Jacobians and Gradients
for Cartesian Differential Categories

Jean-Simon Pacaud Lemay[*]

Mathematics and Computer Science Department
Mount Allison University
Sackville, New Brunswick, Canada

`jsplemay@gmail.com` [†]

Cartesian differential categories come equipped with a differential combinator that formalizes the directional derivative from multivariable calculus. Cartesian differential categories provide the categorical semantics of the differential λ-calculus and have also found applications in causal computation, incremental computation, game theory, differentiable programming, and machine learning. There has recently been a desire to provide a (coordinate-free) characterization of Jacobians and gradients in Cartesian differential categories. One's first attempt might be to consider Cartesian differential categories which are Cartesian closed, such as models of the differential λ-calculus, and then take the curry of the derivative. Unfortunately, this approach excludes numerous important examples of Cartesian differential categories such as the category of real smooth functions. In this paper, we introduce linearly closed Cartesian differential categories, which are Cartesian differential categories that have objects of linear maps, a bilinear evaluation map, and the ability of currying maps which are linear in their second argument. As such, the Jacobian of a map is defined as the curry of its derivative. Many well-known examples of Cartesian differential categories are linearly closed, such as, in particular, the category of real smooth functions. We also explain how a Cartesian closed differential category is linearly closed if and only if a certain linear idempotent on the internal hom splits. To define the gradient of a map, one requires the ability to take the transpose of the Jacobian, which can be done in a Cartesian reverse differential category. Thus, we define the gradient of a map to be the curry of its reverse derivative and show this equals the transpose of its Jacobian. We also explain how a linearly closed Cartesian reverse differential category is a precisely a linearly closed Cartesian differential category with an appropriate notion of transpose.

1 Introduction

Cartesian differential categories, introduced by Blute, Cockett, and Seely in [5], come equipped with a differential combinator D which provides a categorical axiomatization of the directional derivative from multivariable calculus, and so for every map $A \xrightarrow{f} B$ produces its derivative $A \times A \xrightarrow{\mathsf{D}[f]} B$. There is no shortage of examples of Cartesian differential categories in the literature, but arguably the most important example is SMOOTH, the category of Euclidean spaces $\mathbb{R}^n$ and real smooth functions between them, where the differential combinator, in this case, is precisely the classical directional derivative. Another important class of examples are the Cartesian closed differential categories [8, 9, 19], which provide the categorical semantics of Ehrhard and Regnier's differential λ-calculus [14]. Cartesian (closed) differential categories have found numerous applications in computer science such as being picked up by Katsumata and Sprunger in their work on causal computations [21], by Abadi and Plotkin in their work

---

on differentiable programming languages [1], by Alvarez-Picallo and Ong in their work on incremental computation [2], by Laird, Manzonetto, and McCusker on their work in game theory [17], and most recently for their usage in machine learning with the introduction of Cartesian *reverse* differential categories [10, 12] which have been shown to be a suitable setting for reverse gradient descent by Cruttwell, Gavranović, Ghani, Wilson, and Zanasi in [13, 23].

An important concept in differential calculus is the Jacobian matrix, where recall that for a smooth function $\mathbb{R}^n \xrightarrow{F = \langle f_1, \ldots, f_n \rangle} \mathbb{R}^m$, its Jacobian matrix at point $\vec{x} \in \mathbb{R}^n$ is given by the $n \times m$ matrix $\mathbf{J}(F)(\vec{x})$ whose coordinates are the partial derivatives of f_i evaluated at $\vec{x}$. The Jacobian matrix at $\vec{x}$'s associated $\mathbb{R}$-linear function is called the total derivative of F, and evaluating this linear function at $\vec{y}$ results in the derivative $\mathsf{D}[F](\vec{x}, \vec{y})$. Thus, the Jacobian of F can be interpreted as a map $\mathbb{R}^n \xrightarrow{\mathbf{J}(F)} \mathsf{LIN}(\mathbb{R}^n, \mathbb{R}^m)$, where the codomain is the vector spaces of $\mathbb{R}$-linear functions from $\mathbb{R}^n$ to $\mathbb{R}^m$, and can therefore be understood as sort of curry $\mathbb{R}^n \times \mathbb{R}^n \xrightarrow{\mathsf{D}[F]} \mathbb{R}^m$. Interpreting the Jacobian in this manner shows how the Jacobian is a special case of both the Fréchet derivative and the Gateaux derivative. For a smooth function $\mathbb{R}^n \xrightarrow{f} \mathbb{R}$, its gradient at $\vec{x} \in \mathbb{R}^n$ is the transpose of its Jacobian at $\vec{x}$, $\nabla(f)(\vec{x}) = \mathbf{J}(F)(\vec{x})^\mathsf{T} \in \mathbb{R}^n$. Since $\mathsf{LIN}(\mathbb{R}, \mathbb{R}^n) \cong \mathbb{R}^n$, the gradient can therefore be interpreted as a map $\mathbb{R}^n \xrightarrow{\nabla(f)} \mathsf{LIN}(\mathbb{R}, \mathbb{R}^n)$.

The notions of Jacobians and gradients have yet to be formally defined in a Cartesian differential category. Such concepts would be highly desirable, specifically if one wishes to formalize machine learning algorithms in Cartesian differential categories. This need for Jacobians and gradients is expressed by Katsumata and Sprunger where they state following in the conclusion of their paper [21]: "Though we would like to say our abstract treatment of differentiation can be used directly by machine learning practitioners, it appears this is not the case yet. The derivative of a morphism in a Cartesian differential category is not the same as having an explicit Jacobian or gradient. A gradient can be recovered from this morphism by applying it to all the basis vectors, but when there are millions of parameters in a machine learning model, this idea is computationally disastrous. We think that by adding some structure to Cartesian differential categories, such as a designated closed subcategory, we could give a theoretical treatment allowing for more explicit representation of Jacobians." This highlights a clear need for axiomatizing (coordinate-free) Jacobians and gradients in the context of Cartesian differential categories.

Based on the above discussion, and as suggested by Katsumata and Sprunger, one's first attempt might be to consider defining Jacobians in a Cartesian closed differential category and define the Jacobian of a map $A \xrightarrow{f} B$ as the curry of its derivative $A \times A \xrightarrow{\mathsf{D}[f]} B$, so $A \xrightarrow{\lambda(\mathsf{D}[f])} [A, B]$. While this is a very reasonable and promising idea where one can get quite far in extracting the main properties of the Jacobian, there is a flaw. The Cartesian closed differential category approach would exclude numerous examples of Cartesian differential categories, specifically those with a "finite-dimensional flavour" and many machine learning related models. In particular, SMOOTH is not Cartesian closed since, intuitively, the set of smooth functions from $\mathbb{R}^n$ to $\mathbb{R}^m$ is not a finite-dimensional $\mathbb{R}$-vector space and therefore not isomorphic to a Euclidean space $\mathbb{R}^k$. Thus another approach is required. Luckily, note that the codomain of the Jacobian is $\mathsf{LIN}(\mathbb{R}^n, \mathbb{R}^m)$, which is isomorphic to $\mathbb{R}^{nm}$, and so can therefore be interpreted as an object in SMOOTH. We may generalize this idea in numerous Cartesian differential categories since there is a notion of linearity based on the differential combinator (and which often coincides with the notion of linearity from linear algebra). Thus, one needs the notion of internal *linear* homs.

In this paper, we introduce the notion of a **linearly closed Cartesian differential category**, which is a Cartesian differential category with internal linear hom $\mathscr{L}(A, B)$ (which can be interpreted as an object which represents the set of linear maps from A to B), a bilinear evaluation map $\mathscr{L}(A, B) \times A \xrightarrow{\varepsilon_\ell} B$, and the ability of currying maps which are linear in their second argument. Many analogues of the basic Cartesian

closed properties hold for the linearly closed setting. One of the axioms of the differential combinator states precisely that for any map $A \xrightarrow{f} B$, its derivative $A \times A \xrightarrow{\mathsf{D}[f]} B$ is linear in its second argument. Therefore, the Jacobian of f is defined as the curry of $\mathsf{D}[f]$, $A \xrightarrow{\mathbf{J}(f):=\lambda_\ell(\mathsf{D}[f])} \mathscr{L}(A,B)$, and numerous of the basic properties of the Jacobian from classical calculus hold. Many important examples of Cartesian differential categories are linearly closed, such as any differential Lawvere theory like SMOOTH or the Lawvere theory of polynomials over a commutative semiring, and also the coKleisli category of a differential category which is symmetric monoidal closed [4]. We also explain how a Cartesian closed differential category is linearly closed if and only if a certain linear idempotent on the internal hom splits. Therefore, a certain idempotent completion of a Cartesian closed differential category results in a linearly closed Cartesian differential category. Lastly, to define the gradient of a map, one requires the ability to take the transpose (or dagger) of linear maps. This can be achieved in a linearly closed Cartesian reverse differential category, which we explain is precisely a linear closed Cartesian differential category equipped with linear transpose maps $\mathscr{L}(A,B) \xrightarrow{\tau} \mathscr{L}(B,A)$. Therefore, the gradient of a map $A \xrightarrow{f} B$ is defined by taking the curry of its reverse derivative $A \times B \xrightarrow{\mathsf{R}[f]} A$, which is linear in its second argument, $A \xrightarrow{\nabla(f):=\lambda_\ell(\mathsf{R}[f])} \mathscr{L}(B,A)$, which we show is equal to the tranpose of its Jacobian, so $\nabla(f) = \tau \circ \mathbf{J}(f)$. In future works, the notions of Jacobians and gradient will be particularly useful when generalizing and applying automatic differentiation and machine learning algorithms, such as back-propagation or (reverse) gradient descent, in the setting of a Cartesian (reverse) differential category. In fact, internal linear homs are a key concept in Vákár's recent work on automatic differentiation [22].

2 Background: Cartesian Differential Categories and Linear Maps

In this background section, we review Cartesian (closed) differential categories and linear maps, as well as providing examples and a term logic notation that will help with intuition. For a more in-depth introduction to Cartesian differential categories, we refer the reader to [5, 11].

The underlying structure of a Cartesian differential category is that of a Cartesian left additive category, which in particular allows one to have zero maps and sums of maps, while also allowing for maps that do not preserve said sums or zeros. Maps that do preserve the additive structure are called *additive* maps. It is important to note that we do not assume that our Cartesian left additive categories necessarily have negatives, which allows for examples from computer science. For a category with (chosen) finite products we denote the (chosen) binary product as $\times$, with projection maps $A \times B \xrightarrow{\pi_0} A$ and $A \times B \xrightarrow{\pi_1} B$ and pairing object $\langle -, - \rangle$, and the (chosen) terminal object as $\top$.

Definition 2.1 *A **left additive category** [5, Definition 1.1.1] is a category $\mathbb{X}$ such that each hom-set $\mathbb{X}(A,B)$ is a commutative monoid with addition $\mathbb{X}(A,B) \times \mathbb{X}(A,B) \xrightarrow{+} \mathbb{X}(A,B)$, $(f,g) \mapsto f+g$, and zero $0 \in \mathbb{X}(A,B)$, and such that pre-composition preserves the additive structure: $(f+g) \circ a = f \circ a + g \circ a$ and $0 \circ a = 0$. A map $A \xrightarrow{f} B$ is said to be **additive** [5, Definition 1.1.1] if post-composition by f preserves the additive structure: $f \circ (a+b) = f \circ a + g \circ b$ and $f \circ 0 = 0$. A **Cartesian left additive category** [1] [5, Definition 1.2.1] is a left additive category $\mathbb{X}$ which has finite products and such that all the projection maps $A \times B \xrightarrow{\pi_0} A$ and $A \times B \xrightarrow{\pi_1} B$ are additive.*

Cartesian differential categories are Cartesian left additive categories that come equipped with a differential combinator, whose axioms capture the basic properties of the directional derivative from

---

[1]We note that the definition of a Cartesian left additive category presented here is not precisely that given in [5, Definition 1.2.1], but it is indeed equivalent, as explained in [11]

multivariable differential calculus. In the following definition, it is important to note that we follow the convention used in more recent work on Cartesian differential categories which flips the convention used in [5], so that the linear argument of the derivative is in the second argument rather than in the first.

Definition 2.2 *A **Cartesian differential category** [5, Definition 2.1.1] is a Cartesian left additive category* $\mathbb{X}$ *equipped with a **differential combinator** D, which is a family of operators* $\mathbb{X}(A,B) \xrightarrow{\text{D}} \mathbb{X}(A \times A, B)$, *where for a map* $A \xrightarrow{f} B$, *the resulting map* $A \times A \xrightarrow{\text{D}[f]} B$ *is called the derivative of f, and such that:*

[CD.1] $\text{D}[f+g] = \text{D}[f] + \text{D}[g]$ *and* $\text{D}[0] = 0$ **[CD.4]** $\text{D}[\langle f,g \rangle] = \langle \text{D}[f], \text{D}[g] \rangle$

[CD.2] $\text{D}[f] \circ \langle a, b+c \rangle = \text{D}[f] \circ \langle a,b \rangle + \text{D}[f] \circ \langle a,c \rangle$ **[CD.5]** $\text{D}[g \circ f] = \text{D}[g] \circ \langle f \circ \pi_0, \text{D}[f] \rangle$
       and $\text{D}[f] \circ \langle a,0 \rangle = 0$ **[CD.6]** $\text{D}[\text{D}[f]] \circ \langle \langle a,b \rangle, \langle 0,c \rangle \rangle = \text{D}[f] \circ \langle a,c \rangle$

[CD.3] $\text{D}[1_A] = \pi_1$ *and* $\text{D}[\pi_j] = \pi_j \circ \pi_1$ **[CD.7]** $\text{D}[\text{D}[f]] \circ \langle \langle a,b \rangle, \langle c,0 \rangle \rangle = \text{D}[\text{D}[f]] \circ \langle \langle a,c \rangle, \langle b,0 \rangle \rangle$

To help with the intuition, it is useful to use the term logic of Cartesian differential categories as introduced in [5, Section 4], which expresses the differential combinator as: $\text{D}[f](a,b) := \frac{\text{d}f(x)}{\text{d}x}(a) \cdot b$. **[CD.1]** says that the derivative of a sum is equal to the sum of the derivatives, $\frac{\text{d}f(x)+g(x)}{\text{d}x}(a) \cdot b = \frac{\text{d}f(x)}{\text{d}x}(a) \cdot b + \frac{\text{d}g(x)}{\text{d}x}(a) \cdot b$, and that the derivative of zero maps is zero, $\frac{\text{d}0}{\text{d}x}(a) \cdot b = 0$. **[CD.2]** says that derivatives are additive in their second argument, so that $\frac{\text{d}f(x)}{\text{d}x}(a) \cdot (b+c) = \frac{\text{d}f(x)}{\text{d}x}(a) \cdot b + \frac{\text{d}f(x)}{\text{d}x}(a) \cdot c$ and $\frac{\text{d}f(x)}{\text{d}x}(a) \cdot 0 = 0$. **[CD.3]** tells us what the derivatives of identity maps and projections maps are, so that $\frac{\text{d}x}{\text{d}x}(a) \cdot b = b$ and $\frac{\text{d}\pi_i(x_0,x_1)}{\text{d}(x_0,x_1)}(a_0,a_1) \cdot (b_0,b_1) = b_i$. **[CD.4]** says the derivative of a pairing of maps is equal to the pairing of the derivatives, $\frac{\text{d}\langle f(x),g(x) \rangle}{\text{d}x}(a) \cdot b = \left\langle \frac{\text{d}f(x)}{\text{d}x}(a) \cdot b, \frac{\text{d}g(x)}{\text{d}x}(a) \cdot b \right\rangle$. **[CD.5]** is the chain rule which tells us what the derivative of a composition of functions is, so $\frac{\text{d}g(f(x))}{\text{d}x}(a) \cdot b = \frac{\text{d}g(y)}{\text{d}y}(f(a)) \cdot \left(\frac{\text{d}f(x)}{\text{d}x}(a) \cdot b \right)$. The last two axioms **[CD.6]** and **[CD.7]** may look somewhat mysterious but essentially they capture properties of partial differentiation. Indeed, in any Cartesian differential category, one can define partial differentiation by inserting zeros in the total differential [5, Section 4.5], which in the term logic is written respectively as: $\frac{\text{d}f(x,b)}{\text{d}x}(a) \cdot c := \frac{\text{d}f(x,y)}{\text{d}\langle x,y \rangle}(a,b) \cdot (c,0)$ and $\frac{\text{d}f(a,y)}{\text{d}y}(b) \cdot c := \frac{\text{d}f(x,y)}{\text{d}\langle x,y \rangle}(a,b) \cdot (0,c)$. Thus, **[CD.6]** tells us what the partial derivative in the second argument of a derivative is, $\frac{\text{d}\frac{\text{d}f(x)}{\text{d}x}(a) \cdot z}{\text{d}z}(b) \cdot c = \frac{\text{d}f(x)}{\text{d}x}(a) \cdot c$, while **[CD.7]** captures the symmetry of the partial derivatives, $\frac{\text{d}\frac{\text{d}f(x)}{\text{d}x}(y) \cdot b}{\text{d}(y)}(a) \cdot c = \frac{\text{d}\frac{\text{d}f(y)}{\text{d}y}(x) \cdot c}{\text{d}(x)}(a) \cdot b$. We note that partial derivatives are also differential combinators for the simple slice categories [5, Corollary 4.5.2], and the total derivative is equal to the sum of the partial derivatives [5, Lemma 4.5.1], that is, $\frac{\text{d}f(x,y)}{\text{d}\langle x,y \rangle}(a,b) \cdot (c,d) = \frac{\text{d}f(x,b)}{\text{d}x}(a) \cdot c + \frac{\text{d}f(a,y)}{\text{d}y}(b) \cdot d$. More discussion on the differential combinator axioms can be found in [5, Remark 2.1.3].

Here are now our main examples of Cartesian differential categories (see [11] for a list of more examples of Cartesian differential categories):

Example 2.3 Let R be a commutative ring and let R-MOD be the category of R-modules and R-linear maps between them. R-MOD is a Cartesian differential category where for an R-linear map $M \xrightarrow{f} N$, its derivative $M \times M \xrightarrow{\text{D}[f]} N$ is defined as $\text{D}[f](x,y) = f(y)$. More generally, every category with finite biproducts is a where $\text{D}[f] = f \circ \pi_1$.

Example 2.4 Let $\mathbb{R}$ be the set of real numbers. Define SMOOTH as the category whose objects are the Euclidean real vector spaces $\mathbb{R}^n$ and whose maps are the real smooth functions $\mathbb{R}^n \xrightarrow{F} \mathbb{R}^m$ between

them. SMOOTH is a Cartesian differential category where the differential combinator is defined as the directional derivative of a smooth function. Recall that a smooth function $\mathbb{R}^n \xrightarrow{F} \mathbb{R}^m$ is in fact a tuple $F = \langle f_1, \ldots, f_m \rangle$ of smooth functions $\mathbb{R}^n \xrightarrow{f_i} \mathbb{R}$. Then using the convention that $\vec{x} \in \mathbb{R}^n$ are column vectors, the derivative $\mathbb{R}^n \times \mathbb{R}^n \xrightarrow{\mathsf{D}[F]} \mathbb{R}^m$ is defined as multiplying the Jacobian matrix of F at the first argument $\vec{x}$, which is an $m \times n$ matrix $\mathbf{J}(F)(\vec{x})$, with the second argument $\vec{y}$, seen as an $n \times 1$ matrix:

$$\mathsf{D}[F](\vec{x}, \vec{y}) := \mathbf{J}(F)(\vec{x})\vec{y} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(\vec{x}) & \frac{\partial f_1}{\partial x_2}(\vec{x}) & \cdots & \frac{\partial f_1}{\partial x_n}(\vec{x}) \\ \frac{\partial f_2}{\partial x_1}(\vec{x}) & \frac{\partial f_2}{\partial x_2}(\vec{x}) & \cdots & \frac{\partial f_2}{\partial x_n}(\vec{x}) \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_m}{\partial x_1}(\vec{x}) & \frac{\partial f_m}{\partial x_2}(\vec{x}) & \cdots & \frac{\partial f_m}{\partial x_n}(\vec{x}) \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} \sum\limits_{i=1}^{n} \frac{\partial f_1}{\partial x_i}(\vec{x})y_i \\ \vdots \\ \sum\limits_{i=1}^{n} \frac{\partial f_m}{\partial x_i}(\vec{x})y_i \end{bmatrix}$$

When $m = 1$, for a smooth function $\mathbb{R}^n \xrightarrow{f} \mathbb{R}$, $\mathsf{D}[f](\vec{x}, \vec{y})$ is precisely the directional derivative of f at point $\vec{x}$ and along the vector $\vec{y}$.

Example 2.5 An important source of examples of Cartesian differential categories are the coKleisli categories of differential categories [4]. Very briefly (and leaving out most of the details), a differential category [4, Definition 2.4] is a symmetric monoidal category (with tensor product $\otimes$ and unit k) with a comonad ! which comes equipped with a deriving transformation $!A \otimes A \xrightarrow{\mathsf{d}} A$ satisfying certain coherences which capture the basic algebraic properties of differentiation [3, Definition 7]. Examples of differential categories can be found in [3, Section 9]. By [5, Proposition 3.2.1], when a differential category has finite products, the coKleisli category of ! is a Cartesian differential category where the differential combinator is defined using the deriving transformation.

An important class of maps in a Cartesian differential category are the linear maps and maps which are linear in certain arguments. Essentially, a map is linear in an argument if when differentiating with respect to that argument (and keeping the other arguments constant), one gets back the starting map.

Definition 2.6 *In a Cartesian differential $\mathbb{X}$ with differential combinator D:*

(i) *A map $A \xrightarrow{f} B$ is **linear** [5, Definition 2.2.1] if $\mathsf{D}[f] \circ \langle a, b \rangle = f \circ b$ (i.e. $\frac{\mathsf{d}f(x)}{\mathsf{d}x}(a) \cdot b = f(b)$);*

(ii) *A map $A \times B \xrightarrow{f} C$ is **linear in its first argument** if f is linear with respect to the partial derivative in its first argument, that is, $\mathsf{D}[f] \circ \langle \langle a, b \rangle, \langle c, 0 \rangle \rangle = f \circ \langle c, b \rangle$ (i.e. $\frac{\mathsf{d}f(x,b)}{\mathsf{d}x}(a) \cdot c = f(c, b)$);*

(iii) *A map $A \times B \xrightarrow{f} C$ is **linear in its second argument** if f is linear with respect to the partial derivative in its second argument, that is, $\mathsf{D}[f] \circ \langle \langle a, b \rangle, \langle 0, d \rangle \rangle = f \circ \langle a, d \rangle$ (i.e. $\frac{\mathsf{d}f(a,y)}{\mathsf{d}y}(b) \cdot d = f(a, d)$);*

(iv) *A map $A \times B \xrightarrow{f} C$ is **bilinear** if it is linear in its first argument and linear in its second argument, or equivalently if $\mathsf{D}[f] \circ \langle \langle a, b \rangle, \langle c, d \rangle \rangle = f \circ \langle a, d \rangle + f \circ \langle c, b \rangle$ (i.e. $\frac{\mathsf{d}f(x,y)}{\mathsf{d}\langle x,y \rangle}(a, b) \cdot (c, d) = f(a, d) + f(c, b)$).*

Define the subcategory of linear maps $\mathsf{Lin}[\mathbb{X}]$ to be the category whose objects are the same as $\mathbb{X}$ and whose maps are D-linear in $\mathbb{X}$, and let $\mathsf{U} : \mathsf{Lin}[\mathbb{X}] \to \mathbb{X}$ be the obvious forgetful functor.

By **[CD.6]**, for any map f, its derivative $\mathsf{D}[f]$ is linear in its second argument, while by **[CD.3]**, identity maps and projections are linear. In fact, by [5, Lemma 2.2.2], linear maps are also closed under composition, addition, and pairings, which implies that $\mathsf{Lin}[\mathbb{X}]$ is a Cartesian left additive category with the same structure as $\mathbb{X}$. Furthermore, every linear map is also additive (though the converse is not always true), and as a result, the finite product structure of $\mathbb{X}$ becomes a finite biproduct structure in $\mathsf{Lin}[\mathbb{X}]$, where in particular, the injection maps are defined as $A \xrightarrow{\iota_0 := \langle 1_A, 0 \rangle} A \times B$ and $B \xrightarrow{\iota_1 := \langle 0, 1_B \rangle} A \times B$.

Example 2.7 In R-MOD, every R-linear map is linear in the Cartesian differential sense, so we have that $\mathsf{Lin}[R\text{-MOD}] = R\text{-MOD}$. On the other hand, $M \times M' \xrightarrow{f} N$ is linear in its first argument (resp. second argument) if and only if $f(x,y) = f(x,0)$ (resp. $f(x,y) = f(0,y)$). Therefore, the only maps which are bilinear in the Cartesian differential sense are the zero maps. The same story holds true for any category with finite biproducts.

Example 2.8 In SMOOTH, a smooth function $\mathbb{R}^n \xrightarrow{F} \mathbb{R}^m$ is linear in the Cartesian differential sense if and only if it is $\mathbb{R}$-linear in the classical sense, that is, $F(s\vec{x} + t\vec{y}) = sF(\vec{x}) + tF(\vec{y})$ for all $s,t \in \mathbb{R}$ and $\vec{x}, \vec{y} \in \mathbb{R}^n$. So $\mathsf{Lin}[\mathsf{SMOOTH}]$ is the category of $\mathbb{R}$-linear maps between the $\mathbb{R}$-vector spaces $\mathbb{R}^n$. Similarly, a smooth function $\mathbb{R}^n \times \mathbb{R}^k \xrightarrow{G} \mathbb{R}^m$ is linear in its first argument (resp. second argument) if and only if it is $\mathbb{R}$-linear in its first argument $\mathbb{R}^n$ (resp. second argument $\mathbb{R}^k$), that is, $G(s\vec{x} + t\vec{y}, \vec{z}) = sG(\vec{x}, \vec{z}) + tG(\vec{y}, \vec{z})$ (resp. $G(\vec{z}, s\vec{x} + t\vec{y}) = sG(\vec{z}, \vec{x}) + tG(\vec{z}, \vec{x})$). Thus, $\mathbb{R}^n \times \mathbb{R}^k \xrightarrow{G} \mathbb{R}^m$ is bilinear if and only if it is $\mathbb{R}$-bilinear.

We conclude this section by reviewing Cartesian closed differential categories (also sometimes called differential λ-categories [8, 19]) which, as the name suggests, are Cartesian differential categories whose underlying category is also Cartesian closed and such that the differential combinator is compatible with the curry operator. For a Cartesian closed category, we denote the internal hom by $[A,B]$, the evaluation map by $[A,B] \times A \xrightarrow{\varepsilon} B$, and for a map $A \times B \xrightarrow{f} C$, its curry is denoted $A \xrightarrow{\lambda(f)} [B,C]$.

Definition 2.9 *A **Cartesian closed differential category** [9, Section 4.6] is a Cartesian differential category which is also a Cartesian closed category such that all evaluation maps $[A,B] \times A \xrightarrow{\varepsilon} B$ are linear in their first argument (note that this implies that $\lambda(f + g) = \lambda(f) + \lambda(g)$ and $\lambda(0) = 0$).*

Here are now some examples of Cartesian closed differential categories.

Example 2.10 Every model of the differential λ-calculus [14] induces a Cartesian closed differential category [9, Theorem 4.3], and conversely, every Cartesian closed differential category induces a model of the differential λ-calculus [8, Theorem 4.12].

Example 2.11 R-MOD and SMOOTH are not Cartesian closed differential categories since neither are Cartesian closed.

Example 2.12 A differential storage category [3, Definition 10] is a differential category with finite products and Seely isomorphisms, that is, $!(A \times B) \cong !A \otimes !B$ and $!\top \cong k$ are isomorphisms. By [6, Theorem 4.4.2], for a differential storage category whose base category is also symmetric monoidal closed (with internal homs denoted by $A \multimap B$), the coKleisli category is a Cartesian closed differential category where $[A,B] = !A \multimap B$. Examples can be found in [8, Section 5].

Example 2.13 The category of convenient vector spaces and smooth functions between them is a Cartesian closed differential category [20, Example 2.4.2]. For a detailed introduction to convenient vector spaces, see [16]. Briefly, a convenient vector space is a locally convex vector space E such that for every smooth function $\mathbb{R} \xrightarrow{c} E$, called a smooth curve, there exists another smooth curve $\mathbb{R} \xrightarrow{\bar{c}} E$ such that $\bar{c}' = c$. If E and F are convenient vector spaces, then a smooth function is a function $E \xrightarrow{f} F$ which preserves smooth curves, that is, if c is a smooth curve of E, then $f \circ c$ is a smooth curve of F. Let CON be the category of convenient vector spaces and smooth functions between them. CON is a Cartesian closed differential category where the internal hom is given by $[E,F] = \{E \xrightarrow{f} F \mid f \text{ is smooth}\}$, which is indeed a convenient vector space, and for a smooth function $E \xrightarrow{f} F$, its derivative $E \times E \xrightarrow{\mathsf{D}[f]} F$ is defined as $\mathsf{D}[f](x,y) := \lim_{t \to 0} \frac{f(x+ty)-f(x)}{t}$, where $t \in \mathbb{R}$. Note that while SMOOTH is not Cartesian closed, SMOOTH is a sub-Cartesian differential category of CON. Furthermore, CON is also the coKleisli category of a differential storage category [7].

3 Linearly Closed Cartesian Differential Categories and Jacobians

In this section, we introduce the main contribution of this paper: linearly closed Cartesian differential categories, which are the appropriate setting for defining Jacobians (in the Fréchet/Gateaux derivative sense) in the context of Cartesian differential categories.

Definition 3.1 *A **linearly closed Cartesian differential category** is a Cartesian differential category such that for each pair of objects A and B, there is an object $\mathscr{L}(A,B)$, called the **internal linear hom**, and a bilinear map $\mathscr{L}(A,B) \times A \xrightarrow{\varepsilon_\ell} B$ called the **evaluation map**, such that for every map $A \times B \xrightarrow{f} C$ which is linear in its second argument, there exists a unique map $A \xrightarrow{\lambda_\ell(f)} \mathscr{L}(B,C)$, called the **linear curry** of f, such that $f = A \times B \xrightarrow{\lambda_\ell(f) \times 1_B} \mathscr{L}(B,C) \times B \xrightarrow{\varepsilon_\ell} C$.*

It is worth highlighting the major differences between being linearly closed and Cartesian closed. (1) $\mathscr{L}(A,B)$ should be interpreted as the internal version of $\mathsf{Lin}[\mathbb{X}](A,B)$, so the linear maps from A to B, while $[A,B]$ is the internal version of $\mathbb{X}(A,B)$. (2) The evaluation map $\mathscr{L}(A,B) \times A \xrightarrow{\varepsilon_\ell} B$ is bilinear, while $[A,B] \times A \xrightarrow{\varepsilon} B$ is only linear in its first argument. The difference here is due to the fact that linear maps are additive, while an arbitrary map may not be. (3) In a linearly closed setting, we are only able to curry linear arguments. (4) It is possible to be linearly closed and not Cartesian closed, and vice-versa. We stress that being linearly closed does not imply that $\mathsf{Lin}[\mathbb{X}]$ is Cartesian closed. However, if a Cartesian differential category $\mathbb{X}$ has a monoidal representation in the sense of [6, Section 3.2], then we conjecture that being linearly closed is equivalent to $\mathsf{Lin}[\mathbb{X}]$ being monoidal closed (this approach has been studied by Gallagher and MacAdam [15], and we note that both SMOOTH and CON have monoidal representation).

Now recall that **[CD.6]** says that for any map f, $\mathsf{D}[f]$ is linear in its second argument. Therefore, the Jacobian of f is defined as the linear curry of its derivative:

Definition 3.2 *In a linearly closed Cartesian differential category, the **Jacobian** of a map $A \xrightarrow{f} B$ is the map $A \xrightarrow{\mathbf{J}(f)} \mathscr{L}(A,B)$ defined as the linear curry of $A \times A \xrightarrow{\mathsf{D}[f]} B$, that is, $\mathbf{J}(f) := \lambda_\ell(\mathsf{D}[f])$.*

Here are now some example of linearly closed Cartesian differential categories and Jacobians:

Example 3.3 *R*-MOD is not a linearly closed Cartesian differential category since the only bilinear maps are zero maps. The same is true for any category with finite biproducts.

Example 3.4 SMOOTH is a linearly closed Cartesian differential category where the internal linear hom is $\mathscr{L}(\mathbb{R}^n, \mathbb{R}^m) = \mathbb{R}^{nm}$ and the evaluation map $\mathbb{R}^{nm} \times \mathbb{R}^n \xrightarrow{\varepsilon_\ell} \mathbb{R}^m$ is defined as laying out the first argument, which is a column vector of size nm, into a matrix of size $m \times n$ and then multiplying by the second argument, seen as a matrix of size $n \times 1$:

$$\varepsilon_\ell(\vec{x}, \vec{y}) := \begin{bmatrix} x_1 & x_2 & \dots & x_n \\ x_{n+1} & x_{n+2} & \dots & x_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ x_{(m-1)n+1} & x_{(m-1)n+2} & \dots & x_{mn} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} \sum\limits_{i=1}^{n} x_i y_i \\ \vdots \\ \sum\limits_{i=1}^{n} x_{(m-1)n+i} y_i \end{bmatrix}$$

When $m = 1$, the evaluation map is given by the dot product of vectors $\varepsilon_\ell(\vec{x}, \vec{y}) = \vec{x} \cdot \vec{y} = \sum\limits_{i=1}^{n} x_i y_i$. To define the linear curry, let $e_i \in \mathbb{R}^n$ be the canonical basis vectors, $e_i = [0, \dots, 0, 1, 0, \dots, 0]^\mathsf{T}$. Then for a

smooth function $\mathbb{R}^n \times \mathbb{R}^k \xrightarrow{G=\langle g_1,...,g_m\rangle} \mathbb{R}^m$, which is linear in its second argument, define $\mathbb{R}^n \xrightarrow{\lambda(G)} \mathbb{R}^{km}$ as $\lambda(G)(\vec{x}) = [g_1(\vec{x}, e_1), \ldots, g_1(\vec{x}, e_k), g_2(\vec{x}, e_1), \ldots, g_m(\vec{x}, e_k)]^\top$. For a smooth function $\mathbb{R}^n \xrightarrow{F=\langle f_1,...,f_m\rangle} \mathbb{R}^m$, taking $\mathbf{J}(F)(\vec{x}) = \lambda_\ell(\mathsf{D}[F])(\vec{x})$ results precisely in the Jacobain matrix of F at $\vec{x}$ interpreted as column vector of size nm, $\mathbf{J}(F) = [\frac{\partial f_1}{\partial x_1}(\vec{x}), \ldots, \frac{\partial f_1}{\partial x_n}(\vec{x}), \frac{\partial f_2}{\partial x_1}(\vec{x}), \ldots, \frac{\partial f_m}{\partial x_n}(\vec{x})]^\top$, which when post-composed in the evaluation map results in laying it out back into a $m \times n$ matrix.

Example 3.5 For a differential category with finite products, if the base the category is symmetric monoidal closed, then the coKleisli category will be a linearly closed differential category where the internal linear hom is given by $\mathscr{L}(A,B) := A \multimap B$. For a coKleisli map $!A \xrightarrow{f} B$, its Jacobian $!A \xrightarrow{\mathbf{J}(f)} A \multimap B$ is define as curry in the base symmetric monoidal category of the composite $!A \otimes A \xrightarrow{d} !A \xrightarrow{f} B$. Note that unlike in the Cartesian closed case, we do note need to assume the presence of Seely isomorphisms for the linearly closed case. That said, it can be shown that the coKleisli category of a differential storage category is linearly closed if and only if the base category is symmetric monoidal closed. Furthermore, this internal linear hom captures the notion fo linear types in Vákár's work on automatic differentiation [22].

Example 3.6 CON is a linearly closed Cartesian differential category where the internal linear hom is $\mathscr{L}(E,F) = \{E \xrightarrow{f} F \mid f \text{ is smooth and } \mathbb{R}\text{-linear}\} \subset [E,F]$, and the evaluation map $\mathscr{L}(E,F) \times E \xrightarrow{\varepsilon_\ell} F$ is defined in the obvious way, $\varepsilon_\ell(f,x) = f(x)$. A smooth function $E \times E' \xrightarrow{g} F$ is linear in its second argument if and only if g is $\mathbb{R}$-linear in its second argument. So if g is linear in its second argument, then $E \xrightarrow{\lambda_\ell(g)} \mathscr{L}(E',F)$ is defined as $\lambda_\ell(g)(x)(y) = g(x,y)$. For a smooth functor $E \xrightarrow{f} F$, its Jacobian $E \xrightarrow{\mathbf{J}(f)} \mathscr{L}(E,F)$ is defined as $\mathbf{J}(f)(x)(y) = \mathsf{D}[f](x,y)$.

Being linearly closed shares many analogous properties to being Cartesian closed. In term logic notation, we write $\lambda_\ell(f)(a) = \lambda_\ell y.f(a,y)$ and $\varepsilon_\ell(g,a) = g(a)$, and $(\lambda_\ell y.f(a,y))(b) = f(a,b)$, and so $\mathbf{J}(f)(a) = \lambda_\ell y.\frac{\mathrm{d}f(x)}{\mathrm{d}x}(a) \cdot y$.

Proposition 3.7 *Let* $\mathbb{X}$ *be a linearly closed Cartesian differential category.*

1. $\lambda_\ell(f+g) = \lambda_\ell(f) + \lambda_\ell(g)$ *and* $\lambda_\ell(0) = 0$

2. $\mathsf{D}[\lambda_\ell(f)] = \lambda_\ell(\mathsf{D}[f] \circ \langle\langle \pi_0 \circ \pi_0, 0\rangle, \langle \pi_1 \circ \pi_0, \pi_1\rangle\rangle)$ *(i.e.* $\frac{\mathrm{d}\lambda_\ell y.f(x,y)}{\mathrm{d}x}(a) \cdot b = \lambda_\ell y.\frac{\mathrm{d}f(x,y)}{\mathrm{d}x}(a) \cdot c$*).*

3. *There is a functor* $\mathsf{LIN}[\mathbb{X}]^{op} \times \mathsf{LIN}[\mathbb{X}] \xrightarrow{\mathscr{L}} \mathsf{LIN}[\mathbb{X}]$ *which maps a pair of object to the internal linear hom* $\mathscr{L}(A,B)$*, and sends a pair of linear maps* $A \xrightarrow{f} B$ *and* $X \xrightarrow{g} Y$ *to the linear map* $\mathscr{L}(B,X) \xrightarrow{\mathscr{L}(f,g)} \mathscr{L}(A,Y)$ *defined as the linear curry of* $\mathscr{L}(B,X) \times A \xrightarrow{1 \times f} \mathscr{L}(B,X) \times B \xrightarrow{\varepsilon_\ell} X \xrightarrow{g} Y$

4. *There are natural isomorphisms: (i)* $\mathscr{L}(A, B \times C) \cong \mathscr{L}(A,B) \times \mathscr{L}(A,C)$*, (ii)* $\mathscr{L}(A,C) \times \mathscr{L}(B,C) \cong \mathscr{L}(A \times B, C)$*, (iii)* $\mathscr{L}(A, \top) \cong \top \cong \mathscr{L}(\top, A)$*, and (iv)* $\mathscr{L}(A, \mathscr{L}(B,C)) \cong \mathscr{L}(B, \mathscr{L}(A,C))$*.*

PROOF: (Sketch) The computations for (1) and (2) are essentially the same as those in [9, Lemma 4.10]. The proof that $\mathscr{L}$ is a functor, $\mathscr{L}(A, \top) \cong \top$, $\mathscr{L}(A, B \times C) \cong \mathscr{L}(A,B) \times \mathscr{L}(A,C)$, and $\mathscr{L}(A, \mathscr{L}(B,C)) \cong \mathscr{L}(B, \mathscr{L}(A,C))$, are essentially the same as in the Cartesian closed case. On the other hand, $\top \cong \mathscr{L}(\top, A)$ and $\mathscr{L}(A,C) \times \mathscr{L}(B,C) \cong \mathscr{L}(A \times B, C)$ intuitively follow from the fact that $\top$ is an initial object and $\times$ is a coproduct in $\mathsf{LIN}[\mathbb{X}]$. In other words, the only linear map of type $\top \to A$ is 0, while a linear map of type $A \times B \to C$ is actually a pair of linear maps $A \to B$ and $B \to C$. $\square$

We conjecture that being linearly closed induces a non-unital closed category structure on $\mathsf{LIN}[\mathbb{X}]$. The non-unital part comes from the fact that there may not be an object R such that $A \cong \mathscr{L}(R,A)$ for all

objects A. While both SMOOTH and CON have such an object, namely $R = \mathbb{R}$, if we take SMOOTH$_{even}$ the subcategory of Euclidean spaces of even dimension, then SMOOTH$_{even}$ is still a linearly closed Cartesian differential category, but does not have a unit. In future work, it will definitively be of interest of studying linearly closed Cartesian differential closed with both a unit and monoidal representation. Furthermore, note that in the Cartesian closed setting we have that $[A, [B, C]] \cong [A \times B, C]$, however in the linearly closed setting, $\mathscr{L}(A, \mathscr{L}(B, C))$ represents the bilinear maps $A \times B \to C$, which is different from $\mathscr{L}(A \times B, C)$, which represents the linear maps $A \times B \to C$, and so $\mathscr{L}(A, \mathscr{L}(B, C)) \ncong \mathscr{L}(A \times B, C)$.

Next we turn our attention to properties of the Jacobian. In particular, we provide analogues of three basic classical identities: (1) $\mathbf{J}(F + G)(\vec{x}) = \mathbf{J}(F)(\vec{x}) + \mathbf{J}(G)(\vec{x})$, (2) if F is $\mathbb{R}$-linear with associated matrix A (i.e. $F(\vec{x}) = A\vec{x}$) then $\mathbf{J}(F)(\vec{x}) = A$, and (3) $\mathbf{J}(G \circ F)(\vec{x}) = \mathbf{J}(G)(F(\vec{x}))\mathbf{J}(F)(\vec{x})$. The first identity is easy to generalize, while the latter two requires defining some extra maps. For a linear map $A \xrightarrow{f} B$, let $\top \xrightarrow{p_f} \mathscr{L}(A, B)$ be the linear curry of the composite $\top \times A \xrightarrow{\pi_1} A \xrightarrow{f} B$, so in term logic notation $p_f = \lambda_\ell x.f(x)$. Next, define the map $\mathscr{L}(B, C) \times \mathscr{L}(A, B) \xrightarrow{\odot} \mathscr{L}(A, C)$ as the linear curry of the composite $\mathscr{L}(B, C) \times \mathscr{L}(A, B) \times A \xrightarrow{1_{\mathscr{L}(B,C)} \times \varepsilon_\ell} \mathscr{L}(B, C) \times B \xrightarrow{\varepsilon_\ell} C$, and note that $\odot$ captures composition of linear maps, which we write in term logic notation as $g \odot f = \lambda_\ell x.g(f(x))$. We leave it to the reader to check for themselves that in SMOOTH, if $\mathbb{R}^n \xrightarrow{F} \mathbb{R}^m$ is an $\mathbb{R}$-linear map with associated $n \times m$ matrix A, then p_F is precisely A is column vector form, while recall that composition of $\mathbb{R}$-linear maps corresponds to matrix multiplication, and so $\odot$ plays the role of matrix multiplication.

Proposition 3.8 *In a linearly closed Cartesian differential category:*

(i) $\mathbf{J}(f + g) = \mathbf{J}(f) + \mathbf{J}(g)$ *and* $\mathbf{J}(0) = 0$ *(i.e* $\mathbf{J}(f + g)(a) = \mathbf{J}(f)(a) + \mathbf{J}(g)(a)$ *and* $\mathbf{J}(0)(a) = 0$*).*

(ii) *If* $A \xrightarrow{f} B$ *is linear, then* $\mathbf{J}(f) = p_f \circ 0$ *(i.e.* $\mathbf{J}(f)(a) = \lambda_\ell x.f(x)$*).*

(iii) $\mathbf{J}(g \circ f) = \odot \circ \langle \mathbf{J}(g) \circ f, \mathbf{J}(f) \rangle$ *(i.e.* $\mathbf{J}(g \circ f)(a) = \mathbf{J}(g)(f(a)) \odot \mathbf{J}(f)(a)$*)*

PROOF: (Sketch). The first identity follows immediately from **[CD.1]** and Prop.3.7.(1). For the second identity, if f is linear then we using the term logic we compute: $\mathbf{J}(f)(a) = \lambda_\ell y.\frac{\mathrm{d}f(x)}{\mathrm{d}x}(a) \cdot y = \lambda_\ell y.f(y)$. Lastly for the third identity, we use the chain rule **[CD.5]** to compute: $\mathbf{J}(g \circ f)(a) = \lambda_\ell y.\frac{\mathrm{d}g(f(x))}{\mathrm{d}x}(a) \cdot y = \lambda_\ell y.\frac{\mathrm{d}g(z)}{\mathrm{d}z}(f(a)) \cdot \left(\frac{\mathrm{d}f(x)}{\mathrm{d}x}(a) \cdot y \right) = \lambda_\ell y.\mathbf{J}(g)(f(a))(\mathbf{J}(f)(a)(y)) = \mathbf{J}(g)(f(a)) \odot \mathbf{J}(f)(a)$. $\square$

We conjecture that it is most likely possible to provide an equivalent alternative axiomatization of a linearly closed Cartesian differential category instead as a Cartesian left additive category equipped with $\mathscr{L}(-, -)$, ε_ℓ, and $\mathbf{J}$, where one would define the differential combinator as $\mathsf{D}[-] = \varepsilon_\ell \circ (\mathbf{J}(-) \times 1)$. While this is definitely of interest, it does require a bit of work to properly setup everything. As such, this will be a story for another time.

We finish this section by providing necessary and sufficient conditions for when a Cartesian closed differential category is also linearly closed, and then explain how every Cartesian closed differential category embeds into a linearly closed Cartesian differential via splitting linear idempotents. We start with the definition of linear idempotents splitting linearly (which is a special case of [9, Section 4.5]):

Definition 3.9 *In a Cartesian differential category, a **linear split idempotent** is an idempotent $A \xrightarrow{e} A$ which is linear and such that there are linear maps $A \xrightarrow{r} B$ and $B \xrightarrow{s} A$ such that $s \circ r = e$ and $r \circ s = 1_B$. A **linear idempotent complete** Cartesian differential category is a Cartesian differential category such that all linear idempotents are linear split idempotents.*

The key idea is that we will define a linear idempotent on the internal hom $[A,B]$ which linearly splits via the internal linear hom $\mathscr{L}(A,B)$. To define this idempotent, we will require the notion of partial linearization [11, Proposition 5.5]. So for a map $A \times B \xrightarrow{f} C$, define its linearization in context A, as the map $A \times B \xrightarrow{\mathsf{L}^A[f]} C$ which is defined as $\mathsf{D}[f] \circ \langle \langle \pi_0, 0 \rangle, \langle 0, \pi_1 \rangle \rangle$, i.e. $\mathsf{L}^A[f](a,b) = \frac{\mathrm{d}f(a,y)}{\mathrm{d}y}(0) \cdot b$. The map $\mathsf{L}^A[f]$ is now linear in its second argument. So in a Cartesian closed differential category, define $[A,B] \xrightarrow{\ell} [A,B]$ as the linear curry of the partial linearization of the evaluation map $[A,B] \times A \xrightarrow{\mathsf{L}^{[A,B]}[\varepsilon]} B$, that is, $\ell = \lambda_\ell \left(\mathsf{L}^{[A,B]}[\varepsilon] \right)$. In term logic notation, we have that $\ell(f) = \lambda_\ell y. \frac{\mathrm{d}f(x)}{\mathrm{d}x}(0) \cdot y$, and so ℓ is understood as mapping an arbitrary map f to its linearization [11, Proposition 3.6]. As such, it follows that ℓ is an linear idempotent.

Proposition 3.10 *A Cartesian closed differential category is linearly closed if and only if $[A,B] \xrightarrow{\ell} [A,B]$ is a linear split idempotent.*

PROOF: (Sketch) Suppose that our Cartesian closed differential category is also linearly closed. As explained above, ℓ is already idempotent and linear. So it remains to construct a linear splitting of ℓ. First consider the curry of the evaluation map for the linear closed structure, $\mathscr{L}(A,B) \xrightarrow{\lambda(\varepsilon_\ell)} [A,B]$ which is linear since ε_ℓ was linear in its first argument. Next, consider the linear curry of the partial linearization of the evaluation map for the Cartesian closed structure, $[A,B] \xrightarrow{\lambda_\ell\left(\mathsf{L}^{[A,B]}[\varepsilon]\right)} \mathscr{L}(A,B)$ which is also linear since $\mathsf{L}^{[A,B]}[\varepsilon]$ is linear in its first argument. Then it follows that $\lambda_\ell \left(\mathsf{L}^{[A,B]}[\varepsilon] \right) \circ \lambda(\varepsilon_\ell) = 1_{\mathscr{L}(A,B)}$ and $\lambda(\varepsilon_\ell) \circ \lambda_\ell \left(\mathsf{L}^{[A,B]}[\varepsilon] \right) = \ell$. So ℓ is a linear split idempotent. Conversely, suppose that $[A,B] \xrightarrow{\ell} [A,B]$ is a linear split idempotent via $[A,B] \xrightarrow{r} \mathscr{L}(A,B)$ and $\mathscr{L}(A,B) \xrightarrow{s} [A,B]$. Then the internal linear hom is given by the object of the splitting $\mathscr{L}(A,B)$, while the evaluation map is defined as the composite $\varepsilon_\ell := \mathscr{L}(A,B) \times A \xrightarrow{s \times 1_A} [A,B] \times A \xrightarrow{\mathsf{L}^{[A,B]}[\varepsilon]} B$, which is bilinear since $\mathsf{L}^{[A,B]}[\varepsilon]$ is bilinear and s is linear. The linear curry of a map $A \times B \xrightarrow{f} C$ which is linear in its second argument is defined as the composite $\lambda_\ell(f) := A \xrightarrow{\lambda(f)} [A,B] \xrightarrow{r} \mathscr{L}(A,B)$. So our Cartesian closed differential category is linearly closed. □

In a Cartesian closed differential category which is also linearly closed, for a map $A \times B \xrightarrow{f} C$ which is linear in its second argument, its curry $A \xrightarrow{\lambda(f)} [B,C]$ factors through its linear curry $A \xrightarrow{\lambda_\ell(f)} \mathscr{L}(B,C)$ in the sense that $\lambda(f) = A \xrightarrow{\lambda_\ell(f)} \mathscr{L}(B,C) \xrightarrow{s} [A,B]$. It is also worth pointing out that while it is true that $[A,B] \times A \xrightarrow{\mathsf{L}^{[A,B]}[\varepsilon]} B$ is bilinear and if $A \times B \xrightarrow{f} C$ is linear in its second argument, then we also have that $f = \mathsf{L}^{[B,C]}[\varepsilon] \circ (\lambda(f) \times 1_B)$, this does not give a linearly closed structure since $\lambda(f)$ may fail the uniqueness requirement (i.e. $\mathsf{L}^{[A,B]}[\varepsilon]$ is not monic in its first argument). For example, we have that $\mathsf{L}^{[A,B]}[\varepsilon] = \mathsf{L}^{[A,B]}[\varepsilon] \circ (\ell \times 1_A)$ but $\ell \neq 1_{[A,B]}$.

Given any Cartesian differential category, we may build its linear idempotent completion in the obvious way. For a Cartesian differential category $\mathbb{X}$, let $\mathsf{LS}[\mathbb{X}]$ be the category whose objects are pairs $(A, A \xrightarrow{e} A)$ consisting of an object A and a linear idempotent e, and whose maps $(A,e) \xrightarrow{f} (B,e')$ are maps $A \xrightarrow{f} B$ such that $f \circ e = e' \circ f$. By [11, Proposition 4.7, Corollary 4.1], $\mathsf{LS}[\mathbb{X}]$ is a Cartesian differential category where the differential combinator is defined as in $\mathbb{X}$, and $\mathsf{LS}[\mathbb{X}]$ is also linear idempotent complete. Furthermore, if $\mathbb{X}$ was a Cartesian closed differential category, then $\mathsf{LS}[\mathbb{X}]$ is also a Cartesian closed differential category. Therefore, putting all of this together, we obtain the following:

Proposition 3.11 *If $\mathbb{X}$ is a Cartesian closed differential category, then $\mathsf{LS}[\mathbb{X}]$ is a linearly closed Cartesian closed differential category.*

4 Cartesian Reverse Differential Categories and Gradients

In most of the standard conventions of calculus, the gradient of a scalar-valued smooth function is the transpose of its Jacobian. In this section, we explain how Cartesian reverse differential categories are the appropriate setting in which to define gradients and transposes. We briefly review Cartesian reverse differential categories and invite readers to see the full story in the original paper [10].

Definition 4.1 *A **Cartesian reverse differential category** is a Cartesian left additive category $\mathbb{X}$ equipped with a **reverse differential combinator** R, which is a family of operators $\mathbb{X}(A,B) \xrightarrow{\mathsf{R}} \mathbb{X}(A \times B, A)$, where for a map $A \xrightarrow{f} B$, the resulting map $A \times B \xrightarrow{\mathsf{R}[f]} A$ is called the reverse derivative of f, and such that the seven axioms found in [10, Definition 13] hold.*

Example 4.2 SMOOTH is a Cartesian reverse differential category where for a smooth function $\mathbb{R}^n \xrightarrow{F} \mathbb{R}^m$, $F = \langle f_1, \ldots, f_m \rangle$, its reverse derivative $\mathbb{R}^n \times \mathbb{R}^m \xrightarrow{\mathsf{R}[F]} \mathbb{R}^n$ is defined by multiplying the transpose of the Jacobian matrix of F at the first argument $\vec{x}$, which is an $n \times m$ matrix $\mathbf{J}(F)(\vec{x})^\mathsf{T}$, with the second argument $\vec{y}$, viewed as an $m \times 1$ matrix:

$$\mathsf{R}[F](\vec{x},\vec{y}) := \mathbf{J}(F)(\vec{x})^\mathsf{T}\vec{y} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(\vec{x}) & \frac{\partial f_2}{\partial x_1}(\vec{x}) & \cdots & \frac{\partial f_m}{\partial x_1}(\vec{x}) \\ \frac{\partial f_1}{\partial x_2}(\vec{x}) & \frac{\partial f_2}{\partial x_2}(\vec{x}) & \cdots & \frac{\partial f_m}{\partial x_2}(\vec{x}) \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_1}{\partial x_n}(\vec{x}) & \frac{\partial f_2}{\partial x_n}(\vec{x}) & \cdots & \frac{\partial f_m}{\partial x_n}(\vec{x}) \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} = \begin{bmatrix} \sum\limits_{j=1}^{m} \frac{\partial f_j}{\partial x_1}(\vec{x})y_j \\ \vdots \\ \sum\limits_{j=1}^{m} \frac{\partial f_j}{\partial x_n}(\vec{x})y_j \end{bmatrix}$$

When $m = 1$, for a smooth functoion $\mathbb{R}^n \xrightarrow{f} \mathbb{R}$, recall that its gradient at $\vec{x} \in \mathbb{R}^n$ is defined as $\nabla(f)(\vec{x}) = \mathbf{J}(F)(\vec{x})^\mathsf{T} = [\frac{\partial f}{\partial x_1}(\vec{x}), \ldots, \frac{\partial f}{\partial x_n}(\vec{x})]^\mathsf{T}$. So its reverse derivative $\mathbb{R}^n \times \mathbb{R} \xrightarrow{\mathsf{R}[f]} \mathbb{R}^n$ is given by multiplying its gradient at $\vec{x}$ with the scalar in the second argument, $\mathsf{R}[f](\vec{x},y) = \nabla(f)(\vec{x})y$.

Every Cartesian reverse differential category is also a Cartesian differential category [10, Theorem 16], where the differential combinator D is induced by the reverse differential combinator R. Explicitly, for a map $A \xrightarrow{f} B$, note that its second order reverse derivative is of type $(A \times B) \times A \xrightarrow{\mathsf{R}[\mathsf{R}[f]]} A \times B$. Then the derivative of f is defined as the composite $\mathsf{D}[f] := A \times A \xrightarrow{\langle 1_A, 0 \rangle \times 1_A} (A \times B) \times A \xrightarrow{\mathsf{R}[\mathsf{R}[f]]} A \times B \xrightarrow{\pi_1} B$. Conversely, every Cartesian differential category with a **contextual linear dagger** [10, Definition 39] is a Cartesian reverse differential category [10, Theorem 41]. Very briefly, a contextual linear dagger amounts to an operator $(-)^{\dagger[-]}$ which sends every map $A \times B \xrightarrow{f} C$ which is linear in its second argument to a map $A \times C \xrightarrow{f^{\dagger[A]}} B$ which is again linear in its second argument, and such that $(-)^{\dagger[A]}$ makes the category of maps which are linear in context A into a †-category with finite †-biproducts. Then the reverse derivative of a map $A \xrightarrow{f} B$ is defined the taking the dagger in context of its derivative, that is, $\mathsf{R}[f] := A \times B \xrightarrow{\mathsf{D}[f]^{\dagger[A]}} B$. Furthermore, these constructions are inverses of each other [10, Theorem 42].

In a Cartesian reverse differential category, for any map f, its reverse derivative $\mathsf{R}[f]$ is linear in its second argument. Therefore we define the gradient of f as the linear curry of its reverse derivative:

Definition 4.3 *A **linearly closed Cartesian reverse differential category** is a Cartesian reverse differential category whose induced Cartesian differential structure is linearly closed. In a linearly closed Cartesian reverse differential category, the **gradient** of a map $A \xrightarrow{f} B$ is the map $A \xrightarrow{\nabla(f)} \mathscr{L}(B,A)$ defined as the linear curry of $A \times B \xrightarrow{\mathsf{R}[f]} A$, that is, $\nabla(f) := \lambda_\ell(\mathsf{R}[f])$.*

Let us explain why this is the correct definition of the gradient by proving that it is the transpose of the Jacobian. To do so, we first need the notion of a transpose operator:

Definition 4.4 *A linearly closed Cartesian differential category has a **linear transpose** if for every pair of objects A and B, there is a linear map $\mathscr{L}(A,B) \xrightarrow{\tau} \mathscr{L}(B,A)$ such that (i) $\tau \circ \tau = 1$, (ii) $\tau \circ p_{1_A} = p_{1_A}$ and $\tau \circ p_{\pi_j} = \tau \circ p_{\iota_j}$, and (iii) $\tau \circ \odot = \odot \circ \langle \tau \circ \pi_1, \tau \circ \pi_0 \rangle$.*

As we will see below, it turns out that in the linearly closed setting, having a linear transpose is equivalent to having a contextual linear dagger (and so also having a reverse differential combinator). Therefore, the linear transpose axioms are analogues of the fact that in a Cartesian reverse differential category, $\mathsf{Lin}[\mathbb{X}]$ is a †-category with †-biproducts [10, Proposition 24], and so $\tau(f) = f^\dagger$. Explicitly, (i) captures the fact † is an involution $f^{\dagger\dagger} = f$, (ii) captures the fact that † maps identities to identities, $1^\dagger = 1$, and projections to injections, $\pi_j^\dagger = \iota_j$, while (iii) tells us that † is contravariant, $(g \circ f)^\dagger = f^\dagger \circ g^\dagger$. In SMOOTH, if $\mathbb{R}^n \xrightarrow{F} \mathbb{R}^m$ is an $\mathbb{R}$-linear map with associated $n \times m$ matrix A, then $F^\dagger$ is the $\mathbb{R}$-linear map given by the transpose matrix A^T, and so τ does indeed plays the role of the matrix transpose operation.

Proposition 4.5 *A linearly closed Cartesian reverse differential category is precisely a linearly closed Cartesian differential category equipped with a linear transpose. Furthermore:*

(i) $f^{\dagger[A]} = \varepsilon_\ell^{\dagger[\mathscr{L}(A,B)]} \circ (\lambda_\ell(f) \times 1)$

(ii) $\lambda_\ell(f^{\dagger[A]}) = \tau \circ \lambda_\ell(f)$

(iii) $\nabla(f) = \tau \circ \mathbf{J}(f)$

(iv) $\nabla(f+g) = \nabla(f) + \nabla(g)$ *and* $\nabla(0) = 0$

(v) *If f is linear then* $\nabla(f) = p_{f^\dagger} \circ 0$

(vi) $\nabla(g \circ f) = \odot \circ \langle \nabla(f), \nabla(g) \circ f \rangle$

PROOF: (Sketch) Starting with a linearly closed Cartesian reverse differential category, first consider the contextual linear dagger of the evaluation map $\mathscr{L}(A,B) \times B \xrightarrow{\varepsilon_\ell^{\dagger[\mathscr{L}(A,B)]}} A$, and define the transpose map as its linear curry $\mathscr{L}(A,B) \xrightarrow{\tau = \lambda_\ell\left(\varepsilon_\ell^{\dagger[\mathscr{L}(A,B)]}\right)} \mathscr{L}(B,A)$ as its linear curry. Then τ is a linear transpose since, as explained above, the linear transpose axioms will follow from the fact that $\mathsf{Lin}[\mathbb{X}]$ is a †-category with †-biproducts. Conversely, starting with a linearly closed Cartesian differential category with a linear transpose τ, it suffices to construct a contextual linear dagger. For a map $A \times B \xrightarrow{f} C$ which is linear in its second argument, we define its dagger as $f^{\dagger[A]} := A \times C \xrightarrow{\lambda(f) \times 1_C} \mathscr{L}(B,C) \times C \xrightarrow{\tau \times 1_C} \mathscr{L}(C,B) \times C \xrightarrow{\varepsilon_\ell} B$. So we obtain a linearly closed Cartesian reverse differential category. Next, (i) follows from the fact that the contextual linear dagger preserves modification to the context, and so: $f^{\dagger[A]} = (\varepsilon_\ell \circ (\lambda_\ell(f) \times 1))^{\dagger[A]} = \varepsilon_\ell^{\dagger[\mathscr{L}(A,B)]} \circ (\lambda_\ell(f) \times 1)$. Then (ii) is simply the linear curry of (i). For (iii), using (ii) and $\mathsf{R}[f] = \mathsf{D}[f]^{\dagger[A]}$, we compute that $\nabla(f) = \lambda_\ell(\mathsf{R}[f]) = \lambda_\ell(\mathsf{D}[f]^{\dagger[A]}) = \tau \circ \lambda_\ell(\mathsf{D}[f]) = \tau \circ \mathbf{J}(f)$. The remaining three identities are simply the transpose versions of the three identities from Proposition 3.8 and the linear transpose axioms (and that if $A \xrightarrow{f} B$ is linear then $\tau \circ p_f = p_{f^\dagger}$ which is a special case of (ii)).   □

We conjecture that an equivalent alternative axiomatization of a linearly closed Cartesian reverse differential category can be given in terms of a Cartesian left additive category equipped with $\mathscr{L}(-,-)$, ε_ℓ, ∇, and τ, where one would define the reverse differential combinator as $\mathsf{R}[-] = \varepsilon_\ell \circ (\nabla(-) \times 1)$. In other future work, it should also be possible to generalize other important notions from classical differential calculus such as the divergence, the curl, the Laplacian, and the Hessian. In fact, recall that the Hessian matrix is defined as the Jacobian matrix of the gradient. Therefore, the Hessian can be defined in a linearly closed Cartesian reverse differential category as $A \xrightarrow{\mathbf{H}(f) := \mathbf{J}(\nabla(f))} \mathscr{L}(A, \mathscr{L}(B,A))$.

References

[1] M. Abadi & G. D. Plotkin (2019): *A simple differentiable programming language*. Proceedings of the ACM on Programming Languages 4(POPL), pp. 1–28.

[2] M. Alvarez-Picallo & C.-H. L. Ong (2019): *Change actions: models of generalised differentiation*. In: *International Conference on Foundations of Software Science and Computation Structures*, Springer, pp. 45–61.

[3] R. F. Blute, J. R. B. Cockett, J.-S. P. Lemay & R. A. G. Seely (2020): *Differential Categories Revisited*. Applied Categorical Structures 28, pp. 171–235, doi:10.1007/s10485-019-09572-y.

[4] R. F. Blute, J. R. B. Cockett & R. A. G. Seely (2006): *Differential categories*. Mathematical structures in computer science 16(06), pp. 1049–1083.

[5] R. F. Blute, J. R. B. Cockett & R. A. G. Seely (2009): *Cartesian differential categories*. Theory and Applications of Categories 22(23), pp. 622–672.

[6] R. F. Blute, J. R. B. Cockett & R. A. G. Seely (2015): *Cartesian differential storage categories*. Theory and Applications of Categories 30(18), pp. 620–686.

[7] R. F. Blute, T. Ehrhard & C. Tasson (2012): *A convenient differential category*. Cahiers de Topologie et Géométrie Différentielle Catégoriques LIII, pp. 211–232.

[8] A. Bucciarelli, T. Ehrhard & G. Manzonetto (2010): *Categorical models for simply typed resource calculi*. Electronic Notes in Theoretical Computer Science 265, pp. 213–230.

[9] J. D. Cockett, J. R. B.; Gallagher (2019): *Categorical models of the differential λ-calculus*. Mathematical Structures in Computer Science, doi:10.1017/S0960129519000070.

[10] J. R. B. Cockett, G. S. H. Cruttwell, J. D. Gallagher, J.-S. P. Lemay, B. MacAdam, G. Plotkin & D. Pronk (2020): *Reverse Derivative Categories*. LIPIcs 152(CSL 2020), pp. 18:1–18:16.

[11] J. R. B. Cockett & J.-S. P. Lemay (2020): *Linearizing Combinators*. arXiv preprint arXiv:2010.15490.

[12] G. Cruttwell, J. Gallagher & D. Pronk (2020): *Categorical semantics of a simple differential programming language*. In: *Proceedings of Applied Category Theory 2020*.

[13] G. Cruttwell, B. Gavranović, N. Ghani, P. Wilson & F. Zanasi (2021): *Categorical Foundations of Gradient-Based Learning*. arXiv preprint arXiv:2103.01931.

[14] T. Ehrhard & L. Regnier (2003): *The differential lambda-calculus*. Theoretical Computer Science 309(1), pp. 1–41.

[15] J. Gallagher & B. MacAdam: *Discussions on Cartesian differential categories with linear homs*. Unpublished.

[16] A. Kriegl & P. W. Michor (1997): *The convenient setting of global analysis*. 53, American Mathematical Soc.

[17] J. Laird, G. Manzonetto & G. McCusker (2013): *Constructing differential categories and deconstructing categories of games*. Information and Computation 222, pp. 247–264.

[18] J.-S. P. Lemay (2018): *A Tangent Category Alternative to the Faa Di Bruno Construction*. Theory and Applications of Categories 33(35), pp. 1072–1110.

[19] G. Manzonetto (2012): *What is a categorical model of the differential and the resource λ-calculi?* Mathematical Structures in Computer Science 22(3), doi:10.1017/S0960129511000594.

[20] O. Manzyuk (2012): *Tangent bundles in differential lambda-categories*. arXiv preprint arXiv:1202.0411.

[21] D. Sprunger & S. Katsumata (2019): *Differentiable causal computations via delayed trace*. In: *2019 34th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, IEEE, pp. 1–12.

[22] M. Vákár (2021): *CHAD: Combinatory Homomorphic Automatic Differentiation*. arXiv preprint arXiv:2103.15776.

[23] P. Wilson & F. Zanasi (2020): *Reverse Derivative Ascent: A Categorical Approach to Learning Boolean Circuits*. In: *Proceedings of Applied Category Theory 2020*.