Initial Algebras Unchained

A Novel Initial Algebra Construction Formalized in Agda

Thorsten Wissmann
`thorsten-wissmann.de`

Stefan Milius
`stefan-milius.eu`

FAU Erlangen-Nürnberg, Germany

LICS, July 08, 2024

Initial Algebras Unchained

A Novel Initial Algebra Construction Formalized in Agda

Thorsten Wissmann
thorsten-wissmann.de

Stefan Milius
stefan-milius.eu

FAU Erlangen-Nürnberg, Germany

Initial Algebras Unchained

A Novel Initial Algebra Construction Formalized in Agda

Initial Algebras Unchained

A Novel Initial Algebra Construction Formalized in Agda

Initial Algebras Unchained

A Novel Initial Algebra Construction Formalized in Agda

E.g. $FX = \{\bullet\} + X \times X$

$$\begin{tikzcd} FA \arrow[d, "a"] \\ A \end{tikzcd}$$

For a fixed functor $F\colon \mathsf{Set} \to \mathsf{Set}$

Operations

on a Carrier Set

E.g. $A = \mathbb{N}$
$a(\bullet) = 0$
$a(\ell, r) = 1 + \max(\ell, r)$

Initial Algebras Unchained

A Novel Initial Algebra Construction Formalized in Agda

E.g. $FX = \{\bullet\} + X \times X$

T = all binary trees

Constructors

Data structures of type F

$$\begin{tikzcd} FT \arrow[d, "t"] \\ T \end{tikzcd}$$

For a fixed functor $F\colon \mathsf{Set} \to \mathsf{Set}$

Operations

on a Carrier Set

$$\begin{tikzcd} FA \arrow[d, "a"] \\ A \end{tikzcd}$$

E.g. $A = \mathbb{N}$
$a(\bullet) = 0$
$a(\ell, r) = 1 + \max(\ell, r)$

Initial Algebras Unchained

A Novel Initial Algebra Construction Formalized in Agda

E.g. $FX = \{\bullet\} + X \times X$

For a fixed functor $F\colon \mathrm{Set} \to \mathrm{Set}$

$$\begin{tikzcd} FT \arrow[r, "Fh"] \arrow[d, "t"'] & FA \arrow[d, "a"] \\ T \arrow[r, dashed, "h"'] & A \end{tikzcd}$$

T = all binary trees

Constructors

Data structures of type F

Recursion & induction

h = height

Operations

on a Carrier Set

E.g. $A = \mathbb{N}$
$a(\bullet) = 0$
$a(\ell, r) = 1 + \max(\ell, r)$

Initial Algebras Unchained

A Novel Initial Algebra Construction Formalized in Agda

E.g. $FX = \{\bullet\} + X \times X$

T = all binary trees

h = height

For a fixed functor $F\colon \mathsf{Set} \to \mathsf{Set}$

$$
\begin{tikzcd}
FT \arrow[r, "Fh"] \arrow[d, "t"'] & FA \arrow[d, "a"] \\
T \arrow[r, dashed, "h"'] & A
\end{tikzcd}
$$

Constructors

Data structures of type F

Recursion & induction

Operations

on a Carrier Set

E.g. $A = \mathbb{N}$
$a(\bullet) = 0$
$a(\ell, r) = 1 + \max(\ell, r)$

Initial Algebras Unchained

A Novel Initial Algebra Cons

Theorem Adámek '74

Initial F-Algebra
$$=$$
$$\varnothing \to F\varnothing \to \ldots F^{\omega}\varnothing \to \cdots$$
by transfinite induction

☹ not in type-theory e.g. Agda

E.g. $FX = \{\bullet\} + X \times X$

T = all binary trees

h = height

For a fixed functor $F \colon \mathrm{Set} \to \mathrm{Set}$

$$
\begin{tikzcd}
FT \arrow[r, "Fh"] \arrow[d, "t"'] & FA \arrow[d, "a"] \\
T \arrow[r, dashed, "h"'] & A
\end{tikzcd}
$$

Constructors

Data structures of type F

Recursion & induction

Operations

on a Carrier Set

E.g. $A = \mathbb{N}$
$a(\bullet) = 0$
$a(\ell, r) = 1 + \max(\ell, r)$

Initial Algebras Unchained

New

Theorem

Initial F-Algebra
$=$
Colimit of all "finite" recursive F-coalgebras

fully formalized in Agda

Theorem Adámek '74

Initial F-Algebra
$=$
$$\varnothing \to F\varnothing \to \ldots F^{\omega}\varnothing \to \cdots$$
by transfinite induction

not in type-theory e.g. Agda

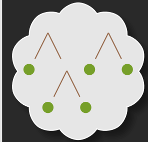
Theorem for an accessible $F\colon \mathscr{C} \to \mathscr{C}$ on an accessible category $\mathscr{C}$

Initial F-Algebra	=	Colimit of all recursive F-coalgebras with presentable ("finite") carrier

Theorem for an accessible $F\colon \mathscr{C} \to \mathscr{C}$ on an accessible category $\mathscr{C}$

Initial F-Algebra = Colimit of all recursive F-coalgebras with presentable ("finite") carrier

Recursive F-Coalgebra (R, r)



$$\begin{tikzcd} FR \arrow[r, "Fh"] & FA \arrow[d, "a", "\forall"'] \\ R \arrow[u, "r"] \arrow[r, dashed, "\exists! h"] & A \end{tikzcd}$$

☺ Well-founded induction

Osius '74

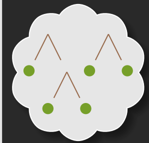
☺ Divide & Conquer Algorithms (e.g. Quicksort)

Capretta, Uustalu, Vene '06

Theorem for an accessible $F\colon \mathscr{C} \to \mathscr{C}$ on an accessible category $\mathscr{C}$

Initial F-Algebra = Colimit of all recursive F-coalgebras with presentable ("finite") carrier

Recursive F-Coalgebra (R, r)



$$\begin{tikzcd} FR \arrow[r, "Fh"] & FA \arrow[d, "a", "\forall"'] \\ R \arrow[u, "r"] \arrow[r, dashed, "\exists! h"] & A \end{tikzcd}$$

☺ Well-founded induction

Osius '74

☺ Divide & Conquer Algorithms (e.g. Quicksort)

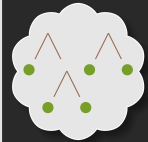
Capretta, Uustalu, Vene '06

Finiteness in a category $\mathscr{C}$

Theorem for an accessible $F\colon \mathscr{C} \to \mathscr{C}$ on an accessible category $\mathscr{C}$

Initial F-Algebra = Colimit of all recursive F-coalgebras with presentable ("finite") carrier

Recursive F-Coalgebra (R, r)



$$\begin{tikzcd}
FR \arrow[r, "Fh"] & FA \arrow[d, "a", "\forall"'] \\
R \arrow[u, "r"] \arrow[r, dashed, "\exists! h"] & A
\end{tikzcd}$$

☺ Well-founded induction

Osius '74

☺ Divide & Conquer Algorithms (e.g. Quicksort)

Capretta, Uustalu, Vene '06

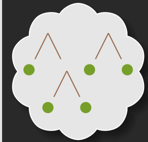
Finiteness in a category $\mathscr{C}$

~~Require Cardinals~~

Theorem for an accessible $F\colon \mathscr{C} \to \mathscr{C}$ on an accessible category $\mathscr{C}$

Initial F-Algebra = Colimit of all recursive F-coalgebras with presentable ("finite") carrier

Recursive F-Coalgebra (R, r)



$$\begin{tikzcd} FR \arrow[r, "Fh"] & FA \arrow[d, "a", "\forall"'] \\ R \arrow[u, "r"] \arrow[r, dashed, "\exists! h"] & A \end{tikzcd}$$

☺ Well-founded induction

Osius '74

☺ Divide & Conquer Algorithms (e.g. Quicksort)

Caretta, Uustalu, Vene '06

Finiteness in a category $\mathscr{C}$

Fix a class of filtered diagrams

~~Require Cardinals~~

X presentable $:\Longleftrightarrow$ $\mathscr{C}(X, -)$ preserves colimits of this class

finite

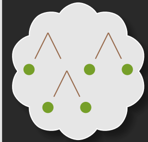
$\mathscr{C}$ accessible $:\Longleftrightarrow$ build all objects from a set of presentables

Theorem for an accessible $F\colon \mathscr{C} \to \mathscr{C}$ on an accessible category $\mathscr{C}$

Initial F-Algebra = Colimit of all recursive F-coalgebras with presentable ("finite") carrier

☺ Instances: λ-presentable finite

Recursive F-Coalgebra (R, r)



$$\begin{tikzcd} FR \arrow[r, "Fh"] & FA \arrow[d, "\forall", swap] \arrow[d, "a"] \\ R \arrow[u, "r"] \arrow[r, dashed, "\exists! h"] & A \end{tikzcd}$$

☺ Well-founded induction

Osius '74

☺ Divide & Conquer Algorithms (e.g. Quicksort)

Capretta, Uustalu, Vene '06

Finiteness in a category $\mathscr{C}$

Fix a class of filtered diagrams

~~Require Cardinals~~

X presentable $:\Longleftrightarrow$ $\mathscr{C}(X, -)$ preserves colimits of this class

finite

$\mathscr{C}$ accessible $:\Longleftrightarrow$ build all objects from a set of presentables

Proof Structure

F-$\mathsf{Coalg}^{\mathsf{fin}}_{\mathsf{rec}}$: Diagram of all recursive (R, r) with $R \in \mathscr{C}_{\mathsf{p}}$

Colimit

Locally finrec coalgebra $c\colon T \to FT$

$Fc\colon FT \to FFT$ is also a locally finrec coalgebra

There exists $h\colon (FT, Fc) \to (T, c)$

Colimit Injection: is the unique $h\colon (R, r) \to (T, c)$ for all $(R, r) \in F$-$\mathsf{Coalg}^{\mathsf{fin}}_{\mathsf{rec}}$

Universal Property: there is a unique $(R, r) \xrightarrow{\exists!} (T, c)$ for all (locally) finrec (R, r)

The identity id_A is the only endomorphism $(T, c) \to (T, c)$

Lambek's Lemma

c is an isomorphism $T \cong FT$

Initial Algebra $c^{-1}\colon FT \to T$

Agda Formalization

`arxiv.org/src/2405.09504/anc/index.html`

>5000 lines (29 files)

using `agda-categories` 0.2.0

Agda Formalization

`arxiv.org/src/2405.09504/anc/index.html`

>5000 lines (29 files)

using `agda-categories` 0.2.0

? Why does the colimit of all presentable ("finite") recursive coalgebras exist? ?

Agda Formalization

`arxiv.org/src/2405.09504/anc/index.html`

>5000 lines (29 files)

using `agda-categories` 0.2.0

? Why does the colimit of all presentable ("finite") recursive coalgebras exist? ?

Set Theory

presentable/finite coalgebras:
(up to iso) just a set

recursive finite coalgebras:
an even smaller set

Agda Formalization

`arxiv.org/src/2405.09504/anc/index.html`

>5000 lines (29 files)

using `agda-categories` 0.2.0

? Why does the colimit of all presentable ("finite") recursive coalgebras exist? ?

Set Theory

presentable/finite coalgebras:
  (up to iso) just a set
recursive finite coalgebras:
  an even smaller set

Agda's Type Theory

finite coalgebras:
  on the set level ℓ
recursive finite coalgebras:
  one level higher $\ell + 1$
  (*for all* algebras ...)

Agda Formalization

`arxiv.org/src/2405.09504/anc/index.html`

>5000 lines (29 files)

using `agda-categories` 0.2.0

? Why does the colimit of all presentable ("finite") recursive coalgebras exist? ?

Set Theory

presentable/finite coalgebras:
(up to iso) just a set

recursive finite coalgebras:
an even smaller set

Agda's Type Theory

finite coalgebras:
on the set level ℓ

recursive finite coalgebras:
one level higher $\ell + 1$
(*for all* algebras ...)

LEM to the rescue

a finite coalgebra is
recursive or not recursive

In Agda

Theorem For $F\colon \mathscr{C} \to \mathscr{C}$ accessible on an accessible category $\mathscr{C}$:

Initial F-Algebra = Colimit of all recursive F-coalgebras with presentable carrier

☺ Instances: λ-presentable finite

Paper

Agda Doc

In Agda

Theorem For $F\colon \mathscr{C} \to \mathscr{C}$ accessible on an accessible category $\mathscr{C}$:

Initial F-Algebra = Colimit of all recursive F-coalgebras with presentable carrier

☺ Instances:
λ-presentable
finite

Conclusions & Future Work

- Agda formalization is challenging
- Decision procedure for recursiveness of "finite" coalgebra?
  (its type: $\forall (C, c)\colon C$ finite $\to \text{recursive}(c) \vee \neg\text{recursive}(c)$)
- Similar theorem for well-founded coalgebras?
- Concrete example for a non-finitary functor in Agda

long chain!

Paper

Agda Doc

Adámek, Jiří. "Free algebras and automata realizations in the language of categories". eng. *Commentationes Mathematicae Universitatis Carolinae* 015.4 (1974), pp. 589–602. URL: http://eudml.org/doc/16649.

Capretta, Venanzio, Tarmo Uustalu, Varmo Vene. "Recursive coalgebras from comonads". *Inf. Comput.* 204.4 (2006), pp. 437–468. DOI: 10.1016/j.ic.2005.08.005. URL: https://doi.org/10.1016/j.ic.2005.08.005.

Osius, Gerhard. "Categorical set theory: A characterization of the category of sets". *Journal of Pure and Applied Algebra* 4.1 (1974), pp. 79–119. ISSN: 0022-4049. DOI: https://doi.org/10.1016/0022-4049(74)90032-2. URL: https://www.sciencedirect.com/science/article/pii/0022404974900322.

Initial Algebras

For $F\colon \mathscr{C} \to \mathscr{C}$

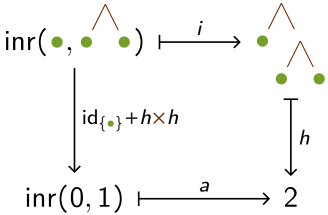
F-Algebra: $A \in \mathscr{C}$ with $a\colon FA \to A$.

Initial F-Algebra: unique homomorphism to every F-algebra.

Initial Algebra for $FX = \{\bullet\} + X \times X$

Initial F-algebra is carried by $I = \textit{all}$ binary trees

$$\begin{tikzcd}
\{\bullet\} + I \times I \arrow[r, "i"] \arrow[d, "\mathrm{id}_{\{\bullet\}}+h\times h"] & I \arrow[d, dashed, "\exists! h"] \\
\{\bullet\} + \mathbb{N} \times \mathbb{N} \arrow[r, "a", "{\substack{\bullet \mapsto 0 \\ (k,n) \mapsto 1+\max(k,n)}}"'] & \mathbb{N}
\end{tikzcd}$$



For $F\colon \mathscr{C} \to \mathscr{C}$

F-Coalgebra: $C \in \mathscr{C}$ with $c\colon C \to FC$.
Coalgebra-to-Algebra morphism: $s\colon (D, d) \to (A, a)$

$$\begin{tikzcd}
C \arrow[r, "g"] \arrow[d, "c"'] & D \arrow[r, "s"] \arrow[d, "d"'] & A \arrow[r, "h"] & B \\
FC \arrow[r, "Fg"] & FD \arrow[r, "Fs"] & FA \arrow[r, "Fh"] \arrow[u, "a"'] & FB \arrow[u, "b"']
\end{tikzcd}$$

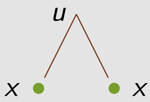
Coalgebra (D, d) is *recursive* if

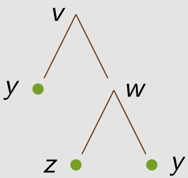
for all $a\colon FA \to A$, there is a unique
Coalgebra-to-Algebra morphism $s\colon (D, d) \to (A, a)$.

Under mild conditions ...

Recursive = Well-Founded = “no infinite path”

R	r	FR
u	$\mapsto$	$\mathrm{inr}(x, x)$
v	$\mapsto$	$\mathrm{inr}(y, w)$
w	$\mapsto$	$\mathrm{inr}(z, y)$
x	$\mapsto$	$\mathrm{inl}(\bullet)$
y	$\mapsto$	$\mathrm{inl}(\bullet)$
z	$\mapsto$	$\mathrm{inl}(\bullet)$





For every $b\colon FB \to B$:

$$h\colon R \longrightarrow B$$

$$h(x) := h(y) := h(z) := b(\mathrm{inl}(\bullet))$$

$$h(u) := b(\mathrm{inr}(h(x), h(x)))$$

$$h(w) := b(\mathrm{inr}(h(z), h(y)))$$

$$h(v) := b(\mathrm{inr}(h(y), h(w)))$$

Definition: Category $\mathscr{D}$ filtered, if ...

- $\mathscr{D}$ is non-empty,
- for every $X, Y \in \mathscr{D}$ there is an *upper bound* $Z \in \mathscr{D}$, that is, there are morphisms $X \to Z$ and $Y \to Z$,
- for every $f, g \colon X \to Y$ there is some $Z \in \mathscr{D}$ and some $h \colon Y \to Z$ with $h \circ g = h \circ f$.

Finitary functor = functor preserving colimits of filtered diagrams

Definition: Object $X \in \mathscr{C}$ is finitely presentable, if ...

$\mathscr{C}(X, -) \colon \mathscr{C} \to \mathsf{Set}$ preserves filtered colimits.

Examples

Sets/Graphs/Posets: finite sets/graphs/posets
Nom: orbit-finite nominal sets
Vector-spaces: finite dimensional vector spaces.
Monoids: defined by finitely many generators + equations.

Category S/X: for a set $S \subseteq \mathbf{obj}\mathscr{C}$ and $X \in \mathscr{C}$

- objects (S, f) for $S \in \mathscr{S}$ and $f\colon S \to X$ (in $\mathscr{C}$), and
- morphisms $h\colon (S, f) \to (T, g)$ for $h\colon S \to T$ with $g \circ h = f$

Canonical Diagram $U_{S/X}\colon S/X \to \mathscr{C}$.

Category $\mathscr{C}$ is locally finitely presentable, if ...

it is cocomplete and
has a set S of finitely presentable objects, s.t.
$X = \operatorname{colim} U_{S/X}$ for all $X \in \mathscr{C}$.

Fix a property Fil on categories entailing filteredness.

Definition

- An object $X \in \mathscr{C}$ is (Fil-)presentable if its hom functor $\mathscr{C}(X, -)$ preserves colimits of diagrams $D\colon \mathscr{D} \to \mathscr{C}$ with $\mathscr{D} \in \mathsf{Fil}$.
- A category $\mathscr{C}$ is *weakly locally presentable* (*weakly lp*, for short) provided that
  - there is a set $\mathscr{C}_{\mathsf{p}} \subseteq \mathscr{C}$ of (Fil-)presentable objects,
  - for all $X \in \mathscr{C}$, the coslice category $(\mathscr{C}_{\mathsf{p}}/X)$ lies in Fil,
  - for all $X \in \mathscr{C}$, the object X is the colimit of $U_{\mathscr{C}_{\mathsf{p}}/X}\colon \mathscr{C}_{\mathsf{p}}/X \to \mathscr{C}$,
  - the coproduct $X + Y$ of presentable objects X, Y exists.