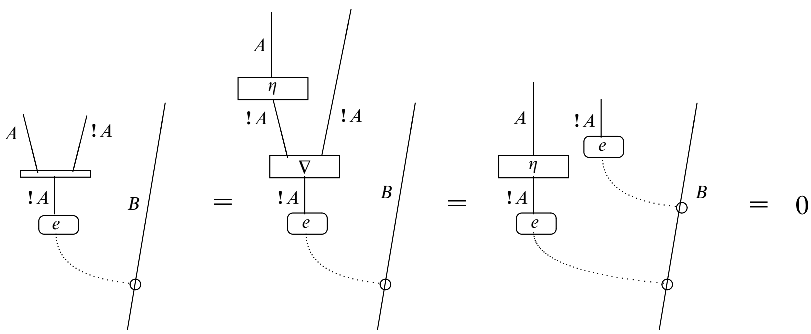
Corollary 4.13. Models of the differential calculus on additive storage categories correspond precisely to differential storage categories, that is, to deriving transformations on these categories satisfying the ∇-rule.

Proof of Theorem 4.12. Given a model of the differential calculus, we obtain a differential category by defining d_X by $d_X = (\eta_X \otimes 1)\nabla$. There are four equations to verify:

[d.1] The rule for constant maps is verified by

$$d_A e_A = (\eta \otimes 1)\nabla e = (\eta \otimes 1)(e \otimes e) = (0 \otimes e) = 0.$$

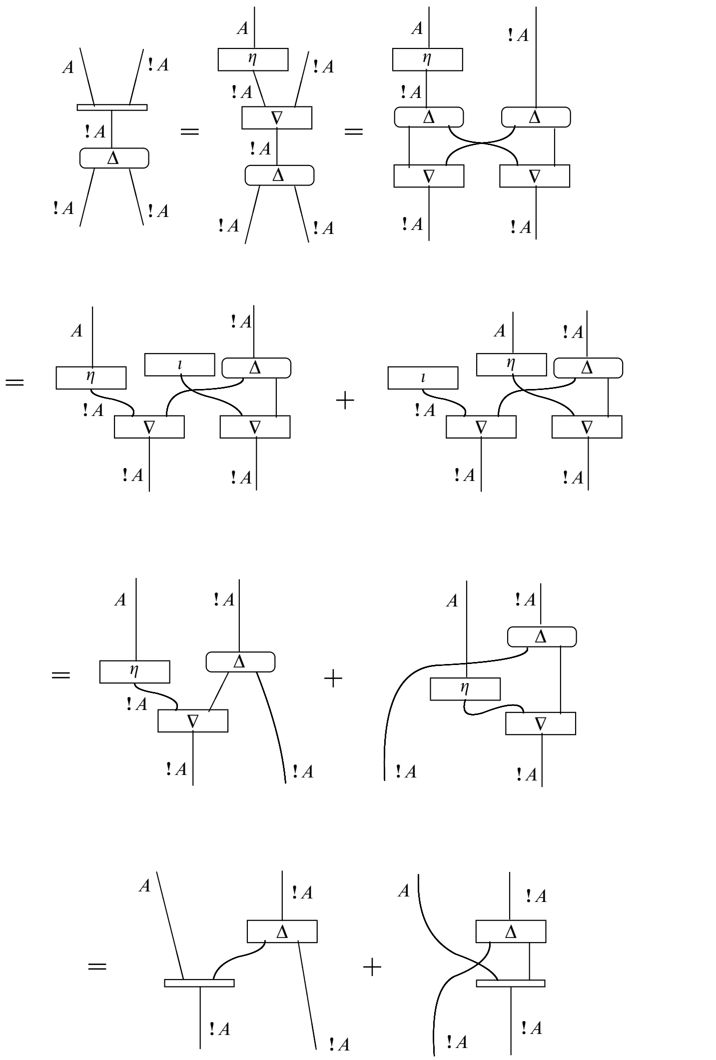
As a circuit calculation this is



[d.2] The product rule for the deriving transformation is given by

$$
\begin{aligned}
d_A\Delta &= (\eta \otimes 1)\nabla\Delta \\
&= (\eta \otimes 1)(\Delta \otimes \Delta)(1 \otimes c_\otimes \otimes 1)(\nabla \otimes \nabla) \\
&= ((\eta \otimes \iota) + (\iota \otimes \eta)) \otimes \Delta)(1 \otimes c_\otimes \otimes 1)(\nabla \otimes \nabla) \\
&= (\eta \otimes \iota \otimes \Delta)(1 \otimes c_\otimes \otimes 1)(\nabla \otimes \nabla) + (\iota \otimes \eta \otimes \Delta)(1 \otimes c_\otimes \otimes 1)(\nabla \otimes \nabla) \\
&= (\eta \otimes \Delta)(1 \otimes 1 \otimes \iota \otimes 1)(\nabla \otimes \nabla) + (\eta \otimes \Delta)(\iota \otimes c_\otimes \otimes 1)(\nabla \otimes \nabla) \\
&= (\eta \otimes \Delta)(\nabla \otimes 1) + (1 \otimes \Delta)(c_\otimes \otimes 1)(1 \otimes ((\eta \otimes 1)\nabla)) \\
&= (1 \otimes \Delta)(d_A \otimes 1) + (\Delta \otimes 1)(c_\otimes \otimes 1)(1 \otimes d_A).
\end{aligned}
$$

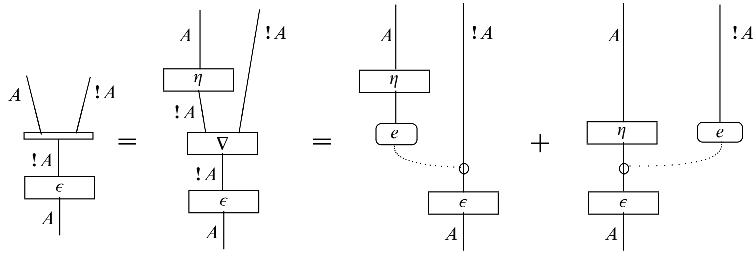
In circuits this is

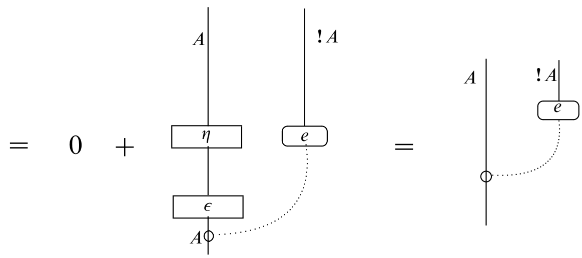


[d.3] Linearity is given by the following calculation:

$$\begin{aligned} d_A \epsilon_A &= (\eta \otimes 1)\nabla\epsilon \\ &= (\eta \otimes 1)(e \otimes \epsilon + \epsilon \otimes e) \\ &= (\eta \otimes 1)(e \otimes \epsilon) + (\eta \otimes 1)(\epsilon \otimes e) \\ &= (0 \otimes \epsilon) + ((\eta\epsilon) \otimes e) \\ &= 1 \otimes e. \end{aligned}$$

Using circuits this is



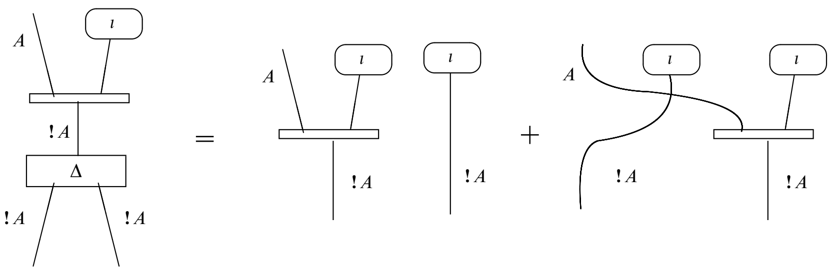


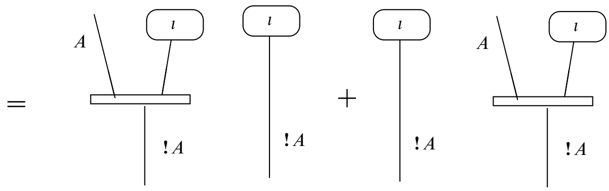
[d.4] Chaining is immediate from **[dC.4]**.

The associativity of ∇ provides the ∇-rule for the deriving transformation.

To prove the converse, we need to show that given a differential category with a bialgebra modality satisfying the ∇-rule, we can define η as $(1 \otimes \iota)\, d_A$ to give us a model of the differential calculus. Again, there are four equations to verify. These are straightforward; we shall present the proofs for the two cases that are not entirely trivial *via* circuit calculations.

[dC.1] This is obvious, since $de = 0$.
[dC.2]

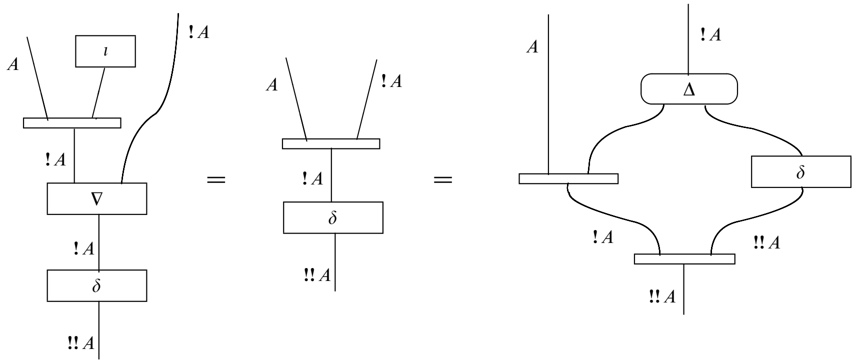




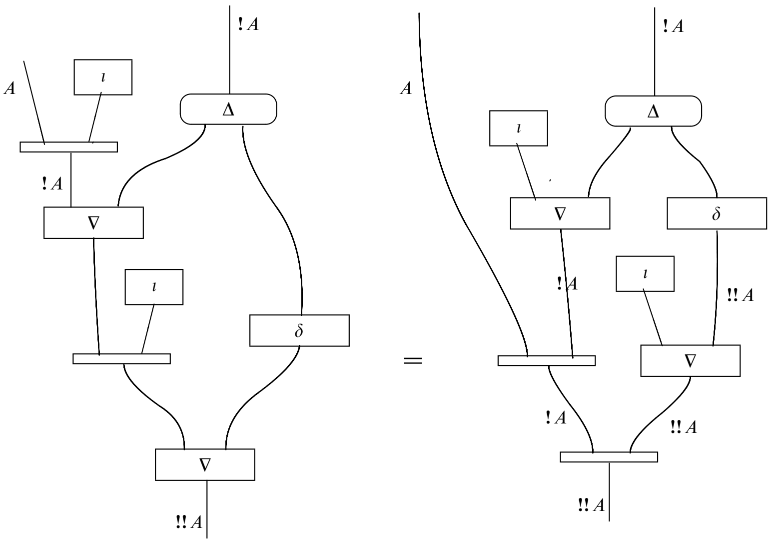
[dC.3] This is obvious, since $\iota e = 1$.

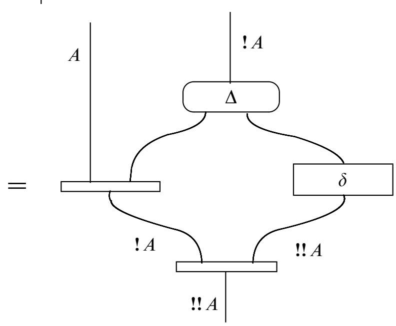
[dC.4] We reduce each side of the equation to the same circuit (in fact, the circuit corresponding to $D(\delta)$); note the use of the ∇-rule (several times):

LHS =



RHS =





Finally, the deriving transformation induced by the differential calculus produced from such a deriving transformation is just the original deriving transformation. Conversely, the η induced by the deriving transformation must reduce to η when the deriving transformation was induced by the differential calculus. For this we have the following calculations:

$$\begin{aligned}(\eta \otimes 1)\nabla &= (((1 \otimes \imath)d) \otimes 1)\nabla \\ &= (1 \otimes \imath \otimes 1)(1 \otimes \nabla)d \\ &= d \\ (1 \otimes \imath)d &= (1 \otimes \imath)(\eta \otimes 1)\nabla \\ &= (\eta \otimes \imath)\nabla \\ &= \eta.\end{aligned}$$

□

5. Concluding remarks

One of the goals of this work has been to establish a categorical framework for differentiable structures following the approach suggested by Ehrhard. While his approach to this matter has been our basic inspiration, these matters have in fact been the subject of research for quite some time. Without trying to be historically complete, we should mention the early work on the subject by Charles Ehresmann, in particular, Ehresmann (1959). This and other related papers are collected in Ehresmann (1980–1983). Ehresmann considered several categories of smooth structures, and stressed the importance of internal categories therein. He also considered extensions to the category of manifolds, which would have more limits and colimits.

We are currently working on a sequel to this paper whose aim is to give an abstract characterisation of those categories that arise as coKleisli categories of differential categories (Blute *et al.* 2006). Using these ideas, we believe it is possible to reproduce Ehresmann's context from ours, and we intend this to be the subject of a further sequel to this work.

Various approaches to building cartesian closed categories of smooth structures have also been suggested, and it would also be interesting to know to what extent these constructions are applicable to our general notion of differentiation. In particular, both the *convenient vector spaces* of Frolicher and Krieg (1988) and the diffeological spaces of Iglesias-Zemmour (2006), whose references give further historical information, seem worth investigating in this regard.

There is also a considerable body of work concerning the development of differential structures in monoidal categories, especially braided monoidal categories, in particular Woronowicz (1989), Majid (1993) and Bespalov (1997)[†]. A basic goal of this work is to develop an abstract version of de Rham cohomology by finding differential graded algebras in these categories. It would be interesting to understand how this work is related to the work presented in the current paper.

[†] A search of the archives at `xxx.lanl.gov` will turn up many other references.

Acknowledgements

All three authors wish to acknowledge NSERC Canada for the support we receive for our research.

We also wish to thank the University of Ottawa (for the support the second author received while on sabbatical there) and PIMS and John MacDonald (for the support we all received when finishing the paper at UBC).

We thank the anonymous referees for their detailed reports, which contained many helpful comments and suggestions.

References

Barr, M. B. (1979) ∗-Autonomous Categories. *Springer-Verlag Lecture Notes in Mathematics* **752**.

Bespalov, Y. and Drabant, B. (1997) Differential Calculus in Braided Abelian Categories. (Preprint, available at `http://arxiv.org/abs/q-alg/9703036`.)

Bierman, G. (1995) What is a categorical model of intuitionistic linear logic? In: Proceedings of the Second International Conference on Typed Lambda Calculus and Applications, Edinburgh, Scotland. *Springer-Verlag Lecture Notes in Computer Science* **902** 73–93.

Blute, R. F., Cockett, J. R. B. and Seely, R. A. G. (1996) **!** and **?**: Storage as tensorial strength. *Mathematical Structures in Computer Science* **6** 313–351.

Blute, R. F., Cockett, J. R. B. and Seely, R. A. G. (2006) Cartesian differential categories (in preparation).

Blute, R. F., Cockett, J. R. B., Seely, R. A. G. and Trimble, T. (1997) Natural deduction and coherence for weakly distributive categories. *Journal of Pure and Applied Algebra* **113** 229–296.

Cockett, J. R. B. (1993) Introduction to distributive categories. *Mathematical Structures in Computer Science* **3** 277–308.

Ehresmann, C. (1959) Catégories topologiques et catégories différentiables. In: *Coll. Géom. Diff. Globale (Bruxelles, 1958)*, Centre Belge Rech. Math. Louvain 137–150.

Ehresmann, C. (1980–1983) *Charles Ehresmann: Oeuvres complètes et commentées.* Edited by A. C. Ehresmann and published as supplements to *Cahiers de Topologie et Géométrie Différentielle, Amien* **21**–**24**.

Ehrhard, T. (2001) On Köthe sequence spaces and linear logic. *Mathematical Structures in Computer Science* **12** 579–623.

Ehrhard, T. (2005) Finiteness spaces. *Mathematical Structures in Computer Science* **15** 615–646.

Ehrhard, T. and Regnier, L. (2003) The differential λ-calculus. *Theoretical Computer Science* **309** (1–3) 1–41.

Ehrhard, T. and Regnier, L. (2005) Differential interaction nets. Invited paper for Workshop on Logic, Language, Information and Computation (WoLLIC). *Electronic Notes in Theoretical Computer Science* **123**.

Frolicher, A. and Kriegl, A. (1988) *Linear Spaces and Differentiation Theory*, Wiley Publishing.

Girard, J.-Y. (1987) Linear Logic. *Theoretical Computer Science* **50** 1–102.

Hyland, M. and Schalk, A. (2003) Glueing and Orthogonality for Models of Linear Logic. *Theoretical Computer Science* **294** 183–231.

Iglesias-Zemmour, P. (2006) *Introduction to Diffeology*. (Online textbook available at `http://www.umpa.ens-lyon.fr/~iglesias`.)

Kassel, C. (1995) *Quantum Groups*, Graduate Texts in Mathematics **155**, Springer-Verlag.

Lang, S. (2002) *Algebra*, Graduate Texts in Mathematics, Springer-Verlag.

Mac Lane, S. (1971) *Categories for the Working Mathematician*, Graduate Texts in Mathematics, Springer-Verlag.

Majid, S. (1993) Free Braided Differential Calculus, Braided Binomial Theorem and the Braided Exponential Map. *Journal of Mathematical Physics* **34** 4843–4856.

Manes, E. G. (1976) *Algebraic Theories*, Graduate Texts in Mathematics **26**, Springer Verlag.

Schalk, A. (2004) What is a categorical model of linear logic? (Course notes.)

Seely, R. A. G. (1989) Linear logic, $*$-autonomous categories and cofree coalgebras. In: Gray, J. and Scedrov, A. (eds.) Categories in Computer Science and Logic. *Contemporary Mathematics* **92**.

Woronowicz, S. L. (1989) Differential calculus on compact matrix pseudogroups (quantum groups). *Comm. Math. Phys.* **122** 125–170.