

Intrinsically correct & total functional programming using guarded recursion

Cass Alexandru¹ Steven Schaefer² Max New²

¹RPTU Kaiserslautern-Landau & Radboud University Nijmegen

²University of Michigan

2026-10-27

- Recent progress in intrinsically correct total functional programming (Alexandru, Urbat, and Wißmann 2026)
- Achieved with the novel notion of a **well founded functor**
- **Gap**: similarities to **locally contractive functors** (Birkedal, Møgelberg, Schwinghammer, and Støvring 2012)
- **Question**: Can we replace wff with lcf? **Answer**: Yes
- Same application, stronger foundation

Technical Overview

- Locally contractive functors give rise to **hylomorphisms**, which describe total functional algorithms
- To instantiate lcf, we need an **enrichment** of our category (indexed datatypes for intrinsically correct programs, corresponding to **families**) in **presheaves**
- We go there and back again: Having instantiated the machinery, we expose a simpler interface for guarded recursion which uses only the **family enrichment of families**

QuickSort as a Coalgebra-to-Algebra Morphism¹

$$\begin{array}{ccc}
 (i_{\leq p}, p, i_{> p}) & \xrightarrow{\text{sort} \times \text{id} \times \text{sort}} & (\text{sort}(i_{\leq p}), p, \text{sort}(i_{> p})) \\
 \uparrow \text{partition} & & \downarrow \text{concat} \\
 ip & \xrightarrow{\text{sort}} & \text{sort}(i_{\leq p}) p \text{sort}(i_{> p})
 \end{array}
 \qquad
 \begin{array}{ccc}
 F_0 I & \xrightarrow{F_1 h} & F_0 O \\
 \uparrow c & (1) & \downarrow a \\
 I & \xrightarrow{h} & O
 \end{array}$$

- for partition: $F := B$ for $BX := 1 + X \times A \times X$, $I := A^*$.
- A c2a-morphism h is called a **hylomorphism** if it is the **unique** c2a-morphism making (1) commute.
- NB: In a non-total programming language, $h := c ; F1 h ; a$ is already a definition. We want to recover a similar definition scheme in a **total** language.

¹due to Capretta, Uustalu, and Vene 2006

Pars pro Toto : Total from Partial Correctness

- The type $A^* \rightarrow 1 + A^* \times A \times A^*$ for partition is **too coarse!**
- As Alexandru, Choudhury, Rot, and Weide 2025: Move to $\text{Set}/\mathcal{MA}$

$$\overline{B} \left(\begin{matrix} X \\ f \end{matrix} \right) := \left(\begin{matrix} 1 \\ * \mapsto \emptyset \end{matrix} \right) + \left(\begin{matrix} \{(l, p, r) \in X \times A \times X \mid \dots\} \\ (l, p, r) \mapsto f(l) \uplus \{p\} \uplus f(r) \end{matrix} \right)$$

- Refined type of partition: $\text{Set}/\mathcal{MA}[(A^*, \text{toBag}), \overline{B}(A^*, \text{toBag})]$
- Note: The multiset indices in functorial positions are smaller than the input index: $f(l), f(r) < f(l) \uplus \{p\} \uplus f(r)$

Approach

- Observation: “indices in functorial positions are smaller than the input index”

$$\begin{array}{ccc}
 F_0 I & \xrightarrow{F_1 h|_{<i}} & F_0 O \\
 \uparrow c & (1) & \downarrow a \\
 I & \xrightarrow{h_i} & O
 \end{array}$$

- $\Rightarrow$ we could construct a solution to (1) “as an inhabitant of the homset using well founded recursion”!
- Next: How this is **made formal**. We begin with moving from Hom-**sets** to Hom-**objs** for some category with the requisite structure for constructing an inhabitant using well founded recursion

Enriched Categories & Enrichment

Fix a set of objects C_0 and a monoidal category $(V, \otimes, \mathbb{1})$.

Definition (Enriched category)

- for every $x, y: C_0$ the internal Hom $[X, Y]$;
 - an internal identity morphism $\text{id}_X: V(\mathbb{1}, [X, X])$ & sequencing operation $V([X, Y] \otimes [Y, Z], [X, Z])$;
 - internal counterparts to unitality & associativity coherences
-
- For an enriched category C' there is the **underlying** plain (i.e. Set-enriched) category with $\Gamma C'[X, Y] := V(\mathbb{1}, [X, Y])$
 - But we're starting **out** with a plain category!

Definition (Enrichment of a category C)

- same data & laws as above;
- $C \cong \Gamma C'$, in particular $[-]: C(X, Y) \cong V(\mathbb{1}, [X, Y])$

Families enriched in Families

- Families (Set^{C_0}) are Cartesian monoidal, inheriting that structure pointwise from Set
- $\mathbb{1} = \prod_{c \in C_0} \mathbb{1}_{\text{Set}}$
- $[X, Y]_c = \text{Set}(Xc, Yc)$
- $\text{Set}^{C_0}(X, Y) = \prod_{c \in C_0} \text{Set}(Xc, Yc) \cong \prod_{c \in C_0} \text{Set}(\mathbb{1}_{\text{Set}}, \text{Set}(Xc, Yc)) = \text{Set}^{C_0}(\mathbb{1}, [X, Y])$

Enrichment of a Functor

Definition (Enrichment of a Functor)

Given:

- categories C, D with enrichments $EC = ([-, -], [-])$ and $ED = ([-, -]', [-]')$ in monoidal category V ;
- a functor $F: C \rightarrow D$

An **enrichment** of F in EC, ED is an internal functorial action

$$F'_1: V([X, Y], [F_0 X, F_0 Y]')$$

- obeying internal counterparts to preservation of id and comp.;
- and such that F_1 and F'_1 **agree**: $([-]: C(X, Y) \cong V(\mathbb{1}, [X, Y]))$
 $\forall X, Y: C_0, f: C(X, Y). [F_1 f]' = [f]; F'_1$

Locally Contractive Functors

Definition

Given:

- V a “model of guarded recursive terms”
- Categories C, D with enrichments $(EC, [-]), (ED, [-]')$ in V
- Functor $F: C \rightarrow D$
- An enriched functor $F': \blacktriangleright EC \rightarrow ED$ with F_0 as object map

F is **locally contractive** if it has as enrichment the sequence of enrichments $EC \xrightarrow{\text{next}} \blacktriangleright EC \xrightarrow{F'} ED$, i.e.

$$\forall x, y : C_0, f : C(x, y) . [F_1 f]' = [f]; \text{next}; F'_1$$

- NB: This is actually a corollary of the normal definition, for F with no preexisting enrichment.

the boundary of the representable: $\text{PSh}(A, <)$ as a model of guarded recursive terms

Need : $\blacktriangleright$, next, Löb

- $\downarrow i = \{j \leq i\}. (\downarrow i) := A(-, i) = \mathfrak{J}i. \Downarrow i := \{j < i\}$
- $\blacktriangleright Xi = \lim_{j < i} Xj = \text{PSh}(A)(\partial \mathfrak{J}i, X)$
- next: $\text{Endo}(\text{PSh}(A))(\text{Id}, \blacktriangleright)$
- next given by functoriality of the presheaf
- Löb : for $H: \text{PSh}(A)(\blacktriangleright [X, Y], [X, Y])$ $\text{Löb}(H): \text{PSh}(A)(\mathbb{1}, [X, Y])$
s.t. $\text{Löb}H = \text{Löb}H; \text{next}; H$
- Löb given by well founded induction

Locally Contractive Functors give Hylomorphisms

Functional Programmer Proof sketch

$$\begin{array}{ccc}
 F_0 I & \overset{1?}{\dashrightarrow} & F_0 O \\
 c \uparrow & (1) & \downarrow a \\
 I & \overset{0?}{\dashrightarrow} & O
 \end{array}$$

- Note that in (1), the functional $H(h) := c; h; a$ has type $(F_0 I \rightarrow F_0 O) \rightarrow (I \rightarrow O)$.
- Local contractivity of F gives us $F'_1: \blacktriangleright(I \rightarrow O) \rightarrow (F_0 I \rightarrow F_0 O)$.
- Putting them together we get

$$\blacktriangleright(I \rightarrow O) \xrightarrow{F'_1} (F_0 I \rightarrow F_0 O) \xrightarrow{H} (I \rightarrow O)$$
- Applying Löb: $(\blacktriangleright [I, O] \rightarrow [I, O]) \rightarrow [I, O]$ gives us $\text{Löb}(F'_1; H): (I \rightarrow O)$.

Change of Base of Enrichment

Let $(M, *, I)$ and $(N, \otimes, J)$ be monoidal categories.

Definition (Lax monoidal functor)

A functor $F: M \rightarrow N$ is **lax monoidal** if it has comparison arrows:

- $\varepsilon: N(J, FI)$, and
- $\mu_{X,Y}: N(Fx \otimes FY, F(x * y))$ agreeing with the unitors and associators of M and N .

Proposition (Change of Base)

Let $EC = ([-, -], \dots)$ be an enriched category for C_0 in M . Let $F: M \rightarrow N$ be a lax monoidal functor. We can define an enriched category $F^*(EC)$ for C_0 in N called the base-change of EC along F with $[X, Y] = F([X, Y])$.

Families Enriched in Presheaves

- We already saw families enriched in families
- To enrich them in Presheaves, we need a lax monoidal functor $\text{Fam}(A) \rightarrow \text{PSh}(A)$
- The functor that does the job is the **cofree presheaf functor**:

$$U \dashv \text{Cofree}, \text{Cofree}(B)(x) := \prod_y \text{Set}(A(y, x), B(y)) = \prod_{y \leq x} B(y)$$
- lax monoidal structure derives from RAPL

Families are Enough: There and Back Again

- We can define a version of $\blacktriangleright$ for families by $\blacktriangleright' := U \circ \blacktriangleright \circ \text{Cofree}$
- Have the enrichment of families in families with internal homs $[-, -]$
- for lcf, the contractive action of F lives in $\text{Psh}(A) (\blacktriangleright(\text{Cofree } [X, Y]), \text{Cofree } [FX, FY])$
- $\cong \langle \text{transpose this homset along } U \dashv \text{Cofree!} \rangle$
- $\text{Fam}(A) (U \blacktriangleright \text{Cofree } [X, Y], [FX, FY]) = \text{Fam}(A) (\blacktriangleright' [X, Y], [FX, FY])$
- The agreement condition can be similarly ported to the setting of families only
- This yields an expression of lcf for enrichment in families
- Transposed along $U \dashv \text{Cofree}$, this is equivalent to the usual lcf

Conclusion

- Novel application of locally contractive functors to total functional programming
- Can be used “behind the scenes” as a theoretical underpinning while the programmer need only work in the setting of indexed datatypes (families)
- Illustrates an interesting novel usecase for lcf in a category which is not self-enriched

TODOs

- Actually see which existing results about lcf are **actually** useful to instantiate for applications
- In particular: Can we use composition theorems to reduce client obligations? Famously local contractivity as is closed under composition with enriched functors
- See whether solving domain equations using lcf is also useful in this setting
- Technical: Currently we're switching back between enriched categories & enrichments because some lax monoidal functors do not preserve underlying categories. Need to make this more elegant