Differential Categories II

R. A. G. Seely

(joint work with Rick Blute & Robin Cockett)

<www.math.mcgill.ca/rags/>
Introduction

- Motivating example of linear logic
  \[ A \Rightarrow B = !A \Rightarrow B \]
  - coKleisli category of cotriple

- Differential λ-calculus of Ehrhard & Regnier

\[ \text{Köthe spaces} \]
\[ \text{Finiteness spaces} \]

Our aims:
- Categorically "reconstruct" the E-R differential structure
- Basic setting: monoidal category with comonad on it

Intuition: The "base category" maps are "linear"
  - CoKleisli maps are "smooth"

An illustration of how this works

A smooth map \( f : \mathbb{R}^3 \rightarrow \mathbb{R}^2 \)
\[ f(x, y, z) = (x^2 + yz, z^2 - xy) \]

Its Jacobian
\[ \begin{pmatrix} 2x + yz & xz & xy \\ -y & -z & 2z^2 \end{pmatrix} \]

For chosen \( <x, y, z> \) this is a linear map \( \mathbb{R}^3 \rightarrow \mathbb{R}^2 \)

ie from \( f : A \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^m \)
we get \( D[f] : A \rightarrow \text{Lin}(\mathbb{R}^n, \mathbb{R}^m) \)

So in our setting we would have this:
\[ f : !A \rightarrow B \]
\[ D[f] : !A \rightarrow (A \Rightarrow B) \]

Linear Hom

To avoid the need for closed structure, we shall take
\[ D[f] : A \otimes !A \rightarrow B \]
Out line of talk

- Basic definition of differential category
  - coalgebra modality on a (semi)additive symmetric monoidal category
  - differential combinator

- Question: how can we characterize the coKleisli category of such a diff cat?
  - we shall drop the tensor structure
  - define two secondary notions:
    - Cartesian differential categories
    - pre differential categories
      - @-free diff cats
    - examine the connection between these
      & between pre diff cats & diff cats

[Technical remark: all this is "cleanest" in the "strong" case - our focus today]
Basic context

- (semi) additive symmetric monoidal category * 
- Commutative monoid enriched [no assumption of biproducts-yet!]
- Ex: Sets & relations in (semi)additive but not AbGrp enriched

Coalgebra modality!
- A cat-triple (comonoid) 
- \( T \xleftarrow{e} !X \xrightarrow{\Delta} !X \otimes !X \) is a comonoid 
- \((!X, \Delta, e)\) is a comonoid 
- \( \delta: !X \rightarrow ! !X \) is a comonoid morphism

Intuition: \( !A \rightarrow B \) is "a differentiable map \( A \rightarrow B \)"
(but we need more structure to realize this)

Storage

Given a s.m.cat with products and a comonoid! 
a comonoidal transformation \( s_0: ! \rightarrow ! \) 
from \((X, x, 1) \Rightarrow (X, \otimes, T)\) amounts to 
\( s_0: !!(1) \rightarrow T \) and \( s_2: !!(x \otimes y) \rightarrow !x \otimes !y \)

\[
\begin{align*}
S_0 & : !((1)) \rightarrow T \quad \text{and} \quad S_2 : !((x \otimes y)) \rightarrow !x \otimes !y \\
S_5 & : !((x \otimes y) \otimes z) \rightarrow !((x \otimes y) \otimes z) \\
& \downarrow \quad \quad \downarrow \quad \quad \downarrow \quad \quad \downarrow \quad \quad \downarrow \\
& !((x \otimes y) \otimes z) \rightarrow !x \otimes !y \otimes z \rightarrow !x \otimes ((y \otimes z)) \\
\end{align*}
\]

\[S_5 \Rightarrow !((x \otimes y) \otimes z) \rightarrow !((x \otimes y) \otimes z) \rightarrow !x \otimes !y \otimes z \rightarrow !x \otimes ((y \otimes z)) \]

\[
\begin{align*}
S_2 & : !((x \otimes y)) \rightarrow !x \otimes !y \\
S_0 & : !((1)) \rightarrow T \\
S_0 & : !((x \otimes y)) \rightarrow !x \otimes !y \\
\end{align*}
\]

(+ diagram for symmetry if appropriate)

(as a comonad)

In our setting, requiring that \( ! \) be comonoidal is too strong.
We'd want \( \delta \) to be so, but not \( e \). (The Id functor is not comonoidal)

\[
\begin{align*}
F(X \otimes Y) & \xrightarrow{\alpha} G(X \otimes Y) \\
\downarrow \gamma & \Rightarrow \downarrow \gamma \\
FX \otimes FY & \xrightarrow{T} GX \otimes GY \\
\end{align*}
\]

No such \( \delta \) for \( G \times Id \)
A (co)monoidal cat $X$ with canonical !, products has a storage transformation if there is a comonoidal transformation $s : ! \rightarrow ! : (X, x, 1) \rightarrow (X, \otimes, T)$ so that $s$ is comonoidal.

Key Fact:

For a (symm) monoidal cat with products:

to have a comonoid with (symm) storage trans is equiv.
to having a (cocommutative) coalgebra modality.

(1) Define $s_\lambda : !(X \times Y) \rightarrow !(X \times Y) \otimes !(X \times Y) \rightarrow !X \otimes !Y$

This works!

Examples

- $id$ on any cat with finite products
- $!$ in linear logic
- Dual of "algebra modality"
  - The free algebra $T(x) = \bigoplus_{n \geq 0} X^\otimes^n$
  - The free symmetric algebra $\text{Sym} (X) = \bigoplus_{n \geq 0} X^\otimes^n S_n$
  - The "exterior algebra" $\Lambda (x) = \bigoplus_{n \geq 0} X^\otimes^n / A$

Differential Combinators

$D_{\delta^y} : X(!A, B) \rightarrow X(A \otimes !A, B)$

$D_{\delta^y} = !A \rightarrow (A \rightarrow B)$

This must satisfy:

- naturality (for combinatorics), additivity
- "constants have deriv = 0" $D[e] = 0$
- product rule $(1 \otimes (DF)(EG)) + (1 \otimes (e_\delta)(1 \otimes (EG))) = D[\delta^y(1 \otimes (FG))]$
- "linear maps have constant deriv" $D[ef] = (1 \otimes f)$
- chain rule $D[\delta^y(1 \otimes (DF)(\otimes 1 \otimes f))DF] = (1 \otimes (DF)(\otimes 1 \otimes f))D[\delta^y]$

There is a "circuit calculus" for all this ...
Preliminaries

\( \text{cartesian differential categories} \)

\( \text{(semi)} \)

Left additive category: each hom set is a commutative monoid

\[
\begin{align*}
f(g+h) &= fg + fh \\ f0 &= 0
\end{align*}
\]

\( \text{(semi)} \)

A map \( f \) is left additive if also

\[
(fg)h = fh + gh \\
0h = 0
\]

Prop. The additive maps of a left additive category \( \mathbb{Y} \) form an additive subcategory \( \mathbb{Y}^+ \).

The inclusion \( \mathbb{Y}^+ \rightarrow \mathbb{Y} \) reflects isos.

Eg. Commutative monoids with "set" maps form a left additive, not additive, category.

Left additive: because operations are defined pointwise

Not additive: because maps need not preserve monoid str. Those that do are in \( \mathbb{Y}^+ \).

Cartesian Differential categories

- Left additive
- Products
- A cartesian differential operator

\[
\begin{array}{c}
X \xrightarrow{f} Y \\
\xmapsto{D_x[f]} Y
\end{array}
\]

Think: 1st arg is "linear"; 2nd is "smooth".

Satisfies several axioms:

\[
\begin{align*}
[CD1] & \quad D_x[f + g] = D_x[f] + D_x[g] \quad ; \quad D_x[0] = 0 \\
[CD2] & \quad \langle h + k, \tau \rangle D_x[f] = \langle h, \tau \rangle D_x[f] + \langle k, \tau \rangle D_x[f] \\
& \quad \langle 0, \tau \rangle D_x[f] = 0 \\
[CD3] & \quad D_x[1] = 1 \quad ; \quad D_x[1] = 1 \quad (i = 0, 1) \\
[CD4] & \quad D_x[\langle f, g \rangle] = \langle D_x[f], D_x[g] \rangle \\
[CD5] & \quad D_x[f g] = \langle D_x[f], \pi, f \rangle D_x[g]
\end{align*}
\]

Example: Fin dim vector spaces over \( \mathbb{R} \) (or \( \mathbb{C} \)) with \( \infty \) differentiable maps.

- \( D_x \) given by Jacobian.

\( \vdots \)
"Linear maps"

In a cart diff cat, \( f \) is linear if
\[ D_x[f] = \pi_0 f \]

Prop: The linear maps form an additive subcat \( Y_{\text{lin}} \) of a cart diff cat \( Y \); \( Y_{\text{lin}} \) has (bi)products;
\( Y_{\text{lin}} \rightarrow Y \) reflects isos & creates products.

More: The coKleisli cat of a diff cat also satisfies
- \( D_x[e] = \pi_0 e \)
- \( D_x[S(f)] = \pi_0 S(f) \) (for any \( f \))

Strong Differential categories
- Diff cats with storage transformations \( S_0, S_2 \) (without further coherence - this is weaker than what we called "storage diff. cats")

In this context we can define some maps in the coKleisli cat:
\[ \eta: A \rightarrow S(A) \quad (\text{ie } \eta: \text{id } S(A) \rightarrow S(A) \text{ in } X) \]
\[ \eta: A \rightarrow S(A) := e(1 \otimes \iota) d_0 \]
\[ \xi: S(A) \rightarrow A \otimes S(A) \xrightarrow{\eta} A \rightarrow S(A) \text{ in } X \]
\[ d_x: A \times A \rightarrow S(A) := D_x[\eta] \]

\[ \xi: S(AXA) \rightarrow S(A) \otimes S(A) \xrightarrow{\eta_1} A \rightarrow S(A) \rightarrow S(A) \text{ in } X \]

Notation: \( D_x \) is the diff exponent for the diff cat
(\( D_x \) \( \Gamma \) is the cart diff \( \Gamma \))
\( S \) is the counoved
(called \( ! \) in linear logic)
Proof: The coKleisli cat of a strong diff cat satisfies:
  * $\eta$ is "$\epsilon$-natural"
    \[ \begin{array}{c}
    \eta = (1,0) \quad \text{dx} \\
    \eta = 1 \\
    \text{"$\epsilon$-natural" = "linear"}
    \end{array} \]
  * $S(dx) \epsilon = s_0 (\epsilon \otimes 1) \nu$
  * $d_0 = (\eta \otimes 1) \circ ^{-1} S(dx) \epsilon$ \\

So we can recapture $d_0$ from $dx$ as well as vice versa.

These are properties we shall want to hold in our characterization of coKleisli cats of (strong) diff cats.

Abstract coKleisli category

- Cat $Y$, functor $S: Y \to Y$
- Nat. transf: $\varphi: A \to S(\varphi) A$
- Unnat. transf: $\epsilon : S(\varphi) A \to A$
- $\epsilon \epsilon = S(\epsilon) \epsilon$ \textcolor{red}{"$\epsilon$ is "$\epsilon$-natural"}
- $\varphi \epsilon = 1_A \quad S(\varphi) \epsilon = 1_A^\epsilon$
- $\epsilon_{1A} : S(S(\varphi)) \to S(\varphi)$ is natural in $A$

This makes $(S, \varphi, \epsilon)$ a monad/triple on $Y$

Let $Ye$ be the subcat of $\epsilon$-natural maps.
Then $(S, \epsilon, S(\varphi))$ is a comonad on $Ye$ and $Y$ is its coKleisli cat $(Ye)^*$

Q: Given a comonad $S$, what is the connection between $X$ and $(X_S)\epsilon$ ?
(for any $X$ with a comonad $S$)

Disclaimer: The "nat strong" case is more "fiddley"!
Technical remark

If \( Y \) is an abstract co-Kleisli cat.

1. \[ A \xrightarrow{f} S(A) \xrightarrow{g} S(S(A)) \] is a (split) equalizer (forall \( A \))

2. \[ S(S(A)) \xrightarrow{g} S(A) \xrightarrow{f} A \] is a (split) coequalizer (forall \( A \))

The 2nd diagram is in \( Y \); it characterizes commutative abstract co-Kleisli cats, in the sense:

**FACT:** The canonical functor (when \( X \) carries a co-unit):

\[ X \xrightarrow{(x)} X_s \] is an iso if \( 2 \) is a coequalizer (forall \( A \))

(called such \( X \) an exact co-unit)

In the "strong" context, having of forces exactness

Since \( eS = S(e) \), \( gS = 1 \), \( g(e) = e \)

(Without "strength", \( X \xrightarrow{(x)} X_s \) need not even preserve what structure \( X \) has, so things become "fiddly", as we've said before)

\( Y \) is a strong abstract differential category if in addition:

- left additive
- \( e, S(f) \) (forall \( f \)) are additive

[This guarantees \( Y_e \) is in fact additive]

Furthermore, the co-Kleisli cat of a co-unit on an additive cat satisfies these, so this characterizes co-Kleisli cats for commutative additive cats]

\( Y \) is an abstract additive co-Kleisli category if in addition:

- cartesian diff. cat at \( T_\varepsilon, A \) are \( \varepsilon \)-natural,
  - \( D_x[e] = T_\varepsilon e \)
  - \( D_x[S(f)] = T_\varepsilon S(f) \) (forall \( f \))
  - \( \eta = \langle 1, 0 \rangle D_x[p] \) is \( \varepsilon \)-natural

[These guarantee all the properties of the earlier prop., which hold of co-Kleisli cats of strong diff. cats - incl. "\( \varepsilon \)-natural" = "linear"]
Technical note (→"pre-differential categories")

In this context (somewhat less suffices) we can define a "cartesian deriving transformation"

\[ d_x : A \times A \rightarrow S(A) \]

satisfying "the usual axioms", so that this structure is equivalent to having a cartesian differential operator \( D_x \).

An impressionist's view of the axioms:

[cd1] \( d_x S(fg) \epsilon = d_x S(f) \epsilon + d_x S(g) \epsilon \); \( d_x S(\epsilon) = 0 \)

[cd2] \( \langle h+k, v \rangle d_x = \langle h, v \rangle d_x + \langle k, v \rangle d_x \); \( \langle 0, v \rangle d_x = 0 \)

[cd3] \( d_x \epsilon = \Pi \epsilon \)

[cd4] \( d_x S(\langle f, g \rangle) \epsilon = d_x \langle S(f) \epsilon, S(g) \epsilon \rangle \)

[cd5] \( \langle d_x S(f) \epsilon, \Pi; f \rangle \rangle d_x S(g) \epsilon = d_x S(fg) \epsilon \)

\[ \eta : A \rightarrow S(A) \]

\[ \eta = 1 \]

\[ \delta \]

\[ \delta S(\langle h+k, v \rangle) d_x' = \delta [S(h, v) + S(k, v)] d_x' \]

\[ \delta d_x' \epsilon = \epsilon \delta \epsilon \]

\[ \delta S(f) g = \delta \langle d_x f, S(\Pi; f) \rangle d_x g \]

\[ \eta \epsilon = 1 \]

\[ \delta (\langle 1, 0 \rangle) d_x' = \epsilon \eta \]

These are "cokleisli" translations of the axioms for \( d_x, \eta \).
The point of this:

- The coKleisli cat $X_5$ of a strong prediff. cat $X$ is a strong abstract diff cat at $(X_5)_c = X$.

- If $Y$ is a strong abstract diff cat, then $Y_5$ is a $\text{strong}$ prediff cat $[\text{and of course } (Y_5)_c = Y]$.

So we've characterized abstract diff cats as coKleisli cats on prediff cats.

- What about diff cats?
- What happened to $\otimes$?

Prop: $X$ is a strong differential cat iff

- $X$ carries a strong coalgebra modality.
- $X$ is a strong predifferential category at $d'_x = s_2 \circ (6 \otimes 1) \circ \triangledown$.

Recall this property from previous slide.

$\text{Diagram: chase!}$
Look again at our "atlas":

\[
\begin{align*}
X & \sim X \\
\text{Strong} & \leftrightarrow \text{Weak} & \text{PreDiff} & \rightarrow \text{DiffCat}
\end{align*}
\]

\[
Y \rightarrow Y_e
\]

Comment about "the fiddley bits":
- Stepping "down" (dropping "strong") just requires care (some exactness assumptions at times)
- Stepping "up" ("storage" - in the linear case!)
only affects the "abstract" column: how to describe storage "abstractly" can be done by examining the interaction of @ and x in coltensic cats
[details are in the paper!]

Future work?
- Examples would be nice...
- Higher order version of this setting also...
- Connect to other notions of differentiability...
- Some of this is in progress, some still just "in preparation"