Functorial Coalgebraic Logic: The case of many-sorted varieties

Alexander Kurz, Daniela Petrişan

University of Leicester

April 4, 2008

9th International Workshop on Coalgebraic Methods in Computer Science

Overview

- ▶ Motivation
- ▷ Functors with finitary presentations
- ▷ Equational logic for higher-order abstract syntax
- ▷ Modular coalgebraic logic
- ▷ Uniform completeness proofs

Motivation

- Logics for T-coalgebras are suitably described by endofunctors on $\mathcal{A}$

$$\begin{tikzcd}
\mathcal{A} \arrow[loop left, "L"] \arrow[r, bend right] & \mathcal{X} \arrow[l, bend right] \arrow[loop right, "T"]
\end{tikzcd}$$

- Functors having finitary presentation by operations and equations give rise to adequate logics for coalgebras

- Moving to many-sorted varieties is necessary in certain situations

- We are interested in modularity. How can we describe the logics for $T_1 \circ T_2$-coalgebras?

Motivation

- Logics for T-coalgebras are suitably described by endofunctors on $\mathcal{A}$

$$\begin{tikzcd} \mathcal{A} \arrow[loop left, "L"] \arrow[r, bend right] & \mathcal{X} \arrow[l, bend right] \arrow[loop right, "T"] \end{tikzcd}$$

- Functors having finitary presentation by operations and equations give rise to adequate logics for coalgebras

- Moving to many-sorted varieties is necessary in certain situations

- We are interested in modularity. How can we describe the logics for $T_1 \circ T_2$-coalgebras?

Overview

- ▷ Motivation
- ▶ Functors with finitary presentations
- ▷ Equational logic for higher-order abstract syntax
- ▷ Modular coalgebraic logic
- ▷ Uniform completeness proofs

Many-sorted varieties: notations

- S - a set of sorts and Σ -a finitary signature, which can be regarded as a functor:

$$\Sigma : \mathsf{Set}^S \to \mathsf{Set}^S \qquad (\Sigma X)_s = (\coprod_{k \in \omega_f^S} \Sigma_{k,s} \times X^k)_s$$

$E \subseteq \mathsf{Term}_\Sigma(X) \times \mathsf{Term}_\Sigma(X)$ - a set of equations

- a many- sorted variety $\mathcal{A} := \mathsf{Alg}(\Sigma, E)$
- the forgetful functor $U : \mathcal{A} \to \mathsf{Set}^S$ and its left adjoint F.
- a slight generalization of the notion of signature:

$$\Sigma : \mathsf{Set}^{S_1} \to \mathsf{Set}^{S_2} \qquad \Sigma X = (\coprod_{k \in \omega_f^{S_1}} \Sigma_{k,s} \times X^k)_{s \in S_2}$$

Finitary presentations for algebras

- A be a many-sorted algebra in a variety $\mathcal{A}$.

- G is an S-sorted set of generators

- $E = (E_s)_{s \in S}$, $E_s \subset (UFG)_s \times (UFG)_s$ an S sorted set of equations

- A is presented by (G, E) iff A is the coequalizer of the following diagram:

$$\begin{tikzcd} FE \arrow[r, shift left, "\pi_1^\sharp"] \arrow[r, shift right, "\pi_2^\sharp"'] & FG \arrow[r, "q_A"] & A \end{tikzcd}$$

where the maps $\pi_1^\sharp, \pi_2^\sharp$ are induced, via the adjunction, by the projections π_1, π_2 of E on UFG.

How can we define a finitary presentation for $L : \mathcal{A}_1 \to \mathcal{A}_2$?

- LA should be the generated by the elements of A using some operations, encoded in a functor $\Sigma : \mathsf{Set}^{S_1} \longrightarrow \mathsf{Set}^{S_2}$.

- What about the equations? If V is an S_1-sorted set of variables we consider $(E_{V,s})_{s \in S_2}$:

$$E_{V,s} \subseteq (U_2 F_2 \Sigma U_1 F_1 V)_s^2$$

A finitary presentation for $L : \mathcal{A}_1 \to \mathcal{A}_2$ (1)

Definition. A functor $L : \mathcal{A}_1 \to \mathcal{A}_2$ is presented by $\langle \Sigma, E \rangle$, if

(i) for every algebra $A \in \mathcal{A}_1$ the algebra LA is the joint coequalizer taken after all finite sets of S_1-sorted variables V and all valuations $v : V \to U_1 A$.

$$\begin{tikzcd}
F_2 E_V \arrow[r, shift left, "\pi_1^\sharp"] \arrow[r, shift right, "\pi_2^\sharp"'] & F_2 \Sigma U_1 F_1 V \arrow[r, "F_2 \Sigma U_1 v^\sharp"] & F_2 \Sigma U_1 A \arrow[r, "q_A"] & LA
\end{tikzcd}$$

and ...

A finitary presentation for $L : \mathcal{A}_1 \to \mathcal{A}_2$ (2)

(ii) for all morphisms $f : A \to B$ the diagram commutes:

$$\begin{tikzcd}
F_2 \Sigma U_1 A \arrow[r, "q_A"] \arrow[d, "F_2 \Sigma U_1 f"] & LA \arrow[d, "Lf"] \\
F_2 \Sigma U_1 B \arrow[r, "q_B"] & LB
\end{tikzcd}$$

$\mathsf{Alg}(L)$ as an equational class

Theorem. *Let* $\mathcal{A} = \mathsf{Alg}(\Sigma_{\mathcal{A}}, E_{\mathcal{A}})$ *be an* S*-sorted variety and let* $L : \mathcal{A} \to \mathcal{A}$ *be a functor presented by operations* Σ_L *and equations* E_L*. Then* $\mathsf{Alg}(L) \cong \mathsf{Alg}(\Sigma_{\mathcal{A}} + \Sigma_L, E_{\mathcal{A}} + E_L)$*.*

Proof.(Sketch) Define $H : \mathsf{Alg}(L) \to \mathsf{Alg}(\Sigma_{\mathcal{A}} + \Sigma_L, E_{\mathcal{A}} + E_L)$ by

$$\alpha : LA \to A \qquad \mapsto \qquad HA$$

- the underlying set of HA is defined to be UA.

- the interpretation of the operation symbols of $\Sigma_{\mathcal{A}}$ is the same as in the algebra A;

- the interpretation of the operation symbols of Σ_L is given by

$$F\Sigma_L UA \xrightarrow{q_A} LA \xrightarrow{\alpha} A$$

- HA satisfies the equations E_A and E_L

Alg(L) as an equational class

Conversely we define J : Alg$(\Sigma_{\mathcal{A}} + \Sigma_L, E_{\mathcal{A}} + E_L) \to$ Alg(L)

- Consider the map $\rho_A : \Sigma_L UA \to UA$ defined by:

$$(\sigma_{(s_1 \ldots s_n;s)}, x_{i_1}, \ldots, x_{i_n}) \mapsto \sigma_{(s_1 \ldots s_n;s)}(x_{i_1}, \ldots, x_{i_n})$$

$$\begin{tikzcd}
 & & F\Sigma_L UA \arrow[r, "q_A"] \arrow[rr, bend left, "\rho_A^\sharp"] \arrow[dd, "F\Sigma Uf"] & LA \arrow[r, dashed, "\alpha_A"] \arrow[dd, "Lf"] & A \arrow[dd, "f"] \\
FE_L \arrow[r, shift left, "\pi_1^\sharp"] \arrow[r, shift right, "\pi_2^\sharp"'] & F\Sigma_L UFV \arrow[ur, "F\Sigma Uv_1^\sharp"] \arrow[dr, "F\Sigma Uv_2^\sharp"] & & & \\
 & & F\Sigma_L UB \arrow[r, "q_B"] \arrow[rr, bend right, "\rho_B^\sharp"] & LB \arrow[r, dashed, "\alpha_B"] & B
\end{tikzcd}$$

- We define $J(A)$ to be α_A.

A characterization theorem

Theorem. *Let* S_1, S_2 *be sets of sorts,* $\mathcal{A}_1$ *be an* S_1*-sorted variety and* $\mathcal{A}_2$ *be an* S_2*-sorted variety. For a functor* $L : \mathcal{A}_1 \to \mathcal{A}_2$ *the following conditions are equivalent:*

(i) L has a finitary presentation by operations and equations;

(ii) L preserves sifted colimits.

Overview

▷ Motivation

▷ Functors with finitary presentations

▶ Equational logic for higher-order abstract syntax

▷ Modular coalgebraic logic

▷ Uniform completeness proofs

Syntax with variable binders

- Cannot be captured as an initial algebra for functors on Set.

- But we can do it if we move to functors on a presheaf category (Fiore, Plotkin and Turi).

- In particular, the lambda terms up to α-equivalence form an initial algebra for a functor.

Canonical representatives for λ-terms up to α-equivalence

- The method of De Bruijn levels.

$$\frac{1 \leq i \leq n}{x_1, \ldots, x_n \vdash x_i} \quad , \quad \frac{x_1, \ldots, x_n, x_{n+1} \vdash t}{x_1, \ldots, x_n \vdash \lambda x_{n+1}.t}$$

$$\frac{x_1, \ldots, x_n \vdash t_1 \quad x_1, \ldots, x_n \vdash t_2}{x_1, \ldots, x_n \vdash t_1 t_2}$$

- Contexts stratify λ-terms up to α-equivalence and can be regarded as a set of sorts.

- We only have to find the right notion to encompass contexts and the operations allowed on them.

The functor-category $\mathsf{Set}^{\mathbb{F}}$

- $\mathbb{F}$ is the full subcategory of Set with objects $\underline{n} = \{1, \ldots, n\}$ and $\underline{0} = \emptyset$.

- $\mathcal{A} = \mathsf{Set}^{\mathbb{F}}$ is a many-sorted unary variety:
    - the sorts: objects of $\mathbb{F}$
    - operation symbols: morphisms of $\mathbb{F}$
    - equations: $h(x) = f(g(x))$ (when equality holds in $\mathbb{F}$) or $id_n(x) = x$

$$\begin{tikzcd}
\mathcal{A} \arrow[d, bend left, "U"] \\
\mathsf{Set}^{\mathbb{N}} \arrow[u, bend left, "F"] \arrow[u, phantom, "\dashv"]
\end{tikzcd}$$

λ-terms as presheafs

- The equivalence classes of λ-terms over a countable set of variables $V = \{x_1, x_2, \ldots\}$ form a presheaf in $\mathsf{Set}^{\mathbb{F}}$: ΛV_α.

  ◦ $\Lambda V_\alpha(\underline{n})$ is the set of equivalence classes of λ-terms with the free variables contained in the set $\{x_1, \ldots, x_n\}$.

  ◦ For any morphism $\rho : \underline{n} \to \underline{m}$, $\Lambda V_\alpha(\rho)$ substitutes the free variables x_i with $x_{\rho(i)}$.

- For an arbitrary presheaf of variables V, the λ-terms over V form a presheaf in $\mathsf{Set}^{\mathbb{F}}$.

A suitable functor to describe the presheaf of λ-terms

- A coproduct structure on $\mathbb{F}$

$$\begin{tikzcd} & \underline{1} \arrow[d, "new"] \\ \underline{n} \arrow[r, "i"] & \underline{n+1} \end{tikzcd}$$

where i is the inclusion and $new(1) = n+1$.

- The type constructor $\delta : \mathcal{A} \to \mathcal{A}$ for context extension:

$$\delta(A)(\rho) = A(\rho + id_1) \qquad \forall A \in \mathcal{A} \quad \forall \rho \in \mathbb{F}^{Morph}$$

- Let $L : \mathcal{A} \to \mathcal{A}$ be the functor given by

$$LX = \delta X + X \times X$$

The algebraic structure of ΛV_α

Theorem.(Fiore, Plotkin, Turi) ΛV_α *is the free* L*-algebra on the presheaf of variables* V

- But $\mathsf{Alg}(L)$ is an equational class, and a presentation can be obtained from:
    - an equational presentation of $\mathcal{A}$ and
    - a finitary presentation of L.

An equational presentation for $\mathcal{A}$: the signature

We consider the following operation symbols:

$$\Sigma_{\mathcal{A}} = \{\sigma_n^{(i)} | \ \ 1 < n, \ 1 \leq i < n\} \cup \{w_n \ \ | \ \ n \geq 0\} \cup \{c_n \ \ | \ \ n > 0\}$$

with the intended interpretation:

$\sigma_n^{(i)}$ - the transposition $(i, i+1)$ of the set $\underline{n}$,

c_n - a contraction $c_n : \underline{n+1} \to \underline{n}$, given by

$$c_n(i) = i \ \ \forall i \leq n, \quad c_n(n+1) = n$$

w_n - the inclusion of $\underline{n}$ into $\underline{n+1}$.

An equational presentation for $\mathcal{A}$: the equations $E_{\mathcal{A}}$ (1)

-the equations coming from the presentation of the symmetric group:

$$\begin{array}{ll} (\sigma_n^{(i)})^2(x) = id_n(x) & 1 \leq i < n \\ \sigma_n^{(i)}\sigma_n^{(j)}(x) = \sigma_n^{(j)}\sigma_n^{(i)}(x) & j \neq i \pm 1; 1 \leq i, j < n \\ (\sigma_n^{(i)}\sigma_n^{(i+1)})^3(x) = id_n(x) & 1 \leq i < n-1 \end{array} \qquad (\mathrm{E}_1)$$

-the equations coming form the presentation of the monoid of functions on $\underline{n}$ (Aizenstat):

$$\begin{array}{l} A\sigma_n^{(1)} = \sigma_n^{(3)}A\sigma_n^{(3)} = (3,4,\ldots,n)A(3,4,\ldots,n) = [(1,n)A]^2 = A \\ (\sigma_n^{(2)}A)^2 = A\sigma_n^{(2)}A = (A\sigma_n^{(2)})^2 \\ (\sigma_n^{(2)}(1,n)A)^2 = (A\sigma_n^{(2)}(1,n))^2 \end{array} \qquad (\mathrm{E}_2)$$

An equational presentation for $\mathcal{A}$: the equations $E_{\mathcal{A}}$ (2)

-and some extra equations:

$$
\begin{array}{lrr}
c_n\sigma^{(n)}_{n+1}(y) = c_n(y) & & (\mathsf{E}_3) \\
c_n w_n(x) = id_n(x) & & (\mathsf{E}_4) \\
\sigma^{(i)}_{n+1} w_n(x) = w_n\sigma^{(i)}_n(x) & 1 \le i < n & (\mathsf{E}_5) \\
\sigma^{(n+1)}_{n+2} w_{n+1} w_n(x) = w_{n+1} w_n(x) & & (\mathsf{E}_6) \\
\sigma^{(i)}_n c_n(y) = c_n\sigma^{(i)}_{n+1}(y) & i < n-1 & (\mathsf{E}_7) \\
c_n\sigma^{(n-1)}_{n+1}\sigma^{(n)}_{n+1} w_n(x) = \sigma^{(n-1)}_n w_{n-1} c_{n-1}(x) & & (\mathsf{E}_8) \\
c_1 c_2\sigma^{(1)}_3 = c_1 c_2 & & (\mathsf{E}_9)
\end{array}
$$

A finitary presentation for L

- The operation symbols: lam_n, app_n for each $n \in \mathbb{N}$; (semantically they correspond to λ-abstraction and application).

- The respective signature functor $\Sigma_L : \mathsf{Set}^{\mathbb{N}} \longrightarrow \mathsf{Set}^{\mathbb{N}}$ is given by:

$$(\Sigma_L X)_m = \{lam_{m+1}\} \times X_{m+1} + \{app_m\} \times X_m \times X_m$$

- For any presheaf $V \in \mathcal{A}$ let $\rho_V : \Sigma UV \longrightarrow ULV$ be the map defined by

$$(lam_{n+1}, t) \mapsto t \qquad \forall t \in V(n+1) = (\delta V)(n)$$

$$(app_n, t_1, t_2) \mapsto (t_1, t_2) \qquad \forall t_1, t_2 \in V(n)$$

A finitary presentation for L - the equations

- The equations E_L should correspond to the kernel pair of the adjoint transpose $\rho_V^\sharp : F\Sigma UV \to LV$.

$$\begin{array}{ll}
\sigma_n^{(i)}(lam_{n+1}, t) = (lam_{n+1}, \sigma_{n+1}^{(i)} t) & [t] \\
w_n(lam_{n+1}, t) = (lam_{n+2}, \sigma_{n+2}^{(n+1)} w_{n+1} t) & [t] \\
c_n(lam_{n+2}, t') = (lam_{n+1}, \sigma_{n+1}^{(n)} c_{n+1} \sigma_{n+2}^{(n)} \sigma_{n+2}^{(n+1)} t') & [t'] \\
\sigma_n^{(i)}(app_n, t_1, t_2) = (app_n, \sigma_n^{(i)} t_1, \sigma_n^{(i)} t_2) & [t_1, t_2] \\
w_n(app_n, t_1, t_2) = (app_n, w_n t_1, w_n t_2) & [t_1, t_2] \\
c_n(app_n, t_1, t_2) = (app_n, c_n t_1, c_n t_2) & [t_1, t_2]
\end{array}$$

Representing different implementations of λ-terms

- If V is the presheaf defined by $V(\rho) = \rho$ for all morphisms ρ in $\mathbb{F}$, the free L-algebra over V gives an implementation of $\lambda-$terms by the De Bruijn levels method.

- How can we obtain different implementations for λ-terms?

- One possible approach: equip $\mathbb{F}$ with different coproduct structures!

- But this implies working with a different functor than L.

- Let's keep L and use different presheaves of variables!

Representing different implementations of λ-terms

- If V is the presheaf defined by $V(\rho) = \rho$ for all morphisms ρ in $\mathbb{F}$, the free L-algebra over V gives an implementation of $\lambda-$terms by the De Bruijn levels method.

- How can we obtain different implementations for λ-terms?

- One possible approach: equip $\mathbb{F}$ with different coproduct structures!

- But this implies working with a different functor than L.

- Let's keep L and use different presheaves of variables!

Representing different implementations of λ-terms

If W is the presheaf of variables defined explicitly by

$$W(n) = n \quad W(c_n)(1) = 1 \quad W(c_n)(i) = i - 1; i > 1$$

$$W(w_n)(i) = i + 1$$

$$W(\sigma_n^{(i)}) = \sigma_n^{(n-i)}$$

we obtain the presheaf ΛW_α of $\lambda-$terms implemented by the De Bruijn indices method.

Overview

- ▷ Motivation
- ▷ Functors with finitary presentations
- ▷ Equational logic for higher-order abstract syntax
- ▶ Modular coalgebraic logic
- ▷ Uniform completeness proofs

Combining syntax and proof systems of two coalgebraic logics

• logics for T_1-coalgebras	L_1	$\langle \Sigma_1, E_1 \rangle$
• logics for T_2-coalgebras	L_2	$\langle \Sigma_2, E_2 \rangle$
• Logics for $T_2 \circ T_1$-coalgebras	$L = L_2 \circ L_1$	?

- A finitary presentation for $L = L_2 \circ L_1$ certainly exists, but can it be obtained in a modular way?

Combining syntax and proof systems of two coalgebraic logics

• logics for T_1-coalgebras	L_1	$\langle \Sigma_1, E_1 \rangle$
• logics for T_2-coalgebras	L_2	$\langle \Sigma_2, E_2 \rangle$
• Logics for $T_2 \circ T_1$-coalgebras	$L = L_2 \circ L_1$	?

- A finitary presentation for $L = L_2 \circ L_1$ certainly exists, but can it be obtained in a modular way?

The two-sorted composition of functors

- A coalgebra $X \rightarrow T_2 \circ T_1(X)$ first goes to an intermediate state $T_1(X)$ and then to the proper state $T_2(T_1(X))$.

Definition *Given two functors* $L_1 : \mathcal{A}_s \rightarrow \mathcal{A}_i$ *and* $L_2 : \mathcal{A}_i \rightarrow \mathcal{A}_s$ *between any two categories, the two-sorted composition of* L_1 *with* L_2 *is the functor* $\bar{L} : \mathcal{A}_i \times \mathcal{A}_s \rightarrow \mathcal{A}_i \times \mathcal{A}_s$ *mapping* $A = (A_{\mathsf{i}}, A_{\mathsf{s}})$ *to* $(\bar{L}A)_{\mathsf{s}} = L_2(A_{\mathsf{i}})$ *and* $(\bar{L}A)_{\mathsf{i}} = L_1 A_{\mathsf{s}}$.

Proposition *Consider categories* $\mathcal{A}_{\mathsf{i}}, \mathcal{A}_{\mathsf{s}}$ *which are lfp and two finitary functors* $L_1 : \mathcal{A}_s \rightarrow \mathcal{A}_i$ *and* $L_2 : \mathcal{A}_i \rightarrow \mathcal{A}_s$. *Let* $\bar{L}$ *be the two-sorted composition of* L_1 *with* L_2. *Then the* s*-component of the initial* $\bar{L}$*-algebra is the initial* $L_2 L_1$*-algebra.*

The two-sorted composition of functors

- A $X \to T_2 \circ T_1(X)$ coalgebra first goes to an intermediate state $T_1(X)$ and then to the proper state $T_2(T_1(X))$.

Definition *Given two functors* $L_1 : \mathcal{A}_s \to \mathcal{A}_i$ *and* $L_2 : \mathcal{A}_i \to \mathcal{A}_s$ *between any two categories, the two-sorted composition of* L_1 *with* L_2 *is the functor* $\bar{L} : \mathcal{A}_i \times \mathcal{A}_s \to \mathcal{A}_i \times \mathcal{A}_s$ *mapping* $A = (A_\mathsf{i}, A_\mathsf{s})$ *to* $(\bar{L}A)_\mathsf{s} = L_2(A_\mathsf{i})$ *and* $(\bar{L}A)_\mathsf{i} = L_1 A_\mathsf{s}$*.*

Proposition *Consider categories* $\mathcal{A}_\mathsf{i}, \mathcal{A}_\mathsf{s}$ *which are lfp and two finitary functors* $L_1 : \mathcal{A}_s \to \mathcal{A}_i$ *and* $L_2 : \mathcal{A}_i \to \mathcal{A}_s$*. Let* $\bar{L}$ *be the two-sorted composition of* L_1 *with* L_2*. Then the* s*-component of the initial* $\bar{L}$*-algebra is the initial* $L_2 L_1$*-algebra.*

A presentation for the two-sorted composition of two functors

- Consider $\mathcal{A}_\mathsf{i}, \mathcal{A}_\mathsf{s}$ - many-sorted varieties $L_1 : \mathcal{A}_s \to \mathcal{A}_i$ presented by $\langle \Sigma_1, E_1 \rangle$ and $L_2 : \mathcal{A}_i \to \mathcal{A}_s$ presented by:$\langle \Sigma_2, E_2 \rangle$

- Define $\bar{\Sigma} : \mathsf{Set}^{S_1} \times \mathsf{Set}^{S_2} \longrightarrow \mathsf{Set}^{S_1} \times \mathsf{Set}^{S_2}$ by

$$(\bar{\Sigma} X)_\mathsf{s} = \Sigma_2 X_\mathsf{i}$$

$$(\bar{\Sigma} X)_\mathsf{i} = \Sigma_1 X_\mathsf{s}$$

where $X = (X_\mathsf{i}, X_\mathsf{s})$ denotes and element of $\mathsf{Set}^{S_1} \times \mathsf{Set}^{S_2}$.

- Equations are given by $\bar{E}_\mathsf{s} = E_2$, $\bar{E}_\mathsf{i} = E_1$.

Theorem *Then $\langle \bar{\Sigma}, \bar{E} \rangle$ is a presentation of the two-sorted composition $\bar{L}$ of L_1 with L_2.*

An example: $\mathcal{A}_i = \mathcal{A}_s = \mathsf{BA}$

- $\vdash_{\mathsf{i}} \psi$ and $\vdash_{\mathsf{s}} \phi$ to assert that ψ, ϕ are formulas of sort i, s

- The formulas of both sorts are closed under Boolean operations and for all n-ary operation symbols σ_i in Σ_i, formulas are closed under:

$$\frac{\vdash_{\mathsf{i}} \psi_1, \ldots, \vdash_{\mathsf{i}} \psi_n}{\vdash_{\mathsf{s}} \sigma_2(\psi_1, \ldots, \psi_n)} \qquad \frac{\vdash_{\mathsf{s}} \phi_1, \ldots, \vdash_{\mathsf{s}} \phi_n}{\vdash_{\mathsf{i}} \sigma_1(\phi_1, \ldots, \phi_n)}$$

An example: $\mathcal{A}_i = \mathcal{A}_s = \mathsf{BA}$

- The axioms are given by equations E_1, E_2, sortwise.

- The rules of the calculus are those of equational logic. The only rules that make the two sorts interact are the congruence rules:

$$\frac{\vdash_{\mathsf{i}} \psi_1 = \psi'_1, \ldots, \vdash_{\mathsf{i}} \psi_n = \psi'_2}{\vdash_{\mathsf{s}} \sigma_2(\psi_1, \ldots, \psi_n) = \sigma_2(\psi'_1, \ldots, \psi'_n)} \qquad \frac{\vdash_{\mathsf{s}} \phi_1 = \phi'_1, \ldots, \vdash_{\mathsf{s}} \phi_n = \phi'_n}{\vdash_{\mathsf{i}} \sigma_1(\phi_1, \ldots, \phi_n) = \sigma_1(\phi'_1, \ldots, \phi'_n)}$$

Logics for the sum of two functors

- the logic of T $\qquad\qquad$ L
- the logic of $T+T$ $\qquad\qquad$ $\mathsf{BA} \xrightarrow{\langle L,L\rangle} \mathsf{BA}\times\mathsf{BA} \xrightarrow{L_+} \mathsf{BA}$
- Here $L_+ : \mathsf{BA}\times\mathsf{BA}\to\mathsf{BA}$ captures the logic of $+ : \mathsf{Set}\times\mathsf{Set}\to\mathsf{Set}$ and has a finitary presentation:

Overview

▷ Motivation

▷ Functors with finitary presentations

▷ Equational logic for higher-order abstract syntax

▷ Modular coalgebraic logic

▶ Uniform completeness proofs

Complete logics for set-functors

Definition of L

$$\begin{tikzcd}
\mathcal{A} \arrow[loop left, "L"] \arrow[r, bend right, "S"'] & \mathcal{X} \arrow[l, bend right, "P"'] \arrow[loop right, "T"] \\
\mathcal{A}_0 \arrow[u, "I"] &
\end{tikzcd}$$

where $\mathcal{A}$ is lfp with a small subcategory $\mathcal{A}_0$ of finitely presentable objects.

- We then define L on $\mathcal{A}_0$ as

$$LA = PTSA$$

and extend L continuously from $\mathcal{A}_0$ to $\mathcal{A}$.

- L preserves filtered colimits, whereas PTS need not to do so.

Complete logics for set-functors

Definition of $\delta : LP \to PT$.

$$\begin{tikzcd}
PX & LPX \arrow[r, "\delta_X"] & PTX \\
A_i \arrow[u, "c_i"] & LA_i \arrow[u, "Lc_i"] \arrow[r, "\cong"] & PTSA_i \arrow[u, "PTc_i^\sharp"']
\end{tikzcd}$$

Lemma δ_X as defined above is injective.

Theorem. *The logic given by L as defined above is complete for T-coalgebras.*

Complete logics for set-functors

Definition of $\delta : LP \to PT$.

$$\begin{tikzcd}
PX & LPX \arrow[r, "\delta_X"] & PTX \\
A_i \arrow[u, "c_i"] & LA_i \arrow[u, "Lc_i"] \arrow[r, "\cong"] & PTSA_i \arrow[u, "PTc_i^\sharp"']
\end{tikzcd}$$

Lemma δ_X as defined above is injective.

Theorem. *The logic given by L as defined above is complete for T-coalgebras.*

Complete logics for set-functors

Definition of $\delta : LP \to PT$.

$$\begin{tikzcd}
PX & LPX \arrow[r, "\delta_X"] & PTX \\
A_i \arrow[u, "c_i"] & LA_i \arrow[u, "Lc_i"] \arrow[r, "\cong"] & PTSA_i \arrow[u, "PTc_i^\sharp"]
\end{tikzcd}$$

Lemma δ_X as defined above is injective.

Theorem. *The logic given by L as defined above is complete for T-coalgebras.*

Thank you!