Reverse Faà di Bruno's Formula for Cartesian Reverse Differential Categories

Aaron Biggin* Jean-Simon Pacaud Lemay†
Macquarie University
Sydney, New South Wales, Australia
aaron.biggin@students.mq.edu.au js.lemay@mq.edu.au

Reverse differentiation is an essential operation for automatic differentiation. Cartesian reverse differential categories axiomatize reverse differentiation in a categorical framework, where one of the primary axioms is the reverse chain rule, which is the formula that expresses the reverse derivative of a composition. Here, we present the reverse differential analogue of Faà di Bruno's Formula, which gives a higher-order reverse chain rule in a Cartesian reverse differential category. To properly do so, we also define partial reverse derivatives and higher-order reverse derivatives in a Cartesian reverse differential category.

1 Introduction

There are two types of derivative operations used in automatic differentiation: **forward differentiation** and the **reverse differentiation** (also referred to as forward mode and reverse mode differentiation). For a smooth function $F : \mathbb{R}^n \to \mathbb{R}^m$, its forward derivative is the usual total derivative and so is of type $\mathsf{D}[F] : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^m$, while its reverse derivative is of type $\mathsf{R}[F] : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$, and is defined in terms of the transpose of the Jacobian of F. As such, the forward derivative and the reverse derivative are the transposes of each other. Forward differentiation is more efficient when the dimension of the output is much larger than the dimension of the input, $n \leq m$, while reverse differentiation is more efficient when the dimension of the input is much larger than the dimension of the output, $n \geq m$. Since many optimization algorithms, especially those used in machine learning, deal with large input data sets that are characterized by functions of type $\mathbb{R}^n \to \mathbb{R}$ where n is quite large, it is the reverse derivative which is often more practical in these setting. Motivated by this, there has been significant interest in studying the categorical foundations of reverse differentiation [2, 5, 7, 8, 14], which fits in the larger program of providing the categorical foundations for machine learning [6, 9, 10, 12, 13, 15].

Cartesian differential categories [1] provide the categorical foundations of forward differentiation, as well as the categorical semantics of the differential λ-calculus. In particular, a Cartesian differential category comes equipped with a **forward differential combinator**, which is an operator that captures forward differentiation of maps. The axioms of a forward differential combinator are analogues of the fundamental identities of the total derivative from differential calculus, such as the famous chain rule for the forward derivative of a composition. On the other hand, Cartesian reverse differential categories provide the categorical foundations of reverse differentiation. This time, a Cartesian reverse differential category comes equipped with a **reverse differential combinator**, which is an operator that captures reverse differentiation of maps. The axioms of the reverse differential combinator are the reverse differential counterparts of those for forward. In particular, one of the axioms is the reverse chain rule for the

*Funded by a Summer Vacation Research Scholarship from Macquarie University
†Funded by an ARC DECRA (DE230100303) and an AFOSR Research Grant (FA9550-24-1-0008)

Submitted to:
SOS 2007

reverse derivative of a composition. Every reverse differential combinator induces a forward differential combinator as well as a transpose operation, called a **contextual linear dagger**, such that the reverse derivative and forward derivative are transposes of one another [2, Thm 42]. As such, every Cartesian reverse differential category is also a Cartesian differential category [2, Thm 16]. Therefore, there is interest in studying and developing the reverse differential counterparts of well-known forward differential concepts in a Cartesian reverse differential category.

Famously, **Faà di Bruno's Formula** provides a higher-order chain rule for the formula of a higher-order forward derivative of a composition. Faà di Bruno's Formula also holds in a Cartesian differential category [11]. The main objective of this paper is to provide the reverse differential version of Faà di Bruno's Formula for a higher-order reverse chain rule. As we will see, the reverse Faà di Bruno's Formula surprisingly involves both reverse derivatives and forward derivatives. To properly give the reverse Faà di Bruno's Formula, we will first need to develop the appropriate notions of **partial reverse derivatives** and **higher-order reverse derivatives** in a Cartesian reverse differential category in such a way that they are the transposes of their respective forward derivative counterparts. So, we will show that the transpose of the partial forward derivative is indeed the partial reverse derivative. However, to also get that the transpose of the higher order forward derivative is the higher order reverse derivative, we introduce an extra compatibility relation between the reverse differential combinator and the forward differential combinator, which we call the stable rule. This additional rule is not very imposing and quite natural. In fact, the stable rule holds in prominent examples of Cartesian reverse differential categories. Then, in the same way that the reverse chain rule can be computed by taking the transpose of the chain rule, by taking the transpose of Faà di Bruno's Formula, we can compute the reverse Faà di Bruno's Formula.

To the best of the authors' knowledge, this is the first expression of Faà di Bruno's Formula for reverse mode differentiation. The reverse Faà di Bruno's Formula will hopefully help improve calculations in (categorical) models for automatic differentiation and machine learning. Moreover, an important application of Faà di Bruno's Formula for Cartesian differential categories is that it is a crucial formula in the construction of *cofree* Cartesian differential categories [4, 11]. Therefore, in future work, it would be interesting to understand if the reverse Faà di Bruno's Formula can be used somehow to construct *cofree* Cartesian reverse differential categories.

2 Reverse Differentiation

In this section, we review Cartesian reverse differential categories and, in particular, also develop the notion *partial* reverse derivatives. Partial reverse differentiation is quite practical and provides a helpful new perspective on the axioms of reverse differentiation and, as we will see in the next section, the relation to forward differentiation and linear transpose operation.

Here, following what was done for forward differentiation in [11], we will introduce the definition of a Cartesian reverse differential category relative to a fixed commutative semiring k, slightly generalizing the original definition from [2]. As such, in this relative setting, the underlying structure of a Cartesian reverse differential category is that of a *Cartesian left k-linear category* [11, Sec 2.1], which can be described as a category with finite products which is *skew*-enriched over the category of k-modules and k-linear maps between them [11]. Essentially, this means that each hom-set is a k-module, so we have zero maps and can take the sum of maps, but also allow for maps which do not preserve zeroes or sums. Maps which do preserve the module structure are called *k-linear maps*. Explicitly, a **left k-linear category** is a category $\mathbb{X}$ such that each hom-set $\mathbb{X}(A,B)$ is a k-module with scalar multiplication

$\cdot : k \times \mathbb{X}(A,B) \to \mathbb{X}(A,B)$, addition $+ : \mathbb{X}(A,B) \times \mathbb{X}(A,B) \to \mathbb{X}(A,B)$, and zero $0 \in \mathbb{X}(A,B)$; and such that pre-composition preserves the k-linear structure: $(r \cdot f + s \cdot g) \circ x = r \cdot (f \circ x) + s \cdot (g \circ x)$. A map $f : A \to B$ is said to be **k-linear** if post-composition by f preserves the k-linear structure: $f \circ (r \cdot x + s \cdot y) = r \cdot (f \circ x) + s \cdot (f \circ y)$. Now for a category with finite products, we denote the product by $\times$, the projection maps by $\pi_j : A_1 \times \ldots \times A_n \to A_j$, and the pairing operation as $\langle -, \ldots, - \rangle$. Then a **Cartesian left k-linear category** is a left k-linear category $\mathbb{X}$ such that $\mathbb{X}$ has finite products and all projection maps π_j are k-linear. We note that when taking $k = \mathbb{N}$, the semiring of natural numbers, (Cartesian) left $\mathbb{N}$-linear categories and their $\mathbb{N}$-linear maps are the same thing as (Cartesian) left additive categories and their additive maps from [1, Def 1.1.1 & 1.2.1].

A Cartesian reverse differential category is a Cartesian k-linear category which comes equipped with a *reverse differential combinator*, which is an operator that sends maps to their reverse derivative. In this relative to k setting, the reverse differential combinator still satisfies essentially the same seven axioms as in [2, Def 13], but where we upgrade the first two **[RD.1]** and **[RD.2]** from simple additivity to k-linearity in the obvious way (so **[RD.3]** to **[RD.7]** are exactly the same). So, in particular, when taking $k = \mathbb{N}$, a Cartesian $\mathbb{N}$-reverse differential category is precisely the same thing as the original definition of a Cartesian reverse differential category from [2, Def 13].

Definition 2.1 *[2, Def 13] A **Cartesian k-reverse differential category** is a Cartesian left k-linear category which comes equipped with a **reverse differential combinator** R, which is a family of functions $\mathsf{R} : \mathbb{X}(A,B) \to \mathbb{X}(A \times B, A)$, satisfying the seven axioms **[RD.1]** to **[RD.7]** described below. For a map $f : A \to B$, the map $\mathsf{R}[f] : A \times B \to B$ is called the **reverse derivative** of f.*

Before we review the axioms of the reverse differential combinator, it may be helpful to first review the canonical example of a Cartesian reverse differential category:

Example 2.2 *Let $\mathbb{R}$ be the set of real numbers. Define* SMOOTH *as the Lawvere theory of real smooth functions, that is, the category whose objects are the Euclidean real vector spaces $\mathbb{R}^n$ and whose maps are real smooth functions $F : \mathbb{R}^n \to \mathbb{R}^m$ between them.* SMOOTH *is a Cartesian $\mathbb{R}$-reverse differential category where for a smooth function $F = \langle f_1, \ldots, f_m \rangle : \mathbb{R}^n \to \mathbb{R}^m$, which recall is in fact a tuple $F = \langle f_1, \ldots, f_m \rangle$ of smooth functions $f_j : \mathbb{R}^n \to \mathbb{R}$, its reverse derivative $\mathsf{R}[F] : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is defined as $\mathsf{R}[F](\vec{x}, \vec{y}) = \left(\sum_{j=1}^{m} \frac{\partial f_j}{\partial x_1}(\vec{x}) y_j, \ldots, \sum_{j=1}^{m} \frac{\partial f_j}{\partial x_n}(\vec{x}) y_j \right)$. In the special case of a smooth function of type $f : \mathbb{R}^n \to \mathbb{R}$, its reverse derivative $\mathsf{R}[f] : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$ is essentially the tuple of partial derivative of f, that is, $\mathsf{R}[f](\vec{x}, y) = \left(\frac{\partial f}{\partial x_1}(\vec{x}) y, \ldots, \frac{\partial f}{\partial x_n}(\vec{x}) y \right)$. In other words, the reverse derivative of f is the gradient of f at the $\mathbb{R}^n$ input vector $\vec{x}$ scalar multiplied by the $\mathbb{R}$ input y.*

Other examples of Cartesian reverse differential categories can be found in [2, 5], which include any dagger category with dagger biproducts, the Lawvere theory of polynomials over a fixed semiring, and the coKleisli category of a monoidal reverse differential category.

So what are the seven axioms **[RD.1]** to **[RD.7]** of a reverse differential combinator? Briefly, they are: **[RD.1]** that the reverse differential combinator is a k-linear morphism; **[RD.2]** reverse derivatives are k-linear in their second argument; **[RD.3]** what the reverse derivative of identities and projections are; **[RD.4]** the reverse derivative of a tuple is the sum of the reverse derivatives of the components of the tuple; **[RD.5]** the reverse chain rule formula for the reverse derivative of a composition; **[RD.6]** reverse derivatives are *differential* linear; and **[RD.7]** is the symmetry of the mixed partial *forward* derivatives. The first five are straightforward to understand, while the last two will become clearer below. For a

more in-depth discussion on the seven reverse differential combinator axioms (and Cartesian reverse differential categories in general), we invite the reader to see [2, 5].

Simply writing down the equations of **[RD.1]** to **[RD.7]** as in [2, Def 13] can be a bit non-intuitive to parse at first glance. Especially the last two axioms **[RD.6]** and **[RD.7]** are a bit complicated to write down and somewhat intimidating at first. To help better understand and read these axioms, inspired by the term-calculus for forward differentiation [1, Sec 4] (which we will review in the next section), we introduce some term-calculus notation for reverse differentiation. We leave providing an actual term calculus for Cartesian reverse differential categories, which is sound and complete, to future work. So in this paper, we write the reverse derivative as $\mathsf{R}[f](a,b) := \frac{\mathsf{r}f(x)}{\mathsf{r}x}(a)\cdot b$. So the first five axioms are:

[RD.1] $\frac{\mathsf{r}s\cdot f(x)+t\cdot g(x)}{\mathsf{r}x}(a)\cdot b = s\cdot\frac{\mathsf{r}f(x)}{\mathsf{r}x}(a)\cdot b + t\cdot\frac{\mathsf{r}g(x)}{\mathsf{r}x}(a)\cdot b$

[RD.2] $\frac{\mathsf{r}f(x)}{\mathsf{d}x}(a)\cdot(r\cdot b+s\cdot c) = r\cdot\frac{\mathsf{r}f(x)}{\mathsf{r}x}(a)\cdot b + s\cdot\frac{\mathsf{r}f(x)}{\mathsf{r}x}(a)\cdot b$

[RD.3] $\frac{\mathsf{r}x}{\mathsf{r}x}(a)\cdot b = b$ and $\frac{\mathsf{r}x_j}{\mathsf{r}(x_1,\ldots,x_n)}(a_0,\ldots,a_n)\cdot b = \underbrace{(0,\ldots,0,b,0,\ldots,0)}_{j\text{th-component}}$

[RD.4] $\frac{\mathsf{r}\langle f_0(x),\ldots,f_n(x)\rangle}{\mathsf{r}x}(a)\cdot(b_0,\ldots,b_n) = \sum_{j=0}^{n}\frac{\mathsf{r}f_j(x)}{\mathsf{r}x}(a)\cdot b_j$

[RD.5] $\frac{\mathsf{r}g(f(x))}{\mathsf{r}x}(a)\cdot b = \frac{\mathsf{r}f(x)}{\mathsf{r}x}(a)\cdot\left(\frac{\mathsf{r}g(y)}{\mathsf{r}y}(f(a))\cdot b\right)$

To help write the last two axioms **[RD.6]** and **[RD.7]**, it will be useful to first introduce the concept of *partial* reverse derivatives.

Given a map of type $f : A_1\times\ldots\times A_n\to B$, we'd like to be able to take the reverse derivative of f with respect to the component A_j while keeping the rest constant. To do so, consider first the total reverse derivative of f which is of type $\mathsf{R}[f] : A_1\times\ldots\times A_n\times B\to A_1\times\ldots\times A_n$. By the universal property of the product, we know that $\mathsf{R}[f]$ is actually a pair of maps of type $A_1\times\ldots\times A_n\times B\to A_j$ which are, of course, defined by post-composing $\mathsf{R}[f]$ with the respective projections. We interpret these maps as the partial reverse derivatives of f.

Definition 2.3 *In a Cartesian k-reverse differential category, for a map $f : A_1\times\ldots\times A_n\to B$, its* ***j-th partial reverse derivative*** *is the map $\mathsf{R}_j[f] : A_1\times\ldots\times A_n\times B\xrightarrow{\mathsf{R}[f]} A_0\times\ldots\times A_n\xrightarrow{\pi_j} A_j$.*

In the term calculus, we write partial reverse derivatives as follows:

$$\frac{\mathsf{r}f(a_1,\ldots,a_{j-1},x_j,a_{j+1},\ldots,a_n)}{\mathsf{r}x_j}(a_j)\cdot b := \pi_j\left(\frac{\mathsf{r}f(x_1,\ldots,x_n)}{\mathsf{r}(x_1,\ldots,x_n)}(a_1,\ldots,a_n)\cdot b\right)$$

Recall that the reverse derivative of a smooth function was the tuple of the partial derivatives. The same idea holds in a Cartesian reverse differential category, which is, of course, immediate from the universal property of the product. We note that this is the reverse differential analogue of the fact that the total forward derivative is equal to the sum of the partial forward derivatives [11, Lemma 2.8.(i)].

Lemma 2.4 *In a Cartesian k-reverse differential category, for a map $f : A_1\times\ldots\times A_n\to B$, we have that $\mathsf{R}[f] = \langle\mathsf{R}_1[f],\ldots,\mathsf{R}_n[f]\rangle$, which in the term calculus is expressed as:*

$$\begin{aligned}&\frac{\mathsf{r}f(x_1,\ldots,x_n)}{\mathsf{r}(x_1,\ldots,x_n)}(a_1,\ldots,a_n)\cdot b\\ &= \left\langle\frac{\mathsf{r}f(x_1,a_2,\ldots,a_n)}{\mathsf{r}x_1}(a_1)\cdot b,\ldots,\frac{\mathsf{r}f(a_1,\ldots,x_j,\ldots,a_n)}{\mathsf{r}x_j}(a_j)\cdot b,\ldots,\frac{\mathsf{r}f(a_1,\ldots,x_n)}{\mathsf{r}x_n}(a_n)\cdot b\right\rangle\end{aligned}\tag{1}$$

Now that we have partial reverse derivatives, we can smoothly write down the remaining two axioms:

[RD.6] $$\frac{\mathsf{r}\dfrac{\mathsf{r}f(x)}{\mathsf{r}x}(a)\cdot u}{\dfrac{\mathsf{r}u}{\mathsf{r}x}(0)\cdot v}(0)\cdot b = \frac{\mathsf{r}f(x)}{\mathsf{r}x}(a)\cdot b$$

[RD.7] $$\frac{\mathsf{r}\dfrac{\mathsf{r}\dfrac{\mathsf{r}f(x)}{\mathsf{r}x}(y)\cdot u}{\dfrac{\mathsf{r}u}{\mathsf{r}y}(0)\cdot b}(a)\cdot v}{\mathsf{r}v}(0)\cdot c = \frac{\mathsf{r}\dfrac{\mathsf{r}\dfrac{\mathsf{r}f(x)}{\mathsf{r}x}(y)\cdot u}{\dfrac{\mathsf{r}u}{\mathsf{r}y}(0)\cdot c}(a)\cdot v}{\mathsf{r}v}(0)\cdot b$$

While these axioms may still look somewhat strange initially, they will make more sense when considering the relationship between reverse differentiation and forward differentiation.

Another way of describing partial reverse differentiation is as reverse differentiation in context. From this point of view, partial reverse differentiation is an actual reverse differential combinator for *simple slice categories*. To prove this, it amounts to showing that partial reverse differentiation satisfies the seven reverse differential combinator axioms in context.

Lemma 2.5 *In a Cartesian k-reverse differential category, the following equalities hold:*

[RD.1] $$\frac{\mathsf{r}s\cdot f(c_1,x,c_2)+t\cdot g(c_1,x,c_2)}{\mathsf{r}x}(a)\cdot b = s\cdot\frac{\mathsf{r}f(c_1,x,c_2)}{\mathsf{r}x}(a)\cdot b + t\cdot\frac{\mathsf{r}g(c_1,x,c_2)}{\mathsf{r}x}(a)\cdot b$$

[RD.2] $$\frac{\mathsf{r}f(c_1,x,c_2)}{\mathsf{d}x}(a)\cdot(r\cdot b+s\cdot c) = r\cdot\frac{\mathsf{r}f(c_1,x,c_2)}{\mathsf{r}x}(a)\cdot b + s\cdot\frac{\mathsf{r}f(c_1,x,c_2)}{\mathsf{r}x}(a)\cdot b$$

[RD.3] $\frac{\mathsf{r}x_j}{\mathsf{r}x_j}(a)\cdot b = b$ *and* $\frac{\mathsf{r}x_j}{\mathsf{r}x_i}(a)\cdot b = 0$ *if* $i\neq j$;

[RD.4] $$\frac{\mathsf{r}\left\langle f_0(c_1,x,c_2),\ldots,f_n(c_1,x,c_2)\right\rangle}{\mathsf{r}x}(a)\cdot(b_0,\ldots,b_n) = \sum_{j=0}^{n}\frac{\mathsf{r}f_j(c_1,x,c_2)}{\mathsf{r}x}(a)\cdot b_j$$

[RD.5] $$\frac{\mathsf{r}g\left(c_1,f(c_1,x,c_2),c_2\right)}{\mathsf{r}x}(a)\cdot b = \frac{\mathsf{r}f(c_1,x,c_2)}{\mathsf{r}x}(a)\cdot\left(\frac{\mathsf{r}g(c_1,y,c_2)}{\mathsf{r}y}(f(c_1,a,c_2))\cdot b\right)$$

[RD.6] $$\frac{\mathsf{r}\dfrac{\mathsf{r}f(c_1x,c_2)}{\mathsf{r}x}(a)\cdot u}{\dfrac{\mathsf{r}u}{\mathsf{r}x}(0)\cdot v}(0)\cdot b = \frac{\mathsf{r}f(c_1,x,c_2)}{\mathsf{r}x}(a)\cdot b$$

[RD.7] $$\frac{\mathsf{r}\dfrac{\mathsf{r}\dfrac{\mathsf{r}f(c_1,x,c_2)}{\mathsf{r}x}(y)\cdot u}{\dfrac{\mathsf{r}u}{\mathsf{r}y}(0)\cdot b}(a)\cdot v}{\mathsf{r}v}(0)\cdot c = \frac{\mathsf{r}\dfrac{\mathsf{r}\dfrac{\mathsf{r}f(c_1,x,c_2)}{\mathsf{r}x}(y)\cdot u}{\dfrac{\mathsf{r}u}{\mathsf{r}y}(0)\cdot c}(a)\cdot v}{\mathsf{r}v}(0)\cdot b$$

PROOF: To prove these, we first need to compute the following useful identity:

$$\begin{aligned}
&\frac{\mathsf{r}\left\langle c_1,f(v,x,u),c_2\right\rangle}{\mathsf{r}x}(a)\cdot(b_1,b_2,b_3) \overset{=}{\underset{\text{Def.}}{}} \pi_2\left(\frac{\mathsf{r}\left\langle v,f(v,x,u),u\right\rangle}{\mathsf{r}(v,x,u)}(c_1,a,c_2)\cdot(b_1,b_2,b_3)\right)\\
&\overset{=}{\underset{\textbf{[R.4]}}{}}\ \pi_2\left(\frac{\mathsf{r}\left\langle v\right\rangle}{\mathsf{r}(v,x,u)}(c_1,a,c_2)\cdot b_1+\frac{\mathsf{r}f(v,x,u)}{\mathsf{r}(v,x,u)}(c_1,a,c_2)\cdot b_2+\frac{\mathsf{r}\left\langle u\right\rangle}{\mathsf{r}(v,x,u)}(c_1,a,c_2)\cdot b_3\right)\\
&\overset{=}{\underset{\textbf{[R.3]}+(1)}{}}\ \pi_2\left((b_1,0,0)+\left(\frac{\mathsf{r}f(v,a,c_2)}{\mathsf{r}v}(c_1)\cdot b_2,\frac{\mathsf{r}f(c_1,x,c_2)}{\mathsf{r}x}(a)\cdot b_2,\frac{\mathsf{r}f(c_1,a,u)}{\mathsf{r}u}(c_2)\cdot b_2\right)+(0,0,b_3)\right)
\end{aligned}$$

$$\pi_j \text{ is } \overset{=}{k}\text{-lin.} \;\; \pi_2(b_1,0,0)+\pi_2\left(\frac{\mathsf{r}f(v,a,c_2)}{\mathsf{r}v}(c_1)\cdot b_2,\frac{\mathsf{r}f(c_1,x,c_2)}{\mathsf{r}x}(a)\cdot b_2,\frac{\mathsf{r}f(c_1,a,u)}{\mathsf{r}u}(c_2)\cdot b_2\right)+\pi_2(0,0,b_3)$$
$$=0+\frac{\mathsf{r}f(c_1,x,c_2)}{\mathsf{r}x}(a)\cdot b_2+0=\frac{\mathsf{r}f(c_1,x,c_2)}{\mathsf{r}x}(a)\cdot b_2$$

So we have that:

$$\frac{\mathsf{r}\langle c_1,f(v,x,u),c_2\rangle}{\mathsf{r}x}(a)\cdot(b_1,b_2,b_3)=\frac{\mathsf{r}f(c_1,x,c_2)}{\mathsf{r}x}(a)\cdot b_2 \tag{2}$$

Using this and the corresponding axiom **[RD.#]**, we can prove the corresponding partial reverse version. Let's prove the reverse chain rule in context:

$$\frac{\mathsf{r}g(c_1,f(c_1,x,c_2),c_2)}{\mathsf{r}x}(a)\cdot b \overset{=}{\underset{\text{Def.}}{}} \pi_2\left(\frac{\mathsf{r}g(v,f(v,x,u),u)}{\mathsf{r}(v,x,u)}(c_1,a,c_2)\cdot b\right)$$
$$\overset{=}{\underset{\textbf{[R.5]}}{}} \pi_2\left(\frac{\mathsf{r}\langle v,f(v,x,u),u\rangle}{\mathsf{r}(v,x,u)}(c_1,a,c_2)\cdot\left(\frac{\mathsf{r}g(w,y,z)}{\mathsf{r}(w,y,z)}(c_1,f(c_1,a,c_2),c_2)\cdot b\right)\right)$$
$$\overset{=}{\underset{\text{Def.}}{}} \frac{\mathsf{r}\langle c_1,f(v,x,u),c_2\rangle}{\mathsf{r}x}(a)\cdot\left(\frac{\mathsf{r}g(w,y,z)}{\mathsf{r}(w,y,z)}(c_1,f(c_1,a,c_2),c_2)\cdot b\right)$$
$$\overset{=}{\underset{(1)}{}} \frac{\mathsf{r}\langle c_1,f(v,x,u),c_2\rangle}{\mathsf{r}x}(a)\cdot\left(\frac{\mathsf{r}g(w,f(c_1,a,c_2),c_2)}{\mathsf{r}w}(c_1)\cdot b,\frac{\mathsf{r}g(c_1,y,c_2)}{\mathsf{r}y}(f(c_1,a,c_2))\cdot b,\frac{\mathsf{r}g(c_1,f(c_1,a,c_2),z)}{\mathsf{r}z}(c_2)\cdot b\right)$$
$$\overset{=}{\underset{(2)}{}} \frac{\mathsf{r}f(c_1,x,c_2)}{\mathsf{r}x}(a)\cdot\frac{\mathsf{r}g(c_1,y,c_2)}{\mathsf{r}y}(f(c_1,a,c_2))\cdot b$$

The other identities are show in similar fashion. □

Now recall that for a category $\mathbb{X}$ with finite products, for each object $C\in\mathbb{X}$, the simple slice category over C is the category $\mathbb{X}[C]$ whose objects are the same as $\mathbb{X}$ and whose homsets are $\mathbb{X}[C](A,B)=\mathbb{X}(C\times A,B)$. Composition of $f:C\times A\to B$ and $g:C\times B\to D$ is $g\circ\langle\pi_1,f\rangle$, which is written in the term calculus as $g(c,f(c,x))$, and the identity map is the projection $\pi_2:C\times A\to B$. $\mathbb{X}[C]$ also has finite products, and if $\mathbb{X}$ is a Cartesian left k-linear category, then $\mathbb{X}[C]$ will also be a Cartesian k-linear category with the same k-linear structure as $\mathbb{X}$. Then it follows from the above lemma that the simple slice categories of a Cartesian reverse differential category are again Cartesian reverse differential categories. This is the reverse differentiation analogue of [1, Cor 4.5.2].

Corollary 2.6 *Let $\mathbb{X}$ be a Cartesian k-reverse differential category. Then for each object $C\in\mathbb{X}$, the simple slice category $\mathbb{X}[C]$ is a Cartesian k-reverse differential category whose reverse differential combinator R^C sends a map $f:C\times A\to B$ to the map $\mathsf{R}^C[f]:C\times A\times B\xrightarrow{\mathsf{R}[f]}C\times A\xrightarrow{\pi_2}A$.*

PROOF: The necessary reverse differential combinator axioms with respect to composition in the simple slice category are precisely the identities from Lemma 2.5 with context only on the left. □

3 Forward Differentiation and Linear Transpose

The fundamental theorem about Cartesian reverse differential categories is that they are precisely the same as Cartesian *forward* differential categories equipped with a suitable notion of transpose called a *contextual linear dagger*. In particular, this says we obtain forward differentiation and a transpose operation from reverse differentiation, and vice-versa. We will not thoroughly review Cartesian differential

categories and contextual linear daggers here. So for a more in-depth introduction to Cartesian forward differential categories, we invite the reader to see [1, 11], and for an in-depth introduction to contextual linear daggers, we invite the reader to see [2, 5]. Instead, we will focus on revisiting how to build a forward differential combinator and contextual linear dagger from a reverse differential combinator. We will also see how these constructions can be nicely expressed using partial reverse derivatives.

Theorem 3.1 *[2, Thm 42] A Cartesian k-reverse differential category is precisely the same thing as a Cartesian k-differential category with a contextual k-linear dagger.*

PROOF: The proof in the relative to k setting is essentially the same as was done in [2]. □

Briefly, a **Cartesian k-differential category** [11, Sec 2.2] is a Cartesian k-linear category which comes equipped with a **forward differential combinator**, which is an operator which sends maps $f : A \to B$ to their **forward derivative** $\mathsf{D}[f] : A \times A \to B$. The differential combinator axioms are analogues of the well-known identities of the total derivative from differential calculus, such as the chain rule. As previously mentioned, Cartesian differential categories have a very practical term calculus [1, Sec 4]. So we write the forward derivative as $\mathsf{D}[f](a,b) := \dfrac{\mathsf{d}f(x)}{\mathsf{d}x}(a) \cdot b$.

Let's now revisit how to go from reverse differentiation to forward differentiation by rewriting the construction of [2, Thm 16] using partial reverse derivatives. So consider a map $f : A \to B$, then we need to build a map of type $A \times A \to B$. We can first take its reverse derivative to get $\mathsf{R}[f] : A \times B \to A$, and then by taking the partial reverse derivative with respect to B, we get a map of type $\mathsf{R}_2[\mathsf{R}[f]] : A \times B \times A \to B$. Then inserting zero into the B argument of the domain gives us a map of type $A \times A \to B$ as desired. Explicitly, for a map $f : A \to B$, its forward derivative is defined as follows:

$$\frac{\mathsf{d}f(x)}{\mathsf{d}x}(a) \cdot b := \frac{\mathsf{r}\dfrac{\mathsf{r}f(x)}{\mathsf{r}x}(a) \cdot y}{\mathsf{r}y}(0) \cdot b \tag{3}$$

Then **[RD.7]** is precisely expressing **[CD.7]** but written using only reverse differentiation.

Before defining the transpose operation, we first need to quickly discuss *partial* forward derivatives and *differential linearity*. Starting with partial forward derivatives: given a map $f : A_0 \times \ldots \times A_n \to B$, we'd like to take the forward derivative of f with respect to the component A_j while keeping the rest constant. In differential calculus, partial derivatives are obtained by inserting zeroes in the appropriate vector argument of the total derivative. The same idea holds in a Cartesian differential category. Then the **j-th partial forward derivative** [11, Def 2.7] of f is the map $\mathsf{D}_j[f] : A_1 \times \ldots \times A_n \times A_j \to B$ is written in term calculus on the left below and is defined as follows:

$$\frac{\mathsf{d}f(a_1,\ldots,a_{j-1},x_j,a_{j+1},\ldots,a_n)}{\mathsf{d}x_j}(a_j) \cdot b := \frac{\mathsf{d}f(x_1,\ldots,x_n)}{\mathsf{d}(x_1,\ldots,x_n)}(a_1,\ldots,a_n) \cdot (0,\ldots,0,b,0,\ldots,0)$$

Now a map is D-linear in an argument if when forward differentiating with respect to that argument, one gets back the starting map. Explicitly, a map $f : A_1 \times \ldots \times A_n \to B$ is said to be **differential linear (D-linear)** [11, def 2.6] in A_j if when taking its j-th partial derivative, the following equality holds:

$$\frac{\mathsf{d}f(a_1,\ldots,a_{j-1},x_j,a_{j+1},\ldots,a_n)}{\mathsf{d}x_j}(a_j) \cdot b = f(a_1,\ldots,a_{j-1},b,a_{j+1},\ldots,a_n)$$

Properties of differential linear maps can be found in [3, Lemma 2.6], such as the fact that they are closed under composition, k-linear structure, etc. In particular, if f is D-linear in its j-th variable, then it is also

k-linear in its j-th variable [3, Lemma 2.6.(i)] (though the converse is not necessarily true in an arbitrary Cartesian differential category). Moreover, the axiom **[CD.6]** of the forward differential combinator is precisely the statement that forward derivatives are D-linear in their second argument.

A **contextual linear dagger** [2, Def 39] is an involutive and contravariant operation on maps with a D-linear argument, which swaps the codomain with said D-linear argument. Here we will write down the contextual linear dagger with context both on the left and right, which is equivalent to simply having context on the left as in [2]. So for a map $f : C_1 \times A \times C_2 \to B$ which is D-linear in A, the D**-linear transpose in** A of f is the partial reverse derivative in A of f evaluated at 0, that is, the map $f^{\dagger[C_1 \times \_ \times C_2]} : C_1 \times A \times C_2 \times B \to A$ defined as follows:

$$f^{\dagger[C_1 \times \_ \times C_2]}(c_1, b, c_2) := \frac{\mathsf{r}f(c_1, x, c_2)}{\mathsf{r}x}(0) \cdot b \tag{4}$$

Now $f^{\dagger[C_1 \times \_ \times C_2]} : C_1 \times B \times C_2$ is D-linear in B [2, Cor 36] and this operation is involutive [2, Lemma 35], that is, the D-linear transpose of $f^{\dagger[C_1 \times \_ \times C_2]}$ is f. Other properties of the contextual linear dagger can be found in [2, 5]. In particular, for the reverse Faà di Bruno's Formula, it will be useful to recall the fact the contextual linear dagger is contravariant with respect to composition in context.

Lemma 3.2 *In a Cartesian k-reverse differential category, if $f : C_1 \times A \times C_2 \to B$ is D-linear in A and $g : C_1 \times B \times C_2 \to E$ is D-linear in B, then the following equality holds:*

$$\frac{\mathsf{r}g(c_1, f(c_1, x, c_2), c_2)}{\mathsf{r}x}(0) \cdot b = \frac{\mathsf{r}f(c_1, x, c_2)}{\mathsf{r}x}(0) \cdot \left(\frac{\mathsf{r}g(c_1, y, c_2)}{\mathsf{r}y}(0) \cdot b \right) \tag{5}$$

PROOF: The proof is essentially the same as the equational calculations done in the proof of [2, Thm 37]. However, since it is crucial to proving the reverse Faà di Bruno's Formula, revisiting the proof using the term calculus may be useful. So first recall that since f is D-linear in A, it is also k-linear in A. So in particular $f(c_1, 0, c_2) = 0$. So then we compute:

$$\begin{aligned} \frac{\mathsf{r}g(c_1, f(c_1, x, c_2), c_2)}{\mathsf{r}x}(0) \cdot b &\underset{\textbf{[R.5]}}{=} \frac{\mathsf{r}f(c_1, x, c_2)}{\mathsf{r}x}(0) \cdot \left(\frac{\mathsf{r}g(c_1, y, c_2)}{\mathsf{r}y}(f(c_1, 0, c_2)) \cdot b \right) \\ &\underset{\mathsf{D}\text{-lin.} \Rightarrow k\text{-lin.}}{=} \frac{\mathsf{r}f(c_1, x, c_2)}{\mathsf{r}x}(0) \cdot \left(\frac{\mathsf{r}g(c_1, y, c_2)}{\mathsf{r}y}(0) \cdot b \right) \end{aligned}$$

□

Now [2, Thm 42] tells us that the D-linear transpose of the forward derivative is the reverse derivative, that is, $\mathsf{D}[f]^{\dagger[A \times \_]} = \mathsf{R}[f]$, which in term calculus is written as:

$$\frac{\mathsf{r}\dfrac{\mathsf{d}f(x)}{\mathsf{d}x}(a) \cdot u}{\mathsf{r}u}(0) \cdot b = \frac{\mathsf{r}f(x)}{\mathsf{r}x}(a) \cdot b \tag{6}$$

which we note is precisely **[RD.6]**. Moreover, this also implies that $\mathsf{R}[f]$ is D-linear in its second argument. Furthermore, we also get that the D-linear transpose of the reverse derivative is the forward derivative, that is, $\mathsf{R}[f]^{\dagger[A \times \_]} = \mathsf{D}[f]$. Moreover, we can also show that the partial reverse derivative in the second argument of the reverse derivative is precisely the forward derivative:

Lemma 3.3 *In a Cartesian k-reverse differential category, the following equalities hold:*

$$\frac{\mathsf{r}\dfrac{\mathsf{r}f(x)}{\mathsf{r}x}(a_1) \cdot u}{\mathsf{r}u}(b) \cdot a_2 = \frac{\mathsf{d}f(x)}{\mathsf{d}x}(a_1) \cdot a_2 \tag{7}$$

PROOF: We compute:

$$\frac{\mathsf{r}\dfrac{\mathsf{r}f(x)}{\mathsf{r}x}(a_1)\cdot u}{\mathsf{r}u}(b)\cdot a_2 \underset{(6)}{=} \frac{\mathsf{r}\dfrac{\mathsf{d}\dfrac{\mathsf{r}f(x)}{\mathsf{r}x}(a_1)\cdot u}{\mathsf{d}u}(b)\cdot v}{\mathsf{r}v}(0)\cdot a_2 \underset{\text{D-lin.}}{=} \frac{\mathsf{r}\dfrac{\mathsf{r}f(x)}{\mathsf{r}x}(a_1)\cdot v}{\mathsf{r}v}(0)\cdot a_2 \underset{(3)}{=} \frac{\mathsf{d}f(x)}{\mathsf{d}x}(a_1)\cdot a_2$$

□

We conclude this section by showing that the partial reverse derivative is the D-linear transpose of the partial forward derivative.

Lemma 3.4 *In a Cartesian k-reverse differential category, for every map $f : A_1 \times \ldots \times A_n \to B$, we have that $\mathsf{D}_j[f]^{\dagger[A_1\times\ldots\times A_n\times\_]} = \mathsf{R}_j[f]$, which in the term calculus is expressed as follows:*

$$\frac{\mathsf{r}\dfrac{\mathsf{d}f(a_1,\ldots,a_{j-1},x_j,a_{j+1},\ldots,a_n)}{\mathsf{d}x_j}(a_j)\cdot u}{\mathsf{r}u}(0)\cdot b = \frac{\mathsf{r}f(a_1,\ldots,a_{j-1},x_j,a_{j+1},\ldots,a_n)}{\mathsf{r}x_j}(a_j)\cdot b \tag{8}$$

PROOF: WLOG, for simplicity, we prove this identity for a map of type $f : C_1 \times A \times C_2 \to B$. So we compute that:

$$\frac{\mathsf{r}\dfrac{\mathsf{d}f(c_1,x,c_2)}{\mathsf{d}x}(a)\cdot u}{\mathsf{r}u}(0)\cdot b \underset{\text{Def.}}{=} \frac{\mathsf{r}\dfrac{\mathsf{d}f(v,x,w)}{\mathsf{d}(v,x,u)}(c_1,a,c_2)\cdot(0,u,0)}{\mathsf{r}u}(0)\cdot b$$

$$\underset{(3)}{=} \frac{\mathsf{r}\dfrac{\mathsf{r}f(v,x,w)}{\mathsf{r}(v,x,w)}(c_1,a,c_2)\cdot y}{\mathsf{r}y}(0)\cdot(0,u,0)}{\mathsf{r}u}(0)\cdot b \underset{\text{Def.}}{=} \pi_2\left(\frac{\mathsf{r}\dfrac{\mathsf{r}f(v,x,w)}{\mathsf{r}(v,x,w)}(c_1,a,c_2)\cdot y}{\mathsf{r}y}(0)\cdot(s,u,t)}{\mathsf{r}(s,u,t)}(0,0,0)\cdot b\right)$$

$$\underset{\textbf{[R.6]}}{=} \pi_2\left(\frac{\mathsf{r}f(v,x,w)}{\mathsf{r}(v,x,w)}(c_1,a,c_2)\cdot b\right) \underset{\text{Def.}}{=} \frac{\mathsf{r}f(c_1,x,c_2)}{\mathsf{r}x}(a)\cdot b$$

□

4 Reverse Faà di Bruno's Formula

To provide a reverse differentiation version of Faà di Bruno's Formula, we must first workout the appropriate the notion of *higher-order reverse derivatives*. Of course, given a map $f : A \to B$, we first take its reverse derivative $\mathsf{R}[f] : A \times B \to A$, reverse deriving it again gives $\mathsf{R}^2[f] : A \times B \times A \to A \times B$, then reverse deriving again gives $\mathsf{R}^3[f] : A \times B \times A \times A \times B \to A \times B \times A$, so on and so forth. So we quickly see that both the domain and codomain of $\mathsf{R}^n[f]$ expand quite rapidly, and thus $\mathsf{R}^n[f]$ is not necessarily easy to work with. However, it turns out that from Lemma 2.4 and Lemma 3.3, there is a lot of redundant information in $\mathsf{R}^n[f]$. For example, we can compute that $\mathsf{R}^2[f]$ is worked out to be:

$$\frac{\mathsf{r}\dfrac{\mathsf{r}f(x)}{\mathsf{r}x}(y)\cdot z}{\mathsf{r}(y,z)}(a_1,b)\cdot a_2 = \left\langle \frac{\mathsf{r}\dfrac{\mathsf{r}f(x)}{\mathsf{r}x}(y)\cdot b}{\mathsf{r}y}(a_1)\cdot a_2, \frac{\mathsf{d}f(x)}{\mathsf{d}x}(a_1)\cdot a_2 \right\rangle = \left\langle \frac{\mathsf{r}\dfrac{\mathsf{r}f(x)}{\mathsf{r}x}(a_1)\cdot u}{\mathsf{r}u}(b)\cdot a_2, \frac{\mathsf{d}f(x)}{\mathsf{d}x}(a_1)\cdot a_2 \right\rangle$$

So we see that $\mathsf{R}^2[f]$ contains previous known information, since $\mathsf{D}[f]$ is the D-linear transpose of $\mathsf{R}[f]$. So all the new information comes from taking the partial reverse derivative in the first argument. This is how we get the higher-order reverse derivatives. So starting with a map $f : A \to B$, we again first take its reverse derivative $\mathsf{R}[f] : A \times B \to A$, and then this time only take the partial reverse derivative in the first argument to get $\mathsf{R}_1[\mathsf{R}[f]] : A \times B \times A \to A$, then do this again to get $\mathsf{R}_1[\mathsf{R}_1[\mathsf{R}[f]]] : A \times B \times A \times A \to A$, and so on. So after doing this $n+1$ times, we get a map of type $A \times B \times A^{\times^n} \to A$.

Definition 4.1 *In a Cartesian k-reverse differential category, for a map $f : A \to B$, the $n+1$-**th reverse derivative** of f is the map $\rho^{(n+1)}[f] : A \times B \times A^{\times^n} \to A$ defined inductively as $\rho^{(1)}[f] = \mathsf{R}[f]$ and $\rho^{(n+2)}[f] = \mathsf{R}_1\left[\rho^{(n+1)}[f]\right]$.*

In the term calculus, we write higher-order reverse derivatives as follows:

$$\rho^{(n+1)}(a_0, b, a_2, \ldots, a_{n+1}) = \frac{\mathsf{r}^{(n+1)} f(x)}{\mathsf{r}x}(a_0) \cdot b \cdot a_2 \cdot \ldots \cdot a_{n+1}$$

Higher-order forward derivatives are defined similarly. Starting again with a map $f : A \to B$, we can repeatedly derive the first argument to get a map of type $A \times A^{\times^{n+1}} \to B$. So the $n+1$**-th forward derivative** [11, Def 3.1] of f is the map $\partial^{(n+1)}[f] : A \times A^{\times^n} \to B$, which is defined inductively as $\partial^{(1)}[f] = \mathsf{D}[f]$ and $\partial^{(n+2)}[f] = \mathsf{D}_1\left[\partial^{(n+1)}[f]\right]$. In the term calculus, we write higher-order forward derivatives as:

$$\partial^{(n+1)}[f](a_0, a_1, \ldots, a_{n+1}) := \frac{\mathsf{d}^{(n)} f(x)}{\mathsf{d}f(x)}(a_0) \cdot a_1 \cdot \ldots \cdot a_{n+1}$$

Now the $n+1$-th forward derivative is D-linear in each of the last $n+1$ arguments A and is also symmetric in its last $n+1$ arguments [11, Lemma 3.1.(i)]. As such, it does not matter which D-linear argument we transpose. So for convenience, to line up with the type higher-order reverse derivative, we will consider the D-linear transpose of the second argument of the higher-order forward derivative: $\partial^{(n+1)}[f]^{\dagger[A\times\_\times A^n]} : A \times B \times A^{\times^n} \to A$. Now that we'd like that $\partial^{(n+1)}[f]^{\dagger[A\times\_\times A^n]}$ be equal to $\rho^{(n+1)}$. Unfortunately, this does not seem to follow from just **[RD.1]**-**[RD.7]**. As such, we introduce an extra compatibility between the forward differential combinator and the reverse differential combinator.

Given a map $f : A \to B$, we can take its forward derivative $\mathsf{D}[f] : A \times A \to B$, and then take its reverse derivative in the first argument to get $\mathsf{R}_1[\mathsf{D}[f]] : A \times A \times B \to A$. On the other hand, we could have first taken the reverse derivative $\mathsf{R}[f] : A \times B \to A$, and then take its reverse derivative in the first argument to get $\mathsf{R}_1[\mathsf{R}[f]] : A \times B \times A \to A$. Up to swapping the last two arguments, we ask that these two are equal.

Definition 4.2 *A Cartesian k-reverse differential category is said to satisfy the **stable rule** if:*

$$\frac{\mathsf{r}\dfrac{\mathsf{d}f(x)}{x}(y) \cdot a_2}{\mathsf{r}y}(a_1) \cdot b = \frac{\mathsf{r}\dfrac{\mathsf{r}f(x)}{x}(y) \cdot b}{\mathsf{d}y}(a_1) \cdot a_2 \tag{9}$$

The stable rule holds in the main examples of Cartesian reverse differential categories. The stable rule is straightforward to check for either smooth functions or polynomials (see App. A). One can also check that the stable rule holds in the coKleisli category of a monoidal reverse differential category using string diagrams.

With the stable rule, we can show that D-linear transpose of the higher-order forward derivative is the higher-order reverse derivative. For convenience, by convention we define $\rho^{(0)}[f] = f = \partial^{(0)}[f]$.

Lemma 4.3 *In a Cartesian k-reverse differential category whose reverse differential combinator satisfies the stable rule, we have that* $\partial^{(n+1)}[f]^{\dagger[A\times\_\times A^n]} = \rho^{(n+1)}[f]$*, which in the term calculus is expressed as:*

$$\frac{\mathsf{r}\dfrac{\mathsf{d}^{(n+1)}f(x)}{\mathsf{d}x}(a_0)\cdot y\cdot a_2\cdot\ldots\cdot a_{n+1}}{\mathsf{r}y}(0)\cdot b = \frac{\mathsf{r}^{(n+1)}f(x)}{\mathsf{r}x}(a_0)\cdot b\cdot a_2\cdot\ldots\cdot a_{n+1} \tag{10}$$

PROOF: One can show that the stable rule also holds in context, that is:

$$\frac{\mathsf{r}\dfrac{\mathsf{d}g(c_1,x,c_2)}{x}(y)\cdot a_2}{\mathsf{r}y}(a_1)\cdot b = \frac{\mathsf{r}\dfrac{\mathsf{r}g(c_1,x,c_2)}{x}(y)\cdot b}{\mathsf{d}y}(a_1)\cdot a_2 \tag{11}$$

Now note that the stable rule can be rewritten as:

$$\frac{\mathsf{r}\dfrac{\mathsf{d}f(x)}{x}(y)\cdot a_2}{\mathsf{r}y}(a_1)\cdot b = \frac{\mathsf{r}^{(2)}f(x)}{\mathsf{d}(x)}(a_1)\cdot b\cdot\cdot a_2 \tag{12}$$

Since the stable rule holds in context, it is straightforward to show by induction that:

$$\frac{\mathsf{r}\dfrac{\mathsf{d}^{(n)}f(x)}{x}(y)\cdot a_1\cdot\ldots\cdot a_n}{\mathsf{r}y}(a_0)\cdot b = \frac{\mathsf{r}^{(n+1)}f(x)}{\mathsf{r}x}(a_0)\cdot b\cdot a_2\cdot\ldots\cdot a_{n+1} \tag{13}$$

Then applying (6) to the left-hand side gives us precisely that:

$$\begin{aligned}\frac{\mathsf{r}\dfrac{\mathsf{d}^{(n+1)}f(x)}{\mathsf{d}x}(a_0)\cdot y\cdot a_2\cdot\ldots\cdot a_{n+1}}{\mathsf{r}y}(0)\cdot b &\underset{\text{Def.}}{=} \frac{\mathsf{d}\dfrac{\mathsf{r}\dfrac{\mathsf{d}^{(n)}f(x)}{x}(y)\cdot a_1\cdot\ldots\cdot a_n}{\mathsf{d}y}(a_0)\cdot y}{\mathsf{r}y}(0)\cdot b\\ &\underset{(6)}{=} \frac{\mathsf{r}\dfrac{\mathsf{d}^{(n)}f(x)}{x}(y)\cdot a_1\cdot\ldots\cdot a_n}{\mathsf{r}y}(a_0)\cdot b \underset{(13)}{=} \frac{\mathsf{r}^{(n+1)}f(x)}{\mathsf{r}x}(a_0)\cdot b\cdot a_2\cdot\ldots\cdot a_{n+1}\end{aligned}$$

So we get that $\partial^{(n+1)}[f]^{\dagger[A\times\_\times A^n]} = \rho^{(n+1)}[f]$ as desired. □

Now that we have that the higher-order reverse derivative is the D-linear transpose of the higher-order forward derivative, we may now look towards understanding the reverse Faà di Bruno's Formula. To do so, let us first review Faà di Bruno's Formula for forward differentiation in a Cartesian differential category. Let us first introduce some notation. For every $n\in\mathbb{N}$, define the well-ordered set $[n+1]=\{1<\ldots<n+1\}$. Now for every subset $I=\{i_1<\ldots<i_m\}\subseteq[n+1]$, for a vector $\vec{x}=(x_1,\ldots,x_{n+1})$, define $\vec{x}|_I=(x_{i_1},\ldots,x_{i_m})$. Lastly, we denote a *non-empty* partition of $[n+1]$ as $[n+1]=A_1|\ldots|A_k$, and let $|A_j|$ be the cardinality of A_j. Then Faà di Bruno's Formula [11, Lemma 3.14] for the $n+1$-th derivative is given as a sum over the non-empty partitions of $[n+1]$ as follows:

$$\begin{aligned}&\frac{\mathsf{d}^{(n+1)}g(f(x))}{\mathsf{d}x}(a_0)\cdot a_1\cdot a_2\cdot\ldots\cdot a_{n+1}\\ =&\sum_{[n+1]=A_1|\ldots|A_k}\frac{\mathsf{d}^{(k)}g(z)}{\mathsf{d}z}(f(a_0))\cdot\left(\frac{\mathsf{d}^{(|A_1|)}f(x)}{\mathsf{d}x}(a_0)\cdot\vec{a}|_{A_1}\right)\cdot\ldots\cdot\left(\frac{\mathsf{d}^{(|A_k|)}f(x)}{\mathsf{d}x}(a_0)\cdot\vec{a}|_{A_k}\right)\end{aligned} \tag{14}$$

We are finally in a position to work out the reverse Faà di Bruno's Formula. WLOG, for convenience, we will assume that in a non-empty partition $[n+1]=A_1|\ldots|A_k$, that $1\in[n+1]$ is always in $1\in A_1$.

Proposition 4.4 *In a Cartesian k-reverse differential category whose reverse differential combinator satisfies the stable rule, the following equality holds:*

$$\frac{\mathsf{r}^{(n+1)}g(f(x))}{\mathsf{r}x}(a_0)\cdot b\cdot a_2\cdot\ldots\cdot a_{n+1}=$$
$$\sum_{\substack{[n+1]=A_1|\ldots|A_k\\ 1\in A_1}}\frac{\mathsf{r}^{(|A_1|)}f(x)}{\mathsf{r}x}(a_0)\cdot\left(\frac{\mathsf{r}^{(k)}g(y)}{\mathsf{r}y}(a_0)\cdot b\cdot\left(\frac{\mathsf{d}^{(|A_2|)}f(x)}{\mathsf{d}x}(a_0)\cdot\vec{a}|_{A_2}\right)\cdot\ldots\cdot\left(\frac{\mathsf{d}^{(|A_k|)}f(x)}{\mathsf{d}x}(a_0)\cdot\vec{a}|_{A_k}\right)\right)\cdot\vec{a}|_{A_1-\{1\}} \tag{15}$$

PROOF: We compute:

$$\frac{\mathsf{r}^{(n+1)}g(f(x))}{\mathsf{r}x}(a_0)\cdot b\cdot a_2\cdot\ldots\cdot a_{n+1}\overset{=}{(10)}\frac{\mathsf{r}\dfrac{\mathsf{d}^{(n+1)}g(f(x))}{\mathsf{d}x}(a_0)\cdot y\cdot a_2\cdot\ldots\cdot a_{n+1}}{\mathsf{r}y}(0)\cdot b$$
$$\overset{=}{(14)}\frac{\mathsf{r}\displaystyle\sum_{[n+1]=A_1|\ldots|A_k}\frac{\mathsf{d}^{(k)}g(z)}{\mathsf{d}z}(f(a_0))\cdot\left(\frac{\mathsf{d}^{(|A_1|)}f(x)}{\mathsf{d}x}(a_0)\cdot y\cdot\vec{a}|_{A_1-\{1\}}\right)\cdot\ldots\cdot\left(\frac{\mathsf{d}^{(|A_k|)}f(x)}{\mathsf{d}x}(a_0)\cdot\vec{a}|_{A_k}\right)}{\mathsf{r}y}(0)\cdot b$$
$$\overset{=}{\textbf{[R.2]}}\sum_{[n+1]=A_1|\ldots|A_k}\frac{\mathsf{r}\dfrac{\mathsf{d}^{(k)}g(z)}{\mathsf{d}z}(f(a_0))\cdot\left(\dfrac{\mathsf{d}^{(|A_1|)}f(x)}{\mathsf{d}x}(a_0)\cdot y\cdot\vec{a}|_{A_1-\{1\}}\right)\cdot\ldots\cdot\left(\dfrac{\mathsf{d}^{(|A_k|)}f(x)}{\mathsf{d}x}(a_0)\cdot\vec{a}|_{A_k}\right)}{\mathsf{r}y}(0)\cdot b$$
$$\overset{=}{(5)}\sum_{\substack{[n+1]=A_1|\ldots|A_k\\ 1\in A_1}}\frac{\mathsf{r}\dfrac{\mathsf{d}^{(|A_1|)}f(x)}{\mathsf{d}x}(a_0)\cdot y\cdot\vec{a}|_{A_1-\{1\}}}{\mathsf{r}y}(0)\cdot\left(\frac{\mathsf{r}\dfrac{\mathsf{d}^{(k)}g(z)}{\mathsf{d}z}(f(a_0))\cdot z\cdot\ldots\cdot\left(\dfrac{\mathsf{d}^{(|A_k|)}f(x)}{\mathsf{d}x}(a_0)\cdot\vec{a}|_{A_k}\right)}{\mathsf{r}z}(0)\cdot b\right)$$
$$\overset{=}{(10)}\sum_{\substack{[n+1]=A_1|\ldots|A_k\\ 1\in A_1}}\frac{\mathsf{r}^{(|A_1|)}f(x)}{\mathsf{r}x}(a_0)\cdot\left(\frac{\mathsf{r}^{(k)}g(y)}{\mathsf{r}y}(a_0)\cdot b\cdot\left(\frac{\mathsf{d}^{(|A_2|)}f(x)}{\mathsf{d}x}(a_0)\cdot\vec{a}|_{A_2}\right)\cdot\ldots\cdot\left(\frac{\mathsf{d}^{(|A_k|)}f(x)}{\mathsf{d}x}(a_0)\cdot\vec{a}|_{A_k}\right)\right)\cdot\vec{a}|_{A_1-\{1\}}$$

So the reverse Faà di Bruno's Formula holds as desired. □

When $n=0$ in the reverse Faà di Bruno's Formula, we get back precisely the reverse chain rule. Indeed, the only non-empty partition of $[1]$ is $[1]=A_1=\{1\}$, which is why no forward derivatives appear in the reverse chain rule. When $n=1$, the non-empty partitions of $[2]$ are $[1]=A_1=\{1\}|A_2=\{2\}$ and $[2]=A_1=\{1,2\}$. So the reverse Faà di Bruno's Formula for the second reverse derivative is:

$$\frac{\mathsf{r}^{(2)}g(f(x))}{\mathsf{r}x}(a_0)\cdot b\cdot a_2$$
$$=\frac{\mathsf{r}^{(1)}f(x)}{\mathsf{r}x}(a_0)\cdot\left(\frac{\mathsf{r}^{(2)}g(y)}{\mathsf{r}y}(f(a_0))\cdot b\cdot\left(\frac{\mathsf{d}f(x)}{\mathsf{d}x}(a_0)\cdot a_2\right)\right)+\frac{\mathsf{r}^{(2)}f(x)}{\mathsf{r}x}(a_0)\cdot\left(\frac{\mathsf{r}^{(1)}g(y)}{\mathsf{r}y}(f(a_0))\cdot b\right)\cdot a_2$$

We leave it as an exercise for the reader to write down the reverse Faà di Bruno's Formula for the third reverse derivative (which involves five summands), and, for those motivated enough, to also write down the formula for the fourth reverse derivative (which involves fifteen summands).

References

[1] 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.

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

[3] J. R. B. Cockett & J.-S. P. Lemay (2022): *Linearizing Combinators*. *Theory and Applications of Categories* 38(13), pp. 374–431.

[4] J. R. B. Cockett & R. A. G. Seely (2011): *The Faà di Bruno construction*. *Theory and Applications of Categories* 25(15), pp. 394–425.

[5] G. Cruttwell, J. Gallagher, J.-S. P. Lemay & D. Pronk (2022): *Monoidal reverse differential categories*. *Mathematical Structures in Computer Science* 32(10), pp. 1313–1363.

[6] G. Cruttwell, J. Gallagher & B. MacAdam (2019): *Towards formalizing and extending differential programming using tangent categories*. *Proc. ACT* 2019.

[7] G. Cruttwell, J. Gallagher & D. Pronk (2021): *Categorical Semantics of a Simple Differential Programming Language*. *Electronic Proceedings in Theoretical Computer Science* 333, pp. 289–310, doi:10.4204/EPTCS.333.20.

[8] G. Cruttwell & J.-S. P. Lemay (2024): *Reverse Tangent Categories*. In Aniello Murano & Alexandra Silva, editors: *32nd EACSL Annual Conference on Computer Science Logic (CSL 2024)*, *Leibniz International Proceedings in Informatics (LIPIcs)* 288, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, pp. 21:1–21:21, doi:10.4230/LIPIcs.CSL.2024.21. Available at `https://drops-dev.dagstuhl.de/entities/document/10.4230/LIPIcs.CSL.2024.21`.

[9] G.S.H. Cruttwell, B. Gavranovic, N. Ghani, P. Wilson & F. Zanasi (2022): *Categorical Foundations of Gradient-Based Learning*. *ESOP 2022*, pp. 1–28, doi:10.1007/978-3-030-99336-8_1.

[10] B. Fong, D. Spivak & R. Tuyéras (2019): *Backprop as functor: A compositional perspective on supervised learning*. In: *2019 34th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, IEEE, pp. 1–13.

[11] R. Garner & J.-S. P. Lemay (2021): *Cartesian differential categories as skew enriched categories*. *Applied Categorical Structures* 29(6), pp. 1099–1150.

[12] B. Gavranović, P. Lessard, A. Dudzik, T. von Glehn, J. Araújo & P. Veličković (2024): *Categorical Deep Learning: An Algebraic Theory of Architectures*. *arXiv preprint arXiv:2402.15332*.

[13] M. Vákár & T. Smeding (2022): *CHAD: Combinatory homomorphic automatic differentiation*. *ACM Transactions on Programming Languages and Systems (TOPLAS)* 44(3), pp. 1–49.

[14] P. Wilson & F. Zanasi (2021): *Reverse Derivative Ascent: A Categorical Approach to Learning Boolean Circuits*. In: Proceedings of the 3rd Annual International *Applied Category Theory Conference 2020*, Cambridge, USA, 6-10th July 2020, 333, Open Publishing Association, pp. 247–260, doi:10.4204/EPTCS.333.17.

[15] P. Wilson & F. Zanasi (2022): *Categories of differentiable polynomial circuits for machine learning*. In: *International Conference on Graph Transformation*, Springer, pp. 77–93.

A Stable Rule for Smooth Functions

For a smooth function $F = \langle f_1, \ldots, f_m \rangle : \mathbb{R}^n \to \mathbb{R}^m$, recall that its reverse derivative $\mathsf{R}[F] : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is:

$$\mathsf{R}[F](\vec{x},\vec{y}) = (\mathsf{R}_1[F](\vec{x},\vec{y}), \ldots, \mathsf{R}_n[F](\vec{x},\vec{y})) = \left(\sum_{j=1}^{m} \frac{\partial f_j}{\partial x_1}(\vec{x}) y_j, \ldots, \sum_{j=1}^{m} \frac{\partial f_j}{\partial x_n}(\vec{x}) y_j \right)$$

while its forward derivative $\mathsf{D}[F] : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^m$ is:

$$\mathsf{D}[F](\vec{x},\vec{z}) = (\mathsf{D}[f_1](\vec{x},\vec{y}), \ldots, \mathsf{D}[f_n](\vec{x},\vec{y})) = \left(\sum_{i=1}^{n} \frac{\partial f_1}{\partial x_i}(\vec{x}) z_i, \ldots, \sum_{i=1}^{n} \frac{\partial f_m}{\partial x_i}(\vec{x}) z_i \right)$$

On the other hand, For a smooth function $G = \langle g_1, \ldots, g_k \rangle : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^k$, its partial reverse derivative in its first argument $\mathbb{R}^n$ is $\mathsf{R}_1[G] : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^k \to \mathbb{R}^n$ defined as follows:

$$\mathsf{R}_1[G](\vec{x},\vec{y},\vec{z}) = \left(\sum_{j=1}^{k} \frac{\partial g_j}{\partial x_1}(\vec{x},\vec{y}) z_j, \ldots, \sum_{j=1}^{k} \frac{\partial f_j}{\partial x_n}(\vec{x},\vec{y}) z_j \right)$$

We need to show that $\mathsf{R}_1[\mathsf{D}[F]] : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ and $\mathsf{R}_1[\mathsf{R}[F]] : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}^n$ are equal up to permutation of the second and third arguments. We first compute $\mathsf{R}_1[\mathsf{D}[F]]$ to be:

$$\begin{aligned} \mathsf{R}_1[\mathsf{D}[F]](\vec{x},\vec{z},\vec{y}) &= \left(\sum_{j=1}^{m} \frac{\partial \mathsf{D}[f_j]}{\partial x_1}(\vec{x},\vec{z}) y_j, \ldots, \sum_{j=1}^{m} \frac{\partial \mathsf{D}[f_j]}{\partial x_n}(\vec{x},\vec{z}) y_j \right) \\ &= \left(\sum_{j=1}^{m} \sum_{i=1}^{n} \frac{\partial^2 f_j}{\partial x_i \partial x_1}(\vec{x}) z_i y_j, \ldots, \sum_{j=1}^{m} \sum_{i=1}^{n} \frac{\partial^2 f_j}{\partial x_i \partial x_n}(\vec{x}) z_i y_j \right) \end{aligned}$$

On the other we compute $\mathsf{R}_1[\mathsf{R}[F]]$ to be:

$$\begin{aligned} \mathsf{R}_1[\mathsf{R}[F]](\vec{x},\vec{y},\vec{z}) &= \left(\sum_{i=1}^{n} \frac{\partial \mathsf{R}_i[F]}{\partial x_1}(\vec{x}) z_i, \ldots, \sum_{i=1}^{n} \frac{\partial \mathsf{R}_i[F]}{\partial x_n}(\vec{x}) z_i \right) \\ &= \left(\sum_{i=1}^{n} \sum_{j=1}^{m} \frac{\partial^2 f_j}{\partial x_i \partial x_1}(\vec{x}) y_j z_i, \ldots, \sum_{i=1}^{n} \sum_{j=1}^{m} \frac{\partial^2 f_j}{\partial x_i \partial x_n}(\vec{x}) y_j z_i \right) \end{aligned}$$

So we have that $\mathsf{R}_1[\mathsf{D}[F]](\vec{x},\vec{z},\vec{y}) = \mathsf{R}_1[\mathsf{R}[F]](\vec{x},\vec{y},\vec{z})$ as desired. So smooth functions satisfy the stable rule. The same proof works for polynomials.