Multi-(Co)Iteration, Categorically

Tarmo Uustalu

Inst. of Cybernetics at Tallinn Technical Univ.
Akadeemia tee 21, EE-12618 Tallinn, Estonia
email tarmo@cs.ioc.ee

Abstract Generic (total) functional programming combinators for multi-iteration (an iteration-like scheme that inducts over several data structures simultaneously) and multi-coiteration (a coiteration-like scheme that, simultaneously to coinducting over a codata structure, inducts over one or more data structures) are described.

1 Introduction

The research in generic functional programming is, in its present state, largely a quest for useful generic combinators with nice calculational properties. This note is a study of generic (total) functional programming combinators for multi-iteration and multi-coiteration from the point of view of their calculational properties. Multi-iteration and multi-coiteration are relatives of (plain) iteration and coiteration, which are the basic function definition schemes associated to inductive and coinductive types. Multi-iteration, treated in [FSZ94] in a syntactically oriented fashion and in [HIT97] in a category-theoretic framework, is a natural way for defining a function with several arguments, all of inductive types, which, when evaluated, consumes all argument data at the same pace. Typical bi-iterative functions are functions of essentially comparative character. Multi-coiteration, although it does not seem to have received much attention, is an equally natural way for defining a function to a coinductive type, with all but one of its arguments inductive, which, when evaluated, produces the value codata at the same pace as it consumes the argument data. Typical bi-coiterative functions are functions of copying character. It has to be noted that multi-coiteration is not dual to multi-iteration; multi-iteration relates to multi-coiteration in much the same fashion as iteration with (static) parameters, studied carefully in [Par97], relates to coiteration with (static) parameters. Multi-iteration and and multi-coiteration might, as a matter of fact, be thought of as iteration and coiteration with dynamically destructed parameters, but such way of thinking obscures the symmetry aspect of the multi schemes.

In this note, we give definitions by characterization and list some laws for three generic combinators for multi-iteration and multi-coiteration, the uncurried ($^{\mathrm{uc}}$), left-curried ($^{\mathrm{c}}$) and right-curried ($^{\mathrm{c}'}$) multi-catamorphism and multi-anamorphism combinators. One of our intentions is to show that the equal in beauty to the uncurried multi-catamorphism combinator among the multi-anamorphism combinators is the the left-curried one.

Throughout the note, we are working with one base category $\mathcal{C}$ which we require to be cartesian closed, i.e., to have finite products $(- \times -')$ and exponentials (higher types) $(-' \Rightarrow -)$. $\mathtt{cur}$ is the left-currying combinator; $\mathtt{cur}'$ is the right-currying combinator, $\mathtt{cur}' f = \mathtt{cur}(f \circ \mathtt{swap})$, where $\mathtt{swap} = \langle \mathtt{snd}, \mathtt{fst} \rangle$. $\mathtt{uncur}$ is the uncurrying combinator, $\mathtt{uncur}\, f = g$ iff $f = \mathtt{cur}\, g$; $\mathtt{exch}$ is the argument swap combinator for curried functions, $\mathtt{exch}\, f = \mathtt{cur}'(\mathtt{uncur}\, f)$. If F is an endofunctor on $\mathcal{C}$ such that a data type exists with base F, then μF is such data type, $\mathtt{In}_F$ the associated constructor, and $(\!| - |\!)_F$ is the F-catamorphism combinator. If a codata type exists with base F, then νF is such codata type, $\mathtt{Out}_F$ the associated destructor, and $[\!(-)\!]_F$ is the F-anamorphism combinator.

2 Multi-Catamorphisms

Let F, G, H be endofunctors on $\mathcal{C}$, and assume that data types exists both with base F and base G.

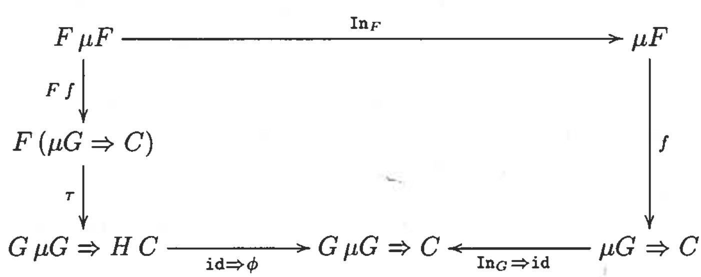
The key to the definition by characterizations of three kinds of $F,G;H$-bi-catamorphisms, which we will present shortly, lies in the following proposition.

Proposition 1. *Given* $\phi : H\,C \to C$.

- *If* $\tau : F(Y \Rightarrow Z) \to G\,Y \Rightarrow H\,Z$ *is natural in* Z, Y *[notation* $NAT^{c}_{F,G;H}(\tau)$*], then a unique* $f : \mu F \to \mu G \Rightarrow C$ *exists such that*

$$(\mathtt{In}_G \Rightarrow \mathtt{id}) \circ f \circ \mathtt{In}_F = (\mathtt{id} \Rightarrow \phi) \circ \tau \circ F\,f$$

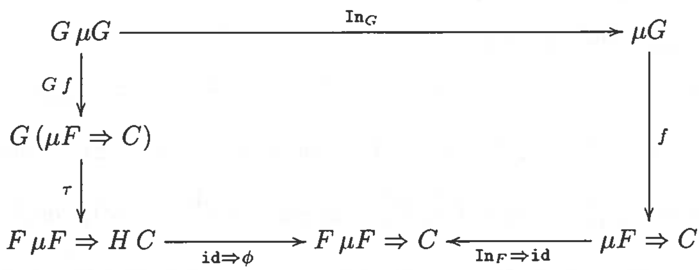
  [notation $CATA^{c}(\tau, \phi, f)$*], i.e.,*



- *If* $\tau : G(X \Rightarrow Z) \to F\,X \Rightarrow H\,Z$ *is natural in* Z, X *[notation* $NAT^{c'}_{F,G;H}(\tau)$*], then a unique* $f : \mu G \to \mu F \Rightarrow C$ *exists such that*

$$(\mathtt{In}_F \Rightarrow \mathtt{id}) \circ f \circ \mathtt{In}_G = (\mathtt{id} \Rightarrow \phi) \circ \tau \circ G\,f$$

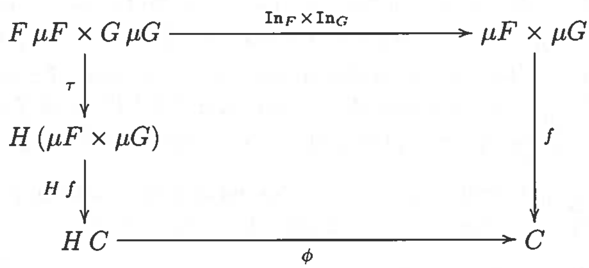
[notation $CATA^{c'}(\tau, \phi, f)$*], i.e.,*



– *If* $\tau : F\,X \times G\,Y \to H(X \times Y)$ *is natural in* X, Y *[notation* $NAT^{uc}_{F,G;H}(\tau)$*], then a unique* $f : \mu F \times \mu G \to C$ *exists such that*

$$f \circ (\mathbf{In}_F \times \mathbf{In}_G) = \phi \circ H\,f \circ \tau$$

[notation $CATA^{uc}(\tau, \phi, f)$*], i.e.,*



Proposition 1 follows straightforwardly from the following lemma which is easy to verify.

Lemma 1. *Given* $\phi : H\,C \to C$*.*

– *If* $NAT^{c}_{F,G;H}(\tau)$*, then* $(\mathbf{In}_G^{-1} \Rightarrow \phi) \circ \tau : F(\mu G \Rightarrow C) \to \mu G \Rightarrow C$ *and*

$$CATA^{c}(\tau, \phi, f) \;\equiv\; f = (\!|\, (\mathbf{In}_G^{-1} \Rightarrow \phi) \circ \tau \,|\!)_F$$

– *If* $NAT^{c'}_{F,G;H}(\tau)$*, then* $(\mathbf{In}_F^{-1} \Rightarrow \phi) \circ \tau : G(\mu F \Rightarrow C) \to \mu F \Rightarrow C$ *and*

$$CATA^{c'}(\tau, \phi, f) \;\equiv\; f = (\!|\, (\mathbf{In}_F^{-1} \Rightarrow \phi) \circ \tau \,|\!)_G$$

– *If* $NAT^{uc}_{F,G;H}(\tau)$*, then* $NAT^{c}_{F,G;H}(\mathbf{cur}(H(\mathbf{uncur}\,\mathbf{id}) \circ \tau))$ *and* $NAT^{c'}_{F,G;H}(\mathbf{cur}'(H(\mathbf{uncur}'\,\mathbf{id}) \circ \tau))$ *and*

$$\begin{aligned} & CATA^{uc}(\tau, \phi, f) \\ \equiv\;& CATA^{c}(\mathbf{cur}(H(\mathbf{uncur}\,\mathbf{id}) \circ \tau), \phi, \mathbf{cur}\,f) \\ \equiv\;& CATA^{c'}(\mathbf{cur}'(H(\mathbf{uncur}'\,\mathbf{id}) \circ \tau), \phi, \mathbf{cur}'\,f) \end{aligned}$$

Also easy to verify (but not needed in order to prove Proposition 1) is this lemma used below.

Lemma 2. *Given* $\phi : H\,C \to C$.

– *If* $NAT^{\mathrm{c}}_{F,G;H}(\tau)$, *then* $NAT^{\mathrm{uc}}_{F,G;H}(\mathtt{uncur}(\tau \circ F(\mathtt{cur\,id})))$ *and*

$$CATA^{\mathrm{c}}(\tau, \phi, f) \;\equiv\; CATA^{\mathrm{uc}}(\mathtt{uncur}(\tau \circ F(\mathtt{cur\,id})), \phi, \mathtt{uncur} f)$$

– *If* $NAT^{\mathrm{c}'}_{F,G;H}(\tau)$, *then* $NAT^{\mathrm{uc}}_{F,G;H}(\mathtt{uncur}'(\tau \circ G(\mathtt{cur}'\,\mathtt{id})))$ *and*

$$CATA^{\mathrm{c}'}(\tau, \phi, f) \;\equiv\; CATA^{\mathrm{uc}}(\mathtt{uncur}'(\tau \circ G(\mathtt{cur}'\,\mathtt{id})), \phi, \mathtt{uncur}' f)$$

By Proposition 1, the following is a well-formed definition of three kinds of $F, G; H$-bi-catamorphisms.

Definition 1. *Given* $\phi : H\,C \to C$.

- *If* $NAT^{\mathrm{c}}_{F,G;H}(\tau)$, *then say the* f *such that* $CATA^{\mathrm{c}}(\tau, \phi, f)$ *to be the* left-curried $F, G; H$-bi-catamorphism via τ of ϕ *[notation* $(\!|\, \tau, \phi \,|\!)^{\mathrm{c}}$*].*
- *If* $NAT^{\mathrm{c}'}_{F,G;H}(\tau)$, *then say the* f *such that* $CATA^{\mathrm{c}'}(\tau, \phi, f)$ *to be the* right-curried $F, G; H$-bi-catamorphism via τ of ϕ *[notation* $(\!|\, \tau, \phi \,|\!)^{\mathrm{c}'}$*].*
- *If* $NAT^{\mathrm{uc}}_{F,G;H}(\tau)$, *then say the* f *such that* $CATA^{\mathrm{uc}}(\tau, \phi, f)$ *to be the* uncurried $F, G; H$-bi-catamorphism via τ of ϕ *[notation* $(\!|\, \tau, \phi \,|\!)^{\mathrm{uc}}$*].*

The following calculational laws of *typing and cancellation* for bi-catamorphisms are trivial consequences of the defining characterizations.

Corollary 1. *Given* $\phi : H\,C \to C$.

– *If* $NAT^{\mathrm{c}}_{F,G;H}(\tau)$, *then* $(\!|\, \tau, \phi \,|\!)^{\mathrm{c}} : \mu F \to \mu G \Rightarrow C$ *and*

$$(\mathtt{In}_G \Rightarrow \mathtt{id}) \circ (\!|\, \tau, \phi \,|\!)^{\mathrm{c}} \circ \mathtt{In}_F = (\mathtt{id} \Rightarrow \phi) \circ \tau \circ F\,(\!|\, \tau, \phi \,|\!)^{\mathrm{c}}$$

– *If* $NAT^{\mathrm{c}'}_{F,G;H}(\tau)$, *then* $(\!|\, \tau, \phi \,|\!)^{\mathrm{c}'} : \mu G \to \mu F \Rightarrow C$ *and*

$$(\mathtt{In}_F \Rightarrow \mathtt{id}) \circ (\!|\, \tau, \phi \,|\!)^{\mathrm{c}'} \circ \mathtt{In}_G = (\mathtt{id} \Rightarrow \phi) \circ \tau \circ G\,(\!|\, \tau, \phi \,|\!)^{\mathrm{c}'}$$

– *If* $NAT^{\mathrm{uc}}_{F,G;H}(\tau)$, *then* $(\!|\, \tau, \phi \,|\!)^{\mathrm{uc}} : \mu F \times \mu G \to C$ *and*

$$(\!|\, \tau, \phi \,|\!)^{\mathrm{uc}} \circ (\mathtt{In}_F \times \mathtt{In}_G) = \phi \circ H\,(\!|\, \tau, \phi \,|\!)^{\mathrm{uc}} \circ \tau$$

More interesting consequences of the characterizations are the following bi-cata *fusion* laws.

Corollary 2. *Given* $\phi : H\,C \to C$, $\psi : H\,D \to D$.

– *If* $NAT^{\mathrm{c}}_{F,G;H}(\tau)$, $h : D \to C$, *then*

$$h \circ \psi = \phi \circ H\,h \;\supset\; (\mathtt{id} \Rightarrow h) \circ (\!|\, \tau, \psi \,|\!)^{\mathrm{c}} = (\!|\, \tau, \phi \,|\!)^{\mathrm{c}}$$

– *If* $NAT^{c'}_{F,G;H}(\tau)$, $h : D \to C$, *then*

$$h \circ \psi = \phi \circ H\, h \quad \supset \quad (\mathtt{id} \Rightarrow h) \circ (\!|\, \tau, \psi \,|\!)^{c'} = (\!|\, \tau, \phi \,|\!)^{c'}$$

– *If* $NAT^{uc}_{F,G;H}(\tau)$, $h : D \to C$, *then*

$$h \circ \psi = \phi \circ H\, h \quad \supset \quad h \circ (\!|\, \tau, \psi \,|\!)^{uc} = (\!|\, \tau, \phi \,|\!)^{uc}$$

Equally interesting are the cata - cata - bi-cata *fusion* laws.

Corollary 3. *Given* $\phi : H\, C \to C$, $\xi : F\, E \to E$, $\xi' : G\, E' \to E'$.

– *If* $NAT^{c}_{F,G;H}(\tau)$, $h : E \to E' \Rightarrow D$, *then*

$$(\xi' \Rightarrow \mathtt{id}) \circ h \circ \xi = (\mathtt{id} \Rightarrow \phi) \circ \tau \circ F\, h \quad \supset \quad ((\!|\, \xi' \,|\!)_G \Rightarrow \mathtt{id}) \circ h \circ (\!|\, \xi \,|\!)_F = (\!|\, \tau, \phi \,|\!)^{c}$$

– *If* $NAT^{c'}_{F,G;H}(\tau)$, $h : E' \to E \Rightarrow D$, *then*

$$(\xi \Rightarrow \mathtt{id}) \circ h \circ \xi' = (\mathtt{id} \Rightarrow \phi) \circ \tau \circ G\, h \quad \supset \quad ((\!|\, \xi \,|\!)_F \Rightarrow \mathtt{id}) \circ h \circ (\!|\, \xi' \,|\!)_G = (\!|\, \tau, \phi \,|\!)^{c'}$$

– *If* $NAT^{uc}_{F,G;H}(\tau)$, $h : E \times E' \to D$, *then*

$$h \circ (\xi \times \xi') = \phi \circ H\, h \circ \tau \quad \supset \quad h \circ ((\!|\, \xi \,|\!)_F \times (\!|\, \xi' \,|\!)_G) = (\!|\, \tau, \phi \,|\!)^{uc}$$

From Lemmas 1, 2, we may learn the laws stated in the following corollary.

Corollary 4. *Given* $\phi : H\, C \to C$.

– *If* $NAT^{c}_{F,G;H}(\tau)$, *then*

$$(\!|\, \tau, \phi \,|\!)^{c} = (\!|\, (\mathtt{In}_G^{-1} \Rightarrow \phi) \circ \tau \,|\!)_F = \mathtt{cur}(\!|\, \mathtt{uncur}(\tau \circ F(\mathtt{cur\, id})) \,|\!)^{uc}$$

– *If* $NAT^{c'}_{F,G;H}(\tau)$, *then*

$$(\!|\, \tau, \phi \,|\!)^{c'} = (\!|\, (\mathtt{In}_F^{-1} \Rightarrow \phi) \circ \tau \,|\!)_G = \mathtt{cur}'(\!|\, \mathtt{uncur}'(\tau \circ G(\mathtt{cur}'\, \mathtt{id})) \,|\!)^{uc}$$

– *If* $NAT^{uc}_{F,G;H}(\tau)$, *then*

$$\begin{aligned} &(\!|\, \tau, \phi \,|\!)^{uc} \\ &\quad = \mathtt{uncur}(\!|\, \mathtt{cur}(H(\mathtt{uncur\, id}) \circ \tau), \phi \,|\!)^{c} \\ &\quad = \mathtt{uncur}'(\!|\, \mathtt{cur}'(H(\mathtt{uncur}'\, \mathtt{id}) \circ \tau), \phi \,|\!)^{c'} \end{aligned}$$

3 Multi-Anamorphisms

Let F, G, H be endofunctors on $\mathcal{C}$ as before, but assume that a data type exists with base F and a codata type exists with base G.

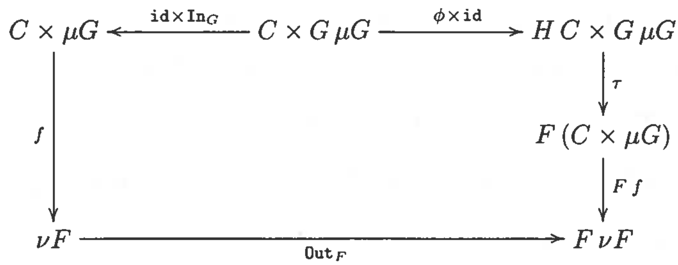
The definition to be presented below of three kinds of $H; F, G$-bi-anamorphisms by characterization hinges upon Proposition 2.

Proposition 2. *Given* $\phi : C \to H\,C$.

- *If* $\tau : H\,X \times G\,Y \to F(X \times Y)$ *is natural in* X, Y *[notation* $NAT^{\mathrm{uc}}_{H;F,G}(\tau)$*], then a unique* $f : C \times \mu G \to \nu F$ *exists such that*

$$\mathtt{Out}_F \circ f \circ (\mathtt{id} \times \mathtt{In}_G) = F\,f \circ \tau \circ (\phi \times \mathtt{id})$$

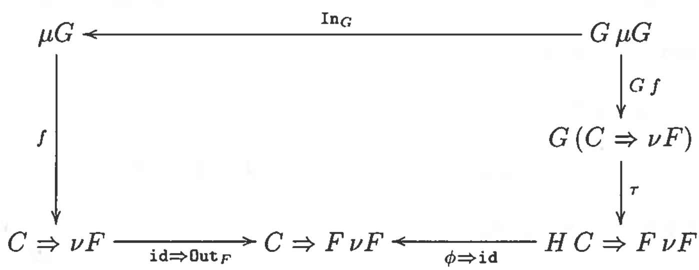
[notation $ANA^{\mathrm{uc}}_{H;F,G}(\tau, \phi, f)$*], i.e.,*



- *If* $\tau : G(X \Rightarrow Z) \to H\,X \Rightarrow F\,Z$ *is natural in* Z, X *[notation* $NAT^{\mathrm{c}'}_{H;F,G}(\tau)$*], then a unique* $f : \mu G \to C \Rightarrow \nu F$ *exists such that*

$$(\mathtt{id} \Rightarrow \mathtt{Out}_F) \circ f \circ \mathtt{In}_G = (\phi \Rightarrow \mathtt{id}) \circ \tau \circ G\,f$$

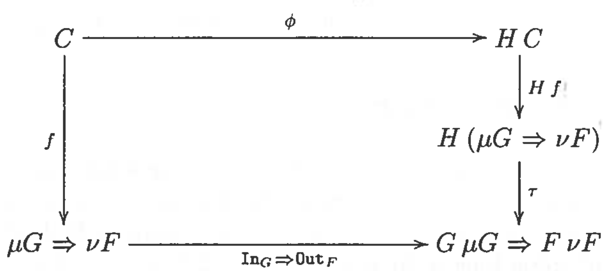
[notation $ANA^{\mathrm{c}'}_{H;F,G}(\tau, \phi, f)$*], i.e.,*



- *If* $\tau : H(Y \Rightarrow Z) \to G\,Y \Rightarrow F\,Z$ *is natural in* Z, Y *[notation* $NAT^{\mathrm{c}}_{H;F,G}(\tau)$*], then a unique* $f : C \to \mu G \Rightarrow \nu F$ *exists such that*

$$(\mathtt{In}_G \Rightarrow \mathtt{Out}_F) \circ f = \tau \circ H\,f \circ \phi$$

[notation $ANA^{\mathrm{c}}_{H;F,G}(\tau, \phi, f)$*], i.e.,*



Proposition 2 is a consequence of the following lemma.

Lemma 3. *Given* $\phi : C \to H\,C$.

- *If* $NAT^{\mathrm{uc}}_{H;F,G}(\tau)$, *then* $\tau \circ (\phi \times \mathtt{In}_G^{-1}) : C \times \mu G \to F(C \times \mu G)$ *and*

$$ANA^{\mathrm{uc}}_{H;F,G}(\tau, \phi, f) \;\equiv\; f = [\,\tau \circ (\phi \times \mathtt{In}_G^{-1})\,]_F$$

- *If* $NAT^{\mathrm{c}'}_{H;F,G}(\tau)$, *then* $(\phi \Rightarrow \mathtt{Out}_F) \circ \tau : G(C \Rightarrow \nu F) \to C \Rightarrow \nu F$ *and*

$$ANA^{\mathrm{c}'}_{H;F,G}(\tau, \phi, f) \;\equiv\; f = (\!|\, (\phi \Rightarrow \mathtt{Out}_F^{-1}) \circ \tau \,|\!)_G$$

- *If* $NAT^{\mathrm{c}}_{H;F,G}(\tau)$, *then* $NAT^{\mathrm{uc}}_{H;F,G}(\mathtt{uncur}(\tau \circ H(\mathtt{cur\,id})))$ *and* $NAT^{\mathrm{c}'}_{H;F,G}(\mathtt{exch}(\tau \circ H(\mathtt{exch\,id})))$ *and*

$$\begin{aligned} & ANA^{\mathrm{c}}_{H;F,G}(\tau, \phi, f) \\ \equiv\;& ANA^{\mathrm{uc}}_{H;F,G}(\mathtt{uncur}(\tau \circ H(\mathtt{cur\,id})), \phi, \mathtt{uncur}\, f) \\ \equiv\;& ANA^{\mathrm{c}'}_{H;F,G}(\mathtt{exch}(\tau \circ H(\mathtt{exch\,id})), \phi, \mathtt{exch}\, f) \end{aligned}$$

The following lemma will only be used below.

Lemma 4. *Given* $\phi : C \to H\,C$.

- *If* $NAT^{\mathrm{uc}}_{H;F,G}(\tau)$, *then* $NAT^{\mathrm{c}}_{H;F,G}(\mathtt{cur}(F\,\mathtt{apply} \circ \tau))$ *and*

$$ANA^{\mathrm{uc}}_{H;F,G}(\tau, \phi, f) \;\equiv\; ANA^{\mathrm{c}}_{H;F,G}(\mathtt{cur}(F\,\mathtt{apply} \circ \tau), \phi, \mathtt{cur} f)$$

- *If* $NAT^{\mathrm{c}'}_{H;F,G}(\tau)$, *then* $NAT^{\mathrm{c}}_{H;F,G}(\mathtt{exch}(\tau \circ G(\mathtt{exch\,id})))$ *and*

$$ANA^{\mathrm{c}'}_{H;F,G}(\tau, \phi, f) \;\equiv\; ANA^{\mathrm{c}}_{H;F,G}(\mathtt{exch}(\tau \circ G(\mathtt{exch\,id})), \phi, \mathtt{exch} f)$$

Proposition 2 entitles us to make the following definition of various kinds of $H; F, G$-bi-anamorphisms.

Definition 2. *Given* $\phi : C \to H\,C$.

- *If* $NAT^{\mathrm{uc}}_{H;F,G}(\tau)$, *then say the* f *such that* $ANA^{\mathrm{uc}}_{H;F,G}(\tau, \phi, f)$ *to be the* uncurried $H; F, G$-bi-anamorphism via τ of ϕ *[notation* $[\,\tau, \phi\,]^{\mathrm{uc}}$*]*.
- *If* $NAT^{\mathrm{c}'}_{H;F,G}(\tau)$, *then say the* f *such that* $ANA^{\mathrm{c}'}_{H;F,G}(\tau, \phi, f)$ *to be the* right-curried $H; F, G$-bi-anamorphism via τ of ϕ *[notation* $[\,\tau, \phi\,]^{\mathrm{c}'}$*]*.
- *If* $NAT^{\mathrm{c}}_{H;F,G}(\tau)$, *then say the* f *such that* $ANA^{\mathrm{c}}_{H;F,G}(\tau, \phi, f)$ *to be the* left-curried $H; F, G$-bi-anamorphism via τ of ϕ *[notation* $[\,\tau, \phi\,]^{\mathrm{c}}$*]*.

The first calculational laws for bi-anamorphisms directly readable off from the defining characterizations are those of *typing and cancellation*.

Corollary 5. *Given* $\phi : C \to H\,C$.

– *If* $NAT^{\mathrm{uc}}_{H;F,G}(\tau)$, *then* $[\tau,\phi]^{\mathrm{uc}} : C \times \mu G \Rightarrow \nu F$ *and*

$$\mathtt{Out}_F \circ [\tau,\phi]^{\mathrm{uc}} \circ (\mathtt{id} \times \mathtt{In}_G) = F\,[\tau,\phi]^{\mathrm{uc}} \circ \tau \circ (\phi \times \mathtt{id})$$

– *If* $NAT^{\mathrm{c}'}_{H;F,G}(\tau)$, *then* $[\tau,\phi]^{\mathrm{c}'} : \mu G \to C \Rightarrow \nu F$ *and*

$$(\mathtt{id} \Rightarrow \mathtt{Out}_F) \circ [\tau,\phi]^{\mathrm{c}'} \circ \mathtt{In}_G = (\phi \Rightarrow \mathtt{id}) \circ \tau \circ G\,[\tau,\phi]^{\mathrm{c}'}$$

– *If* $NAT^{\mathrm{c}}_{H;F,G}(\tau)$, *then* $[\tau,\phi]^{\mathrm{c}} : C \to \mu G \Rightarrow \nu F$ *and*

$$(\mathtt{In}_G \Rightarrow \mathtt{Out}_F) \circ [\tau,\phi]^{\mathrm{c}} = \tau \circ H\,[\tau,\phi]^{\mathrm{c}} \circ \phi$$

More interesting consequences of the characterizations are the bi-ana *fusion* laws.

Corollary 6. *Given* $\phi : C \to H\,C$, $\psi : D \to H\,D$.

– *If* $NAT^{\mathrm{uc}}_{H;F,G}(\tau)$, $h : C \to D$, *then*

$$\psi \circ h = H\,h \circ \phi \;\supset\; [\tau,\psi]^{\mathrm{uc}} \circ (h \times \mathtt{id}) = [\tau,\phi]^{\mathrm{uc}}$$

– *If* $NAT^{\mathrm{c}'}_{H;F,G}(\tau)$, $h : C \to D$, *then*

$$\psi \circ h = H\,h \circ \phi \;\supset\; (h \Rightarrow \mathtt{id}) \circ [\tau,\psi]^{\mathrm{c}'} = [\tau,\phi]^{\mathrm{c}'}$$

– *If* $NAT^{\mathrm{c}}_{H;F,G}(\tau)$, $h : C \to D$, *then*

$$\psi \circ h = H\,h \circ \phi \;\supset\; [\tau,\psi]^{\mathrm{c}} \circ h = [\tau,\phi]^{\mathrm{c}}$$

Also useful in calculations are the laws of ana - cata - bi-ana *fusion*.

Corollary 7. *Given* $\phi : C \to H\,C$, $\xi : E \to F\,E$, $\xi' : G\,E' \to E'$.

– *If* $NAT^{\mathrm{uc}}_{H;F,G}(\tau)$, $h : C \times E' \to E$, *then*

$$\xi \circ h \circ (\mathtt{id} \times \xi') = F\,h \circ \tau \circ (\phi \times \mathtt{id}) \;\supset\; (\!|\xi|\!)_G \circ h \circ (\mathtt{id} \times [\!(\xi')\!]_F) = [\tau,\phi]^{\mathrm{uc}}$$

– *If* $NAT^{\mathrm{c}'}_{H;F,G}(\tau)$, $h : E \to C \Rightarrow E$, *then*

$$(\mathtt{id} \Rightarrow \xi) \circ h \circ \xi' = (\phi \Rightarrow \mathtt{id}) \circ \tau \circ G\,h \;\supset\; (\mathtt{id} \Rightarrow (\!|\xi|\!)_G) \circ h \circ [\!(\xi')\!]_F = [\tau,\phi]^{\mathrm{c}'}$$

– *If* $NAT^{\mathrm{c}}_{H;F,G}(\tau)$, $h : C \to E' \Rightarrow E$, *then*

$$(\xi' \Rightarrow \xi) \circ h = \tau \circ H\,h \circ \phi \;\supset\; ((\!|\xi'|\!)_G \Rightarrow [\!(\xi')\!]_F) \circ h = [\tau,\phi]^{\mathrm{c}}$$

The following laws, finally, can be concluded from Lemmas 3 and 4.

Corollary 8. *Given* $\phi : C \to H\,C$.

– *If* $NAT^{\mathrm{uc}}_{H;F,G}(\tau)$, *then*

$$[\tau,\phi]^{\mathrm{uc}} = [\!(\tau \circ (\phi \times \mathtt{In}_G^{-1}))\!]_F = \mathtt{uncur}[\,\mathtt{cur}(F\,\mathtt{apply} \circ \tau)\,]^{\mathrm{c}}$$

– *If* $NAT^{\mathrm{c}'}_{H;F,G}(\tau)$, *then*

$$[\tau,\phi]^{\mathrm{c}'} = (\!|(\phi \Rightarrow \mathtt{Out}_F^{-1}) \circ \tau|\!)_G = \mathtt{exch}[\,\mathtt{exch}(\tau \circ G(\mathtt{exch}\,\mathtt{id}))\,]^{\mathrm{c}'}$$

– *If* $NAT^{\mathrm{c}}_{H;F,G}(\tau)$, *then*

$$\begin{aligned}&[\tau,\phi]^{\mathrm{c}}\\ &\quad = \mathtt{cur}[\,\mathtt{uncur}(\tau \circ H(\mathtt{cur}\,\mathtt{id})),\phi\,]^{\mathrm{uc}}\\ &\quad = \mathtt{exch}[\,\mathtt{exch}(\tau \circ H(\mathtt{exch}\,\mathtt{id})),\phi\,]^{\mathrm{c}'}\end{aligned}$$

4 Conclusion

Since multi-iteration and multi-coiteration are inherently symmetric schemes of function definition, realizing the fact that some function of interest is definable by one of these two schemes is a discovery worth making, as it means that there are, in fact, many definitions of this function, and situations such as this are the encouraging ones in program calculation. For that reason, well-behaved generic combinators for multi-iteration and multi-coiteration are potentially interesting as library combinators and the laws they obey may well deserve spelling out.

In this note, we described three generic combinators for multi-iteration and as many for multi-coiteration. The laws we listed for these combinators demonstrate that, in case of multi-iteration, the inherent symmetry of the scheme is most transparently captured in the uncurried multi-catamorphism combinator, while the left-curried multi-anamorphism combinator is the most satisfactory representation for the multi-coiteration function definition scheme. The uncurried multi-anamorphism combinator is, however, a neat enough alternative to the left-curried one, if higher types are to be avoided for some reason. Folds and unfolds, generic combinators for iteration and coiteration with parameters, also come in uncurried and curried versions, but since no symmetry is involved in coiteration with parameters, there are no reasons for preferring the left-curried unfold combinator to the uncurried one.

Acknowledgements

Varmo Vene was the first one to tell me about "iterations inducting over multiple data structures" and I owe him thanks for this.

The work reported in this note was partially supported by the Estonian Science Foundation under grant no 2976.

References

[FSZ94] Leonidas Fegaras, Tim Sheard, and Tong Zhou. Improving programs which recurse over multiple inductive structures. In *Proceedings ACM SIGPLAN Workshop on Partial Evaluation and Semantics-Based Program Manipulation, PEPM'94, Orlando, FL, USA, June 1994*, pages 21–32. ACM Press, New York, 1994.

[HIT97] Zhenjiang Hu, Hideya Iwasaki, and Masato Takeichi. An extension of the acid rain theorem. In T. Ida, A. Ohori, and M. Takeichi, editors, *Proceedings 2nd Fuji Intl. Workshop on Functional and Logic Programming, Shonan Village Center, Japan, 1–4 Nov 1996*, pages 91–105. World Scientific, Singapore, 1997.

[Par97] Alberto Pardo. A calculational approach to strong datatypes. In M. Haveraaen and O. Owe, editors, *Selected Papers 8th Nordic Workshop on Programming Theory, NWPT'96, Oslo, Norway, 4-6 Dec 1996*, Research report 248, Dept. of Informatics, Univ. of Oslo, pages 187–197. May 1997.