An Elementary Construction of Final Coalgebras

Raul Andres Leal[1] Clemens Kupke[2]

[1]ILLC
Universiteit van Amsterdam

[2]CWI
Amsterdam

27-06-2008
International Category Theory Conference

Outline

1. Once upon a time there was a set...
   - Coalgebras
   - Final coalgebras
   - Languages for coalgebras
   - An elementary construction
2. Pointless Languages
3. Monadic categories over *Set*
   - A still elementary construction

Outline

1. Once upon a time there was a set. . .
   - Coalgebras
   - Final coalgebras
   - Languages for coalgebras
   - An elementary construction

2. Pointless Languages

3. Monadic categories over *Set*
   - A still elementary construction

Coalgebras: Intuitions

- T-Coalgebras = (T-Algebras)^{op}.
- T-Coalgebras are NOT Eilenberg-Moore Coalgebras. No Axioms
- Observation vs Construction.
- In *Set*, coalgebras represent machines from the point of view of the users:

Battery Chargers (one button machines): $\alpha : A \longrightarrow 1 + A$.

Non deterministic transitions systems: $\alpha : A \longrightarrow \mathcal{P}(A)$

Coalgebras: Formalities

Definition

A coalgebra for a functor $T : \mathbb{A} \longrightarrow \mathbb{A}$ is a morphism (in $\mathbb{A}$)

$$\alpha : A \longrightarrow TA.$$

Morphisms:

A coalgebra morphism $f : \alpha \longrightarrow \beta$ is defined in the "natural" way. We write $Coalg(T)$ for the category of T-coalgebras and coalgebra morphisms.

The Behavior of a State

Behavior

The behavior of a state is the “evolution” of the state.

Under appropriate circumstances we can give a concrete representation to the observable behavior. Using Final coalgebras.

Definition

A final T coalgebra is a terminal object in $Coalg(T)$

The Behavior of a State

Behavior

The behavior of a state is the “evolution” of the state.

Under appropriate circumstances we can give a concrete representation to the observable behavior. Using Final coalgebras.

Definition

A final T coalgebra is a terminal object in $Coalg(T)$

The observable behavior of $\alpha : A \longrightarrow 1 + A$

A state s can...

- lead to the halt of the machine, or
- lead us one step closer to the halt of the machine.
- We will never see the machine stop.

A concrete presentation

Consider the set

$$\overline{\mathbb{N}} = \mathbb{N} \cup \{\infty\}$$

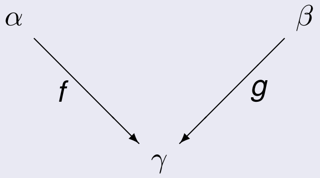
and the function $\zeta : \overline{\mathbb{N}} \longrightarrow 1 + \overline{\mathbb{N}}$ defined as follows:

$$\zeta(0) = *; \quad \zeta(n+1) = n; \quad \zeta(\infty) = \infty.$$

Behavioral Equivalence of States

Definition

Over *Set*, two states $s \in \alpha$ and $s' \in \beta$ are *behavioral equivalent, written* $s \sim s'$, iff there exists a coalgebra γ and morphisms



such that $f(s) = g(s')$.

Nice properties of final coalgebras

Important

Final coalgebras code behavioral equivalence semantically i.e.

$$s \sim s' \text{ iff } f_\alpha(s) = f_\beta(s')$$

Abstract coalgebraic languages

Definition

An *abstract coalgebraic language* is a set $\mathcal{L}$ together with a function

$$Th_\alpha : A \longrightarrow \mathcal{P}\mathcal{L}$$

for each coalgebra $\alpha : A \longrightarrow TA$.

Expressive languages

Important

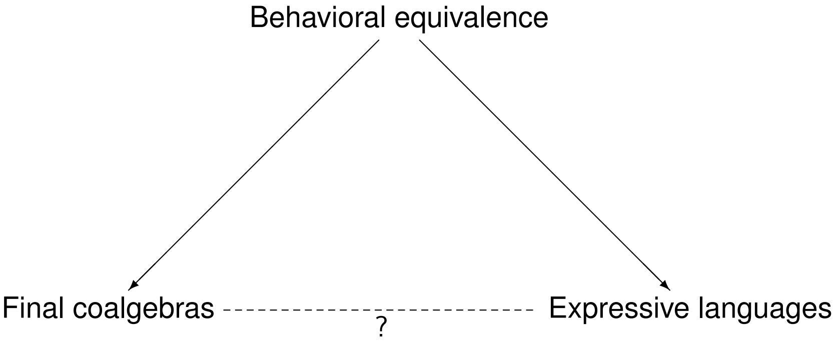
Expressive languages code behavioral equivalence syntactically, i.e.

$$s \sim s' \text{ iff } Th_\alpha(s) = Th_\beta(s').$$

Summary

In *Set* we have:

$$f_\alpha(s) = f_\beta(s') \text{ iff } s \sim s' \text{ iff } Th_\alpha(s) = Th_\beta(s')$$



Our main topic

Theorem (Goldblatt)

For every functor $T : Set \longrightarrow Set$, the existence of a final T-coalgebra is equivalent to the existence of an expressive language with respect to behavioral equivalence.

From final coalgebras to expressive languages (Sets)

Theorem

If there exists a final coalgebra ζ, there exists an expressive abstract coalgebraic language.

Proof.

Take $\mathcal{L} = Z$ and $Th_\alpha = f_\alpha$. □

From expressive languages to final coalgebras

Theorem

If there exists an expressive language, there exists a final coalgebra.

A point wise definition of final coalgebras (still in *Set*)

Proof.

1. Take $Z = \{\Phi \subseteq \mathcal{L} \,|\, (\exists\alpha)(\exists s \in \alpha)(Th_\alpha(s) = \Phi)\}$.
2. construct $\zeta : Z \longrightarrow TZ$ as follows:

$$\begin{array}{ccc} A & \xrightarrow{\;Th_\alpha\;} & Z \\ {\scriptstyle\alpha}\downarrow & & \downarrow{\scriptstyle\zeta} \\ TA & \xrightarrow[T(Th_\alpha)]{} & TZ \end{array}$$

   If $Th_\alpha(s) = \Phi$ then $\zeta(\Phi) = T(Th_\alpha)\alpha(s)$
3. Since $\mathcal{L}$ is expressive, ζ is well defined and $Th_\alpha : \alpha \longrightarrow \zeta$ is the only morphism of coalgebras.

□

A point wise definition of final coalgebras (still in *Set*)

Proof.

1. Take $Z = \{\Phi \subseteq \mathcal{L} \mid (\exists\alpha)(\exists s \in \alpha)(Th_\alpha(s) = \Phi)\}$.

Those are the states of a final coalgebra

2. construct $\zeta : Z \longrightarrow TZ$ as follows:

$$\begin{array}{ccc} A & \xrightarrow{Th_\alpha} & Z \\ \downarrow{\scriptstyle \alpha} & & \downarrow{\scriptstyle \zeta} \\ TA & \xrightarrow[T(Th_\alpha)]{} & TZ \end{array}$$

If $Th_\alpha(s) = \Phi$ then $\zeta(\Phi) = T(Th_\alpha)\alpha(s)$

3. Since $\mathcal{L}$ is expressive, ζ is well defined and $Th_\alpha : \alpha \longrightarrow \zeta$ is the only morphism of coalgebras.

A point wise definition of final coalgebras (still in *Set*)

Proof.

1. Take $Z = \{\Phi \subseteq \mathcal{L} \mid (\exists\alpha)(\exists s \in \alpha)(Th_\alpha(s) = \Phi)\}$.
2. construct $\zeta : Z \longrightarrow TZ$ as follows:

$$\begin{array}{ccc} A & \xrightarrow{\;Th_\alpha\;} & Z \\ {\scriptstyle \alpha}\big\downarrow & & \big\downarrow{\scriptstyle \zeta} \\ TA & \xrightarrow[\;T(Th_\alpha)\;]{} & TZ \end{array}$$

   If $Th_\alpha(s) = \Phi$ then $\zeta(\Phi) = T(Th_\alpha)\alpha(s)$

3. Since $\mathcal{L}$ is expressive, ζ is well defined and $Th_\alpha : \alpha \longrightarrow \zeta$ is the only morphism of coalgebras.

□

A point wise definition of final coalgebras (still in *Set*)

Proof.

1. Take $Z = \{\Phi \subseteq \mathcal{L} \mid (\exists\alpha)(\exists s \in \alpha)(Th_\alpha(s) = \Phi)\}$.
2. construct $\zeta : Z \longrightarrow TZ$ as follows:

$$\begin{array}{ccc} A & \xrightarrow{\;Th_\alpha\;} & Z \\ {\scriptstyle\alpha}\big\downarrow & & \big\downarrow{\scriptstyle\zeta} \\ TA & \xrightarrow[\;T(Th_\alpha)\;]{} & TZ \end{array}$$

   If $Th_\alpha(s) = \Phi$ then $\zeta(\Phi) = T(Th_\alpha)\alpha(s)$

3. Since $\mathcal{L}$ is expressive, ζ is well defined and $Th_\alpha : \alpha \longrightarrow \zeta$ is the only morphism of coalgebras.

□

Outline

Farewell to *Set*

Our aim

To construct final coalgebras over categories different than *Set*

First issue

What is an expressive language outside *Set*?

Pointless languages I

Definition

An *abstract coalgebraic language* is a set $\mathcal{L}$ together with a function

$$Th_{\alpha} : A \longrightarrow \mathcal{P}\mathcal{L}$$

for each coalgebra $\alpha : A \longrightarrow TA$.

- In our construction we are not using the points (formulas) in $\mathcal{L}$.
- In the "real live" $\mathcal{L}$ has an algebraic structure and...
- in the boolean case, our theory maps are functions

$$Th_{\alpha} : A \longrightarrow Uf(\mathcal{L}).$$

Pointless languages II

Definition

Given a functor $T : \mathbb{A} \longrightarrow \mathbb{A}$, an *abstract coalgebraic language* for T-coalgebras is an object $\mathcal{L}$ together with a morphism

$$Th_\alpha : A \longrightarrow \mathcal{L}$$

for each coalgebra $\alpha : A \longrightarrow TA$.

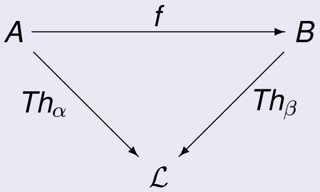
Pointless expressivity I

Expressivity in *Set*

$$s \sim s' \text{ iff } Th_\alpha(s) = Th_\beta(s')$$

From left to right

The following diagram



commutes for every coalgebra morphism f.

Pointless expressivity II

Expressivity in *Set*

$$s \sim s' \text{ iff } Th_{\alpha}(s) = Th_{\beta}(s')$$

From right to left

For every pullback there exists a pair of coalgebra morphism f_1, f_2 such that

$$\begin{array}{ccc} P & \xrightarrow{p_1} & A \\ {\scriptstyle p_2}\downarrow & & \downarrow{\scriptstyle Th_{\alpha}} \\ B & \xrightarrow[Th_{\beta}]{} & \mathcal{L} \end{array} \qquad\qquad \begin{array}{ccc} P & \xrightarrow{p_1} & A \\ {\scriptstyle p_2}\downarrow & & \downarrow{\scriptstyle f_1} \\ B & \xrightarrow[f_2]{} & C \end{array}$$

the diagram on the right commutes.

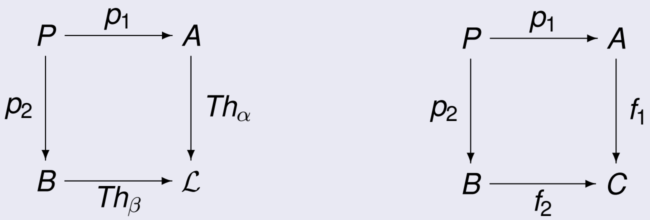
Pointless expressivity II

Expressivity in *Set*

$$s \sim s' \text{ iff } Th_\alpha(s) = Th_\beta(s')$$

From right to left

For every pullback there exists a pair of coalgebra morphism f_1, f_2 such that



the diagram on the right commutes.

From final coalgebras to expressive languages

Theorem

For any functor $T : \mathbb{A} \longrightarrow \mathbb{A}$ over a category with pullbacks; if there exists a final coalgebra ζ, there exists an expressive abstract coalgebraic language.

Proof.

Take $\mathcal{L} = Z$ and $Th_\alpha = f_\alpha$. □

The road to go

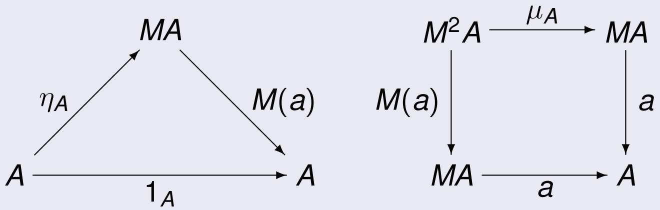
The converse of the previous theorem holds if $\mathbb{A}$ is monadic over *Set*

Outline

Monadic categories: Definition

Definition

An algebra for a monad $M : Set \longrightarrow Set$ is a morphism $a : MA \longrightarrow A$ such the following diagrams



commute.

Our new setting

- We are working with a functor $T : \mathbb{A} \longrightarrow \mathbb{A}$, where $\mathbb{A}$ is monadic over *Set*.
- We have an expressive language $(\mathcal{L}, I)$ for T-coalgebras.
- We will work on the category of Eilenberg-Moore algebras Set^M.

Lifting our construction

Main issues

- The set $Z = \{\Phi \in U\mathcal{L} \mid (\exists \alpha)(\exists s \in U(\alpha))(Th_\alpha(s) = \Phi)\}$ is the carrier of an M-algebra.
- For each T-coalgebra α there exists a morphism

$$f_\alpha : (A, a) \longrightarrow (Z, z).$$

- The function $\zeta : U(Z, z) \longrightarrow UT(Z, z)$ mapping an element $Th_\alpha(s) = \Phi \in \mathcal{L}$ to

$$\zeta(\Phi) = T(f_\alpha)\alpha(s).$$

  is a morphism of M-algebras.
- The morphisms f_α are the only coalgebra morphisms into (Z, z, ζ).

Lifting our construction: We are here!!!

Main issues

- The set $Z = \{\Phi \in U\mathcal{L} \mid (\exists\alpha)(\exists s \in U(\alpha))(Th_\alpha(s) = \Phi)\}$ is the carrier of an M-algebra.
- For each T-coalgebra α there exists a morphism

$$f_\alpha : (A, a) \longrightarrow (Z, z).$$

- The function $\zeta : U(Z, z) \longrightarrow UT(Z, z)$ mapping an element $Th_\alpha(s) = \Phi \in \mathcal{L}$ to

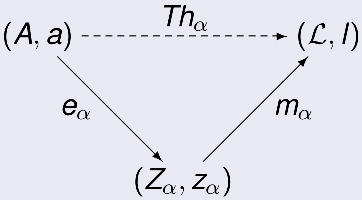
$$\zeta(\Phi) = T(f_\alpha)\alpha(s).$$

  is a morphism of M-algebras.
- The morphisms f_α are the only coalgebra morphisms into (Z, z, ζ).

The set Z is an algebra I

Lemma

For every coalgebra α, the theory arrow $Th_\alpha : (A, a) \longrightarrow (\mathcal{L}, l)$ has a $(RegEpi, Mono)$-factorization



where $Z_\alpha = \{\, Th_\alpha(s) \mid s \in UA \}$.

We are using…

- Every monad over *Set* is regular.
- Regular monads lift $(RegEpi, Mono)$-factorizations.

The set Z is an algebra II

- For each $\Phi \in Z$ choose a coalgebra $(A_\Phi, a_\Phi, \alpha_\Phi)$ and a state $s \in A_\Phi$ such that $Th_\alpha(s) = \Phi$.
- Let $(G, g, \gamma) = \coprod_{\Phi \in Z} \alpha_\Phi$ be the coproduct in $Coalg(T)$.
- By construction the function $Th_\gamma : G \longrightarrow \mathcal{L}$ factors via

$$Z = \{\Phi \in U\mathcal{L} \mid (\exists \alpha)(\exists s \in \alpha)(Th_\alpha(s) = \Phi)\}.$$

- Using the previous lemma we obtain the algebraic structure.

Lifting our construction: We are here!!!

Main issues

- The set $Z = \{\Phi \in U\mathcal{L} \mid (\exists\alpha)(\exists s \in U(\alpha))(Th_\alpha(s) = \Phi)\}$ is the carrier of an M-algebra.
- For each T-coalgebra α there exists a morphism

$$f_\alpha : (A, a) \longrightarrow (Z, z).$$

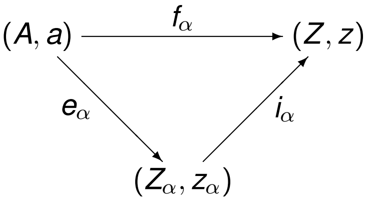
- The function $\zeta : U(Z, z) \longrightarrow UT(Z, z)$ mapping an element $Th_\alpha(s) = \Phi \in \mathcal{L}$ to

$$\zeta(\Phi) = T(f_\alpha)\alpha(s).$$

  is a morphism of M-algebras.
- The morphisms f_α are the only coalgebra morphisms into (Z, z, ζ).

Constructing the final maps

- For each coalgebra α, the inclusion map $i_\alpha : Z_\alpha \longrightarrow Z$ is a morphism of M algebras.
- Define f_α as follows



Remark

For every $s \in A$

$$f_\alpha(s) = Th_\alpha(s)$$

Lifting our construction: We are here!!!

Main issues

- The set $Z = \{\Phi \in U\mathcal{L} \mid (\exists\alpha)(\exists s \in U(\alpha))(Th_\alpha(s) = \Phi)\}$ is the carrier of an M-algebra.
- For each T-coalgebra α there exists a morphism

$$f_\alpha : (A, a) \longrightarrow (Z, z).$$

- The function $\zeta : U(Z, z) \longrightarrow UT(Z, z)$ mapping an element $Th_\alpha(s) = \Phi \in \mathcal{L}$ to

$$\zeta(\Phi) = T(f_\alpha)\alpha(s).$$

  is a morphism of M-algebras.
- The morphisms f_α are the only coalgebra morphisms into (Z, z, ζ).

Defining the structural map I

- The function ζ is well defined because U preserves pullbacks, and $\mathcal{L}$ is expressive.
- For every coalgebra α, the following diagram

$$\begin{array}{ccc} U(A,a) & \xrightarrow{f_\alpha} & U(Z,z) \\ \Big\downarrow{\scriptstyle \alpha} & & \Big\downarrow{\scriptstyle \zeta} \\ UT(A,a) & \xrightarrow[T(f_\alpha)]{} & UT(Z,z) \end{array}$$

commutes in *Set*.

Defining the structural map II

- Consider the coalgebra $(G, g, \gamma) = \coprod_{\Phi \in Z} \alpha_\Phi$.
- We want to fill the following diagram

$$\begin{array}{ccc} (G, g) & \xrightarrow{f_\gamma} & (Z, z) \\ \gamma \downarrow & & \\ T(G, g) & \xrightarrow[T(f_\gamma)]{} & T(Z, z) \end{array}$$

in $\mathbb{A}$.

- The morphism f_γ is a coequalizer in $\mathbb{A}$. Say it coequalizes p and q.
- Now we will show that $T(f_\gamma)\gamma$ also coqequalizes p and q.

Defining the structural map II

- Consider the coalgebra $(G, g, \gamma) = \coprod_{\Phi \in Z} \alpha_\Phi$.
- We want to fill the following diagram

$$\begin{array}{ccc} (G, g) & \xrightarrow{\;f_\gamma\;} & (Z, z) \\ {\scriptstyle \gamma}\downarrow & & \downarrow \\ T(G, g) & \xrightarrow[\;T(f_\gamma)\;]{} & T(Z, z) \end{array}$$

in $\mathbb{A}$.

- The morphism f_γ is a coequalizer in $\mathbb{A}$. Say it coequalizes p and q.
- Now we will show that $T(f_\gamma)\gamma$ also coqequalizes p and q.

Defining The structural map III

- The diagram

$$\begin{array}{ccc} U(G,g) & \xrightarrow{f_\gamma} & U(Z,z) \\ \gamma \downarrow & & \downarrow \zeta \\ UT(G,g) & \xrightarrow[T(f_\gamma)]{} & UT(Z,z) \end{array}$$

  commutes in *Set*.
- Using this we have

$$U(T(f_\gamma)\gamma p) = \zeta U(f_\gamma p) = \zeta U(f_\gamma q) = U(T(f_\gamma)\gamma q)$$

- Since U is faithful we are done.

Lifting our construction: We are here!!!

Main issues

- The set $Z = \{\Phi \in U\mathcal{L} \mid (\exists \alpha)(\exists s \in U(\alpha))(Th_\alpha(s) = \Phi)\}$ is the carrier of an M-algebra.
- For each T-coalgebra α there exists a morphism

$$f_\alpha : (A, a) \longrightarrow (Z, z).$$

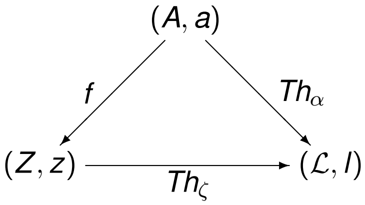
- The function $\zeta : U(Z, z) \longrightarrow UT(Z, z)$ mapping an element $Th_\alpha(s) = \Phi \in \mathcal{L}$ to

$$\zeta(\Phi) = T(f_\alpha)\alpha(s).$$

  is a morphism of M-algebras.
- The morphisms f_α are the only coalgebra morphisms into (Z, z, ζ).

Finish touches

- $Th_\zeta : (Z, z) \longrightarrow (\mathcal{L}, m)$ is a monomorphism (actually it is the inclusion map).
- Every coalgebra morphism $f : \alpha \longrightarrow \zeta$ makes the following diagram



commute.

- The only morphism is f_α.

Main results

Theorem

For any functor $T : \mathbb{A} \longrightarrow \mathbb{A}$ *over a monadic category over* Set*; if there exists an expressive abstract coalgebraic language for* T*, then there exists a final* T*-coalgebra.*

Corollary

The construction still works if $\mathbb{A}$ *is epireflective in a monadic category over* Set*.*

The properties used

The setting

Given a functor

$$U : \mathbb{A} \longrightarrow Set$$

we used...

- U Is faithful.
- $\mathbb{A}$ has pullbacks and coproducts.
- U lifts $(RegEpi, Mono)$-factorizations.
- U lifts the inclusion $i_\alpha : Z_\alpha \longrightarrow Z$ (U lifts factorizations of sinks)

Further work

- To find interesting structured languages.
- Can we provide a more categorical proof?
- What would be the dual of our construction?