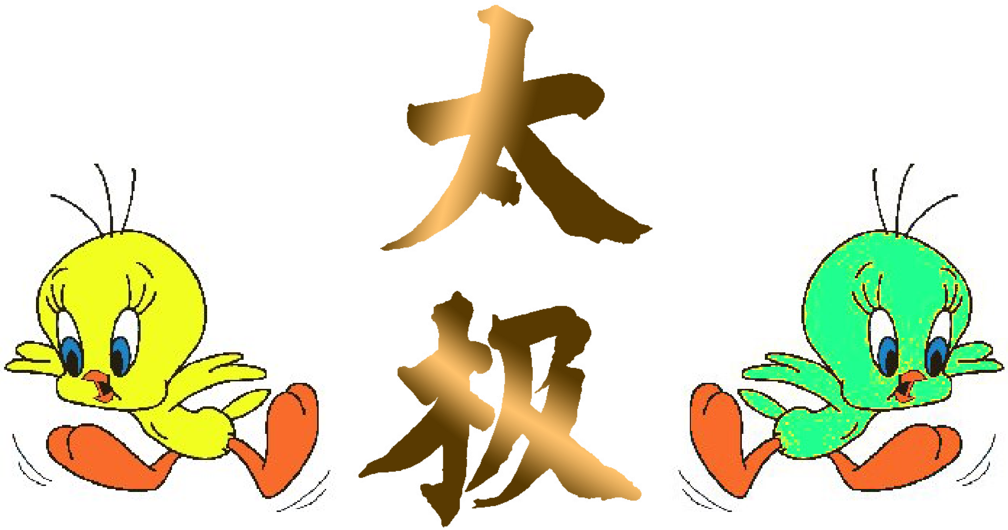
Dialgebraic Specification and Modeling

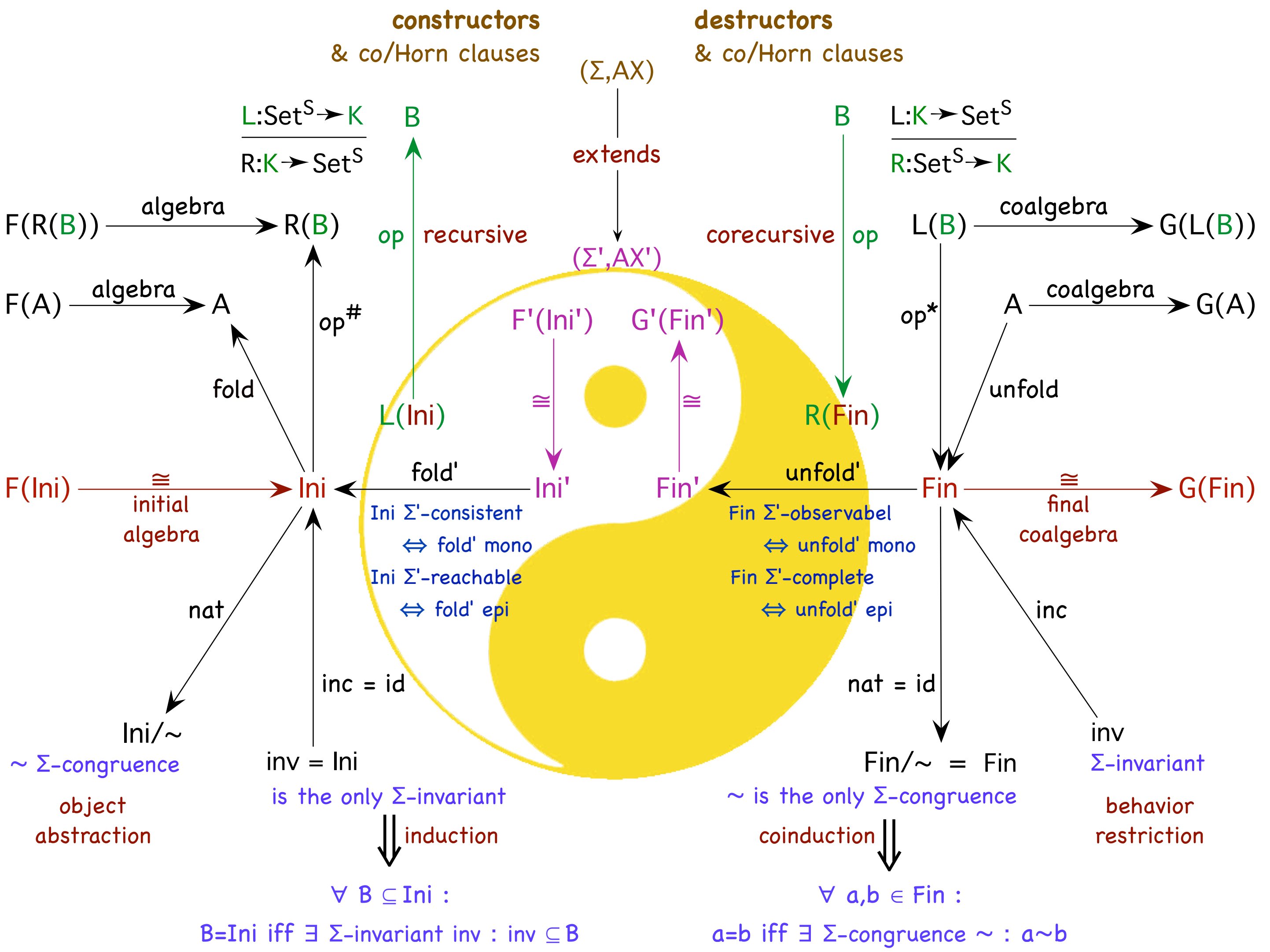
Peter Padawitz
TU Dortmund

September 12, 2011



Contents

The Tai Chi of algebraic modeling



Preliminaries

Set denotes the category of sets with functions as morphisms.

Let I be a set of indices and for all $i \in I$, A_i be a set.

$\prod_{s \in I} A_i$ denotes the product of all A_i.

For all $n > 1$, $A_1 \times \cdots \times A_n = \prod_{i=1}^n A_i$.

$\coprod_{s \in I} A_i$ denotes the coproduct (= disjoint union) of all A_i.

For all $n > 1$, $A_1 + \cdots + A_n = \coprod_{i=1}^n A_i$.

For all $i \in I$, $\pi_i : \prod_{s \in I} A_i \to A_i$ denotes the i-th projection:
For all $a = (a_i)_{i \in I} \in \prod_{s \in I} A_i$, $\pi_i(a) = a_i$.

For all $i \in I$, $\iota_i : A_i \to \coprod_{s \in I} A_i$ denotes the i-th injection:
For all $i \in I$ and $a \in A_i$, $\iota_i(a) = (a, i)$.

Given functions $f_i : A \to A_i$ for all $i \in I$, $\langle f_i \rangle_{i \in I} : A \to \prod_{s \in I} A_i$ denotes the product extension of $\{f_i\}_{i \in I}$: For all $a \in A$, $\langle f_i \rangle_{i \in I}(a) = (f_i(a))_{i \in I}$.

$\prod_{s \in I} f_i = \langle f_i \circ \pi_i \rangle$ and for all $n > 1$, $f_1 \times \cdots \times f_n = \prod_{i=1}^n f_i$.

Given functions $g_i : A_i \longrightarrow A$ for all $i \in I$, $[g_i]_{i\in I} : \coprod_{s\in I} A_i \rightarrow A$ denotes the coproduct extension of $\{f_i\}_{i\in I}$: For all $i \in I$ and $a \in A_i$, $[g_i](a,i) = g_i(a)$.

$\coprod_{s\in I} g_i = [\iota_i \circ g_i]$ and for all $n > 1$, $g_1 + \cdots + g_n = \coprod_{i=1}^{n} g_i$.

For all $a \in \prod_{i\in I} A_i$ and $i, k \in I$, $\pi_i(a[b/k]) =_{def} \begin{cases} b & \text{if } i = k, \\ \pi_i(a) & \text{otherwise.} \end{cases}$

1 denotes the singleton $\{*\}$.

2 denotes the two-element set $\{0,1\}$. The elements of 2 are regarded as truth values.

Let A be a set.

The function $id_A : A \to A$, defined by $id_A = \lambda a.a$, is the **identity** on A.

The relation $\Delta_A \subseteq A \times A$, defined by $\Delta_A = \{(a, a) \mid a \in A\}$ is the **diagonal** of A.

$A^* = \{a \in A^n \mid n \in \mathbb{N}\}$ is the set of **finite words** or **lists** of elements of A.

$\mathcal{B}_{fin}(A) = \{f : A \to \mathbb{N} \mid |supp(f)| < \omega\}$ is the set of **finite bags** or **multisets** of elements of A where $supp(f) = \{a \in A \mid f(a) \neq 0\}$.

$\mathcal{P}_{fin}(A) = \{f : A \to 2 \mid |supp(f)| < \omega\}$ is the set of **finite sets** of elements of A.

$A^{\mathbb{N}}$ is the set of **infinite words** or **lists** of elements of A.

$A^\infty = A^* \cup A^{\mathbb{N}}$ denotes the set of finite or infinite words of elements of A.

CPOs, lattices and fixpoints

Let A be a set and R be a binary relation on A such that R is transitive, i.e., for all $a, b, c \in A$, aRb and bRc implies aRc, and antisymmetric, i.e., for all $a, b \in A$, aRb and bRa implies $a = b$.

R is a **partial order** and A is a **partially ordered set** or **poset** if R is reflexive, i.e., for all $a \in A$, aRa. R is a **total order** and A is a **chain** if for all $a, b \in A$, aRb or bRa. If, in addition, R is irreflexive, i.e., for all $a \in A$, $\neg aRa$, then R is a **strict total order**.

R is **well-founded** if each nonempty subset of A has a minimal element w.r.t. R. If, in addition, R is a strict total order, then R is a **well-order** and, consequently, each nonempty subset of A has a *least* element w.r.t. R.

Let A be a poset with partial order $\leq$, $\geq = \leq^{-1}$ and λ be an ordinal number.

$B = \{a_i \mid i < \lambda\} \subseteq A$ is a **λ-chain** of A if for all ordinals $i < j < \lambda$, $a_i \leq a_j$.

$B = \{a_i \mid i < \lambda\} \subseteq A$ is a **λ-cochain** of A if for all ordinals $i < j < \lambda$, $a_i \geq a_j$.

Remember that an ordinal number is either

- 0 or
- a successor ordinal $n + 1 = n \cup \{n\}$ for some ordinal n,
- a limit ordinal, i.e., an infinite set $\{0, 1, 2, 3, 4, \dots\}$ of ordinals.

A is **λ-complete** or a **λ-CPO** if A has a least element $\bot$ w.r.t. $\leq$ and for each λ-chain B of A, A contains the supremum $\sqcup B$ of B.

A is **λ-cocomplete** or a **λ-coCPO** if A has a greatest element $\top$ w.r.t. $\leq$ and for each λ-cochain B of A, A contains the infimum $\sqcap B$ of B.

Note that $\geq$ is a partial order iff $\leq$ is so, but cocompleteness w.r.t. $\leq$ usually does not agree with completeness w.r.t. $\geq$.

A product of n CPOs is also a CPO. Partial order, least element and suprema are defined componentwise.

The set of functions from a set A to a CPO B is a CPO. The partial order is defined argumentwise: For all $f, g : A \to B$,

$$f \leq g \quad \Leftrightarrow_{def} \quad \forall\, a \in A : f(a) \leq g(a). \tag{1}$$

The least element of $A \to B$ is given by $\Omega = \lambda x.\bot$. Suprema are also defined argumentwise: For all λ-chains $\{f_i : A \to B\}_{i \in \mathbb{N}}$ and $a \in A$,

$$(\sqcup_{i \in \mathbb{N}} f_i)(a) \quad =_{def} \quad \sqcup_{i \in \mathbb{N}}\, f_i(a). \tag{2}$$

A is **directed** if each finite subset of A has a least upper bound w.r.t. R.

Proposition DIR ([43], Cor. 1) Let A be λ-CPO with partial order $\leq$. For all directed subsets B of A with $|B| \leq \lambda$, A contains the supremum $\sqcup B$ of B.

Proof. We show the conjecture only for $\lambda = \omega$ and refer to the proof of [43], Thm. 1, for the generalization to arbitrary ordinal numbers.

Let B be a countable directed subset of A. If B is a chain, then $\sqcup B$ exists because A is ω-complete. Otherwise B is infinite: If B were finite, B would contain two different maximal elements w.r.t. R, which contradicts the directedness of B.

Since B is infinite, there is a bijection $f : \mathbb{N} \longrightarrow B$. We define subsets B_i, $i \in \mathbb{N}$, of B inductively as follows: $B_0 = \{f(0)\}$ and $B_{i+1} = B_i \cup \{f(i), b_i\}$ where $i = min(f^{-1}(B \setminus B_i))$ and b_i is an upper bound of $f(i)$ and (all elements of) B_i. b_i exists because B is directed and $B_i \cup \{f(i)\}$ is a finite subset of B.

For all $i \in \mathbb{N}$, B_i is finite and directed and thus a (countable) chain. Since A is ω-complete, B_i contains the supremum $\sqcup B_i$ of B_i. Since $B_i \subseteq B_{i+1}$, $\{\sqcup B_i \mid i \in \mathbb{N}\}$ is also a countable chain and thus has a supremum c in A. c is the supremum of $C = \cup_{i \in \mathbb{N}} B_i$: For all $i \in \mathbb{N}$ and $b \in B_i$, $b \leq \sqcup B_i \leq c$. Hence c is an upper bound of C. Let d be an upper bound of C. Then for all $i \in \mathbb{N}$, $\sqcup B_i \leq d$ and thus $c \leq d$.

Of course, $\cup_{i \in \mathbb{N}} B_i \subseteq B$. Conversely, let $b \in B$. Since for all $i \in \mathbb{N}$, $|B_i| > i$, there is $k \in \mathbb{N}$ with $b \in B_k$. Hence $B = C$ and thus $c = \sqcup B$. ❏

Let A, B be posets.

$f : A \longrightarrow B$ is **monotone** if for all $a, b \in A$, $a \leq b$ implies $f(a) \leq f(b)$.

Let A, B be λ-CPOs.

$f : A \to B$ is **λ-continuous** if for all λ-chains B of A,

$$f(\sqcup B) = \sqcup\{f(b) \mid b \in B\}.$$

$f : A \to B$ is **λ-cocontinuous** if for all λ-cochains B of A,

$$f(\sqcap B) = \sqcap\{f(b) \mid b \in B\}.$$

If f is λ-co/continuous, then f is monotone.
If f is monotone, then f is λ-continuous iff for all λ-chains B of A, $f(\sqcup B) \leq \sqcup\{f(b) \mid b \in B\}$.
If f is monotone, then f is λ-cocontinuous iff for all λ-cochains B of A, $\sqcap\{f(b) \mid b \in B\} \leq f(\sqcap B)$.
If f is monotone and all λ-co/chains of A are finite, then f is λ-co/continuous.

Given λ-CPOs A and B, $A \to_c B$ denotes the set of λ-continuous functions from A to B.
Since Ω and suprema of λ-chains of λ-continuous functions are λ-continuous, $A \to_c B$ is a λ-CPO.

Kleene's Fixpoint Theorem [37] (also known as Kleene's first recursion theorem)

(1) Let A be an ω-CPO and $f : A \to A$ be ω-continuous.

$\mathit{lfp}(f) = \sqcup_{n \in \mathbb{N}} f^n(\bot)$ is the least fixpoint of f.

(2) Let A be an ω-coCPO and $f : A \to A$ be ω-cocontinuous.

$\mathit{gfp}(f) = \sqcap_{n \in \mathbb{N}} f^n(\top)$ is the greatest fixpoint of f.

Proof.

(1) Since f is ω-continuous, f is monotone and thus $\bot \leq f(\bot) \leq f^2(\bot) \leq \ldots$ is an ω-chain. Since $f(\sqcup_{n \in \mathbb{N}} f^n(\bot)) = \sqcup_{n \in \mathbb{N}} f^{n+1}(\bot) = \sqcup_{n \in \mathbb{N}} f^n(\bot)$, $\mathit{lfp}(f)$ is a fixpoint of f.

Let a be a fixpoint of f. We show $f^n(\bot) \leq a$ for all $n \in \mathbb{N}$ by induction on n: $f^0(\bot) = \bot \leq a$. If $f^n(\bot) \leq a$, then $f^{n+1}(\bot) \leq f(a) = a$ because f is monotone. Hence $\mathit{lfp}(f) \leq a$, i.e., $\mathit{lfp}(f)$ is the least fixpoint of f.

(2) Analogously. ❏

A poset A is a **complete lattice** if each subset B of A has a supremum $\sqcup B$ and an infimum $\sqcap B$ in A.

$\bot = \sqcup\emptyset$ is the least element and $\top = \sqcap\emptyset$ is the greatest element of A.

Let A, B be complete lattices.

$f : A \to B$ is **continuous** if for all $C \subseteq A$, $f(\sqcup C) = \sqcup_{c\in C} f(c)$.

$f : A \to B$ is **cocontinuous** if for all $C \subseteq A$, $f(\sqcap C) = \sqcap_{c\in C} f(c)$.

If f is continuous or cocontinuous, then f is monotone.

Proof. Let $a \leq b$. Then $a \sqcap b = a$ and $a \sqcup b = b$ and thus $f(a) \sqcap f(b) = f(a \sqcap b) = f(a)$ or $f(a) \sqcup f(b) = f(a \sqcup b) = f(b)$. Hence $f(a) \leq f(b)$. ❏

Let A be a poset and $f : A \to A$.

$a \in A$ is **f-closed** if $f(a) \leq a$. a is **f-dense** if $a \leq F(a)$. a is a **fixpoint** of f if $f(a) = a$.

Fixpoint Theorem of Knaster and Tarski [62]

Let A be a complete lattice and $f : A \to A$ be monotone.

(1) $\mathit{lfp}(f) = \sqcap\{a \in A \mid a \text{ is } f\text{-closed}\}$ is the least fixpoint of f.

(2) $\mathit{gfp}(f) = \sqcup\{a \in A \mid a \text{ is } f\text{-dense}\}$ is the greatest fixpoint of f.

Proof.

(1) Let a be f-closed. Then $\mathit{lfp}(f) \leq a$ and thus $f(\mathit{lfp}(f)) \leq f(a) \leq a$, i.e., $f(\mathit{lfp}(f))$ is a lower bound of all f-closed elements of A. Hence (3) $f(\mathit{lfp}(f)) \leq \mathit{lfp}(f)$. Since f is monotone, (3) implies that $f(\mathit{lfp}(f))$ is f-closed and thus (4) $\mathit{lfp}(f) \leq f(\mathit{lfp}(f))$. By (3) and (4), $\mathit{lfp}(f)$ is a fixpoint of f.

Let a be a fixpoint of f. Then a is f-closed and thus $\mathit{lfp}(f) \leq a$.

(2) Analogously. ❏

Zermelo's Fixpoint Theorem ([1], Prop. 1.3.1; [40], Ext. Folk Thm. 6; [9], Thm. 4.1.1)

(1) Let A be a λ-CPO with $|A| < \lambda$, $f : A \to A$ be monotone and $B = \{a_i \mid i < \lambda\}$ be the λ-chain of A that is defined as follows: $a_0 = \bot$, for all ordinals $i < \lambda$, $a_{i+1} = f(a_i)$, and for all limit ordinals $i < \lambda$, $a_i = \sqcup_{k<i} a_k$. For some $i < \lambda$, a_i is the least fixpoint f, i.e., $lfp(f) = f^{|A|}(\bot)$.

(2) Let A be a λ-coCPO with $|A| < \lambda$, $f : A \to A$ be monotone and $B = \{a_i \mid i < \lambda\}$ be the λ-cochain of A that is defined as follows: $a_0 = \top$, for all ordinals $i < \lambda$, $a_{i+1} = f(a_i)$, and for all limit ordinals $i < \lambda$, $a_i = \sqcap_{k<i} a_k$. For some $i < \lambda$, a_i is the greatest fixpoint f, i.e., $gfp(f) = f^{|A|}(\top)$.

Proof.

(1) First we show by transfinite induction on i that

$$\text{for all } i < \lambda,\ a_i \text{ is defined and for all } k < i,\ a_k \leq a_i. \tag{3}$$

Of course, $a_0 = \bot$ is defined. Let $i > 0$. If i is a successor ordinal, then $i = j+1$ for some j. By induction hypothesis, a_j is defined and for all $k < j$, $a_k \leq a_j$. Hence $a_i = f(a_j)$ is defined. Since f is monotone, $a_k = a_{k+1} = f(a_k) \leq f(a_j) = a_i$. If i is a limit ordinal, then by induction hypothesis, for all $k < j < i$, a_j is defined and $a_k \leq a_j$.

Hence $C = \{a_k \mid k < i\}$ is a λ-chain and thus $a_i = \sqcup C$ exists. Hence for all $k < i$, $a_k \leq a_i$.

We conclude from (3) that B is a λ-chain.

Assume that for all $i < \lambda$, $a_i \neq a_{i+1}$. Then $\{a_i \mid i < \lambda\}$ were a subset of A with cardinality λ, which contradicts the assumption that the cardinality of A is less than λ. Hence $a_i = a_{i+1} = f(a_i)$ for some $i < \lambda$, i.e., a_i is a fixpoint of f.

Let b be a fixpoint of f. We show by transfinite induction on i that

for all $i < \lambda$, $a_i \leq b$. (4)

Of course, $a_0 = \bot \leq b$. Let $i > 0$. If i is a successor ordinal, then $i = j + 1$ for some j. By induction hypothesis, $a_j \leq b$ and thus $a_i = a_{j+1} = f(a_j) \leq f(b) = b$ because f is monotone. If i is a limit ordinal, then $a_i = \sqcup_{k<i} a_k$. By induction hypothesis, for all $k < i$, $a_k \leq b$. Hence $a_i \leq b$.

We conclude from (4) that a_i is the *least* fixpoint of f.

(2) Analogously. ❏

Fixpoint induction

Let

(a) A be a complete lattice or a λ-CPO with $|A| < \lambda$ and $f : A \to A$ be monotone or
(b) A be an ω-CPO and f be ω-continuous.

(1) For all f-closed $a \in A$, $\mathit{lfp}(f) \leq a$.
(2) For all $n > 0$ and f^n-closed $a \in A$, $\mathit{lfp}(f) \leq a$.

Proof. (1) Let (a) hold true. If A is a complete lattice, then by the Fixpoint Theorem of Knaster and Tarski, $\mathit{lfp}(f) = \sqcap\{a \in A \mid f(a) \leq a\} \leq a$. If A is a λ-CPO, then by transfinite induction on i, for all $i < \lambda$, $f^i(\bot) \leq a$ because f is monotone and a is f-closed. Hence by Zermelo's Fixpoint Theorem, $\mathit{lfp}(f) = f^{|A|}(\bot) \leq a$. Let (b) hold true. By induction on n, for all $i \in \mathbb{N}$, $f^i(\bot) \leq a$ because f is monotone and a is f-closed. Hence by Kleene's Fixpoint Theorem (1), $\mathit{lfp}(f) = \sqcup_{i \in \mathbb{N}} f^i(\bot) \leq a$.

(2) Let (a) hold true. If A is a complete lattice, then

$$b =_{def} \sqcap_{i>0} f^i(a) \leq f^n(a) \leq a = f^0(a). \tag{4}$$

By (3), for all $i > 0$, $b \leq f^{i-1}(a)$ and thus $f(b) \leq f^i(a)$ because f is monotone. Hence $f(b)$ is a lower bound of $\{f^i(a) \mid i > 0\}$ and thus $f(b) \leq b$, i.e., b is f-closed. By the Fixpoint Theorem of Knaster and Tarski, $lfp(f) = \sqcap\{c \in A \mid f(c) \leq c\}$. Hence (3) implies $lfp(f) \leq b \leq a$. If A is a λ-CPO, then by transfinite induction on i, for all $i < \lambda$, $f^{n*i}(\bot) \leq a$ because f is monotone and a is f-closed. Hence by Zermelo's Fixpoint Theorem, $lfp(f) = f^{|A|}(\bot) \leq a$. Let (b) hold true. By induction on n, for all $i \in \mathbb{N}$, $f^{n*i}(\bot) \leq a$ because f is monotone and a is f-closed. Hence by Kleene's Fixpoint Theorem (1), $lfp(f) = \sqcup_{i \in \mathbb{N}} f^i(\bot) = \sqcup_{i \in \mathbb{N}} f^{n*i}(\bot) \leq a$. ❏

Fixpoint coinduction

Let

(a) A be a complete lattice or a λ-coCPO with $|A| < \lambda$ and $f : A \to A$ be monotone or
(b) A be an ω-coCPO and f be ω-cocontinuous.

(1) For all f-dense $a \in A$, $a \leq gfp(f)$.
(2) For all $n > 0$ and f^n-dense $a \in A$, $a \leq gfp(f)$.

Proof. Analogously. ❏

Computational induction

Let A be an ω-CPO, $f : A \to A$ be ω-continuous and B be an **admissible** subset of A, i.e., for all ω-chains C of A, $C \subseteq B$ implies $\sqcup C \in B$.

If $\bot \in B$ and for all $b \in B$, $f(b) \in B$, then $lfp(f) \in B$. (1)

Proof. (1) provides the induction base and the induction step of a proof by induction on n that for all $n \in \mathbb{N}$, $f^n(\bot) \in B$. Since B is admissable, we conclude $lfp(f) = \sqcup_{n\in\mathbb{N}} f^n(\bot) \in B$ by Kleene's Fixpoint Theorem (1). ❏

Computational coinduction

Let A be an ω-coCPO, $f : A \to A$ be ω-cocontinuous and B be an **co-admissible** subset of A, i.e., for all ω-cochains C of A, $C \subseteq B$ implies $\sqcap C \in B$.

If $\top \in B$ and for all $b \in B$, $f(b) \in B$, then $gfp(f) \in B$.

Proof. Analogously. ❑

Noetherian induction

Let A be a class, R be a well-founded relation on A and B be a subset of A.

If for all $a \in A$, $(\forall\ b \in A : bRa \Rightarrow b \in B)$ implies $a \in B$, then $B = A$.

Proof. Assume that the premise holds true, but there is $a \in A \setminus B$. Then the premise implies bRa and $b \notin B$ for some $b \in A$, i.e., $b \in A \setminus B$. We may repeat this conclusion (with b instead of a) infinitely often and thus obtain a subset of A without a least element w.r.t. R. ❑

If R is a well-order, then Noetherian induction is also called **transfinite induction**.

Categories

poset notion	*categorical notion*
element	object
a	A
ordered pair	morphism
$a \leq b$	$f : A \to B$
least element	initial object
greatest element	final object
upper bound	cocone
lower bound	cone
supremum (least upper bound)	colimit
infimum (greatest lower bound)	limit
λ-complete poset (CPO)	λ-cocomplete category $\mathcal{K}$
λ-cocomplete poset	λ-complete category $\mathcal{K}$

complete lattice	complete and cocomplete category
monotone function	functor
$a \leq b \;\Rightarrow\; f(a) \leq f(b)$	$A \xrightarrow{f} B \;\Rightarrow\; F(A) \xrightarrow{F(f)} F(B)$
f-closed element a: $f(a) \leq a$	α F-algebra: $F(A) \xrightarrow{\alpha} A$
f-dense element a: $a \leq f(a)$	α F-coalgebra: $A \xrightarrow{\alpha} F(A)$
λ-continuous function	λ-cocontinuous functor
$f(\sqcup_{i<\lambda} a_i) = \sqcup_{i<j<\lambda} f(a_i)$	$F(colim\{f_{i,j} : A_i \to A_j\}_{i<\lambda})$ $= \; colim\{F(f_{i,j}) : F(A_i) \to F(A_j)\}_{i<j\lambda}$
λ-cocontinuous function	λ-continuous functor
$f(\sqcap_{i<\lambda} a_i) = \sqcap_{i<\lambda} f(a_i)$	$F(lim\{f_{j,i} : A_j \to A_i\}_{i<j<\lambda})$ $= \; lim\{F(f_{j,i}) : F(A_j) \to F(A_i)\}_{i<j<\lambda}$
Galois connection	adjunction $F \dashv G$
$f(a) \leq b \Leftrightarrow a \leq g(b)$	$\dfrac{A \to G(B)}{F(A) \to B}$

A (locally small) **category** $\mathcal{K}$ consists of

- a class of **objects**, also denoted by $\mathcal{K}$,
- for all $A, B \in \mathcal{K}$ a set $\mathcal{K}(A, B)$ of $\mathcal{K}$**-morphisms**,
- an associative **composition**

$$\circ : \mathcal{K}(A, B) \times \mathcal{K}(B, C) \longrightarrow \mathcal{K}(A, C)$$

$$(f, g) \longmapsto g \circ f,$$

- for all $A \in \mathcal{K}$ an **identity** $id_A \in \mathcal{K}(A, A)$ such that for all $B \in \mathcal{K}$ und $f \in \mathcal{K}(A, B)$, $f \circ id_A = f = id_B \circ f$. id_A is also written as just A.

If the class of all objects of $\mathcal{K}$ is a set, then $\mathcal{K}$ is **small**.

$Mor(\mathcal{K})$ denotes the class of all sets $\mathcal{K}(A, B)$ with $A, B \in \mathcal{K}$.
$f \in \mathcal{K}(A, B)$ is usually written as $f : A \to B$. A is the **source** und B the **target** of f.

A category $\mathcal{L}$ is a **subcategory** of $\mathcal{K}$ if all objects of $\mathcal{L}$ are objects of $\mathcal{K}$ and all $\mathcal{L}$-morphisms are $\mathcal{K}$-morphisms. $\mathcal{L}$ is **full** if all $\mathcal{K}$-morphisms between objects of $\mathcal{L}$ are $\mathcal{L}$-morphisms.

$f \in \mathcal{K}(A, B)$ is (an) **epi(morphismus)** if for all $g, h \in \mathcal{K}(B, C)$, $g \circ f = h \circ f$ implies $g = h$.

$f \in \mathcal{K}(A, B)$ is (a) **mono(morphismus)** if for all $g, h \in \mathcal{K}(C, A)$, $f \circ g = f \circ h$ implies $g = h$.

$g \in \mathcal{K}(B, A)$ is a **retraction** or **split epi** if $g \circ f = id_A$ for some $f \in \mathcal{K}(A, B)$.

$f \in \mathcal{K}(A, B)$ is a **coretraction**, **section** or **split mono** if $g \circ f = id_A$ for some $g \in \mathcal{K}(B, A)$.

$f \in \mathcal{K}(A, B)$ is (an) **iso(morphismus)** and A and B are **isomorphic**, written as $A \cong B$, if f is a retraction and a coretraction.

If $f \in \mathcal{K}(A, B)$ is iso, then $g \in \mathcal{K}(B, A)$ with $g \circ f = id_A$ and $f \circ g = id_B$ is unique.

Isomorphism is the equality of category theory:
Isomorphic objects have the same categorical properties.

Lemma EPIMON Let $f \in \mathcal{K}(A, B)$ and $g \in \mathcal{K}(B, C)$.

- If $g \circ f$ is epi, then g is epi.
- If $g \circ f$ is mono, then f is mono.

The **dual category** $\mathcal{K}^{op}$ of $\mathcal{K}$ is constructed from $\mathcal{K}$ by keeping the objects, but reversing the arrows, i.e., for all $A, B \in \mathcal{K}$, $\mathcal{K}^{op}(A, B) = \mathcal{K}(B, A)$.

The **product category** $\mathcal{K} \times \mathcal{L}$ has pairs (A, B) of objects $A \in \mathcal{K}$ and $B \in \mathcal{L}$ as objects and pairs (f, g) of $\mathcal{K}$-morphisms $f : A \to C$ and $\mathcal{L}$-morphisms $f : B \to D$ as morphisms.

Let $\mathcal{K}$ be a category. A $\mathcal{K}$-object I is **initial in $\mathcal{K}$** if for all $\mathcal{K}$-objects A there is a unique $\mathcal{K}$-morphism $ini^A : I \to A$.

A $\mathcal{K}$-object F is **final** or **terminal in $\mathcal{K}$** if for all $\mathcal{K}$-objects A there is a unique $\mathcal{K}$-morphism $fin^A : A \to F$.

All initial $\mathcal{K}$-objects are isomorphic.

All final $\mathcal{K}$-objects are isomorphic.

The S-sorted set A with $A_s = \emptyset$ for all $s \in S$ is initial in Set^S.

Any S-sorted set A with $|A_s| = 1$ for all $s \in S$ is final in Set^S.

Lemma MINMAX

(1) If I is initial in $\mathcal{K}$, then all $\mathcal{K}$-monomorphisms $f : A \longrightarrow I$ are isomorphisms.

(2) If F is final in $\mathcal{K}$, then all $\mathcal{K}$-epimorphisms $g : F \longrightarrow A$ are isomorphisms.

Proof.

(1) Let I be initial in $\mathcal{K}$. Then $f \circ ini^A = id_I$. Hence $f \circ ini^A \circ f = id_I \circ f = f = f \circ id_A$ and thus $ini^A \circ f = id_A$ because f is mono. Hence f is iso.

(2) Let F be final in $\mathcal{K}$. Then $fin^A \circ g = id_F$. Hence $g \circ fin^A \circ g = g \circ id_F = g = id_A \circ g$ and thus $g \circ fin^A = id_A$ because g is epi. Hence g is iso. ❏

Let $\mathcal{K}$ be a category with final object $1_{\mathcal{K}}$. $X \in \mathcal{K}$ **has the fixpoint property** if for all $\mathcal{K}$-morphisms $f : X \to X$ there is $x : 1_{\mathcal{K}} \to X$ with $f \circ x = x$.

A $\mathcal{K}$-morphism $f : A \to X$ is **ubiquitous** if for all $\mathcal{K}$-morphisms $g : A \to X$ there is $a : 1_{\mathcal{K}} \to A$ with $f \circ a = g \circ a$.

Lawvere's Fixpoint Theorem ([59], Thms. 1 quarto and 5; [70], Thm. 1)

Let $\mathcal{K}$ be a category with final object $1_{\mathcal{K}}$.

(1) $X \in \mathcal{K}$ has the fixpoint property iff there is an ubiquitous $\mathcal{K}$-morphism $f : A \to X$.

(2) Let $\mathcal{K}$ be Cartesian closed (see Adjunctions) and $f : A \to X^A$ be a **surjective morphism**, i.e., for all $g : 1_{\mathcal{K}} \to X^A$ there is $a_g : 1_{\mathcal{K}} \to A$ such that $f \circ a_g = g$. Then X has the fixpoint property.

Proof.

(1) Let $f : A \longrightarrow X$ be ubiquitous and $g : X \longrightarrow X$ be a $\mathcal{K}$-morphism. Then $f \circ a_g = g \circ f \circ a_g$ for some $a_g : 1_{\mathcal{K}} \longrightarrow A$, i.e., $f \circ a_g$ is a fixpoint of g. Conversely, suppose that $X \in \mathcal{K}$ has the fixpoint property. Let $g : X \longrightarrow X$ be a $\mathcal{K}$-morphism with fixpoint x_g. Then $id_X(x_g) = x_g = g(x_g)$. Hence the identity on X is ubiquitous.

(2) By (1), it is sufficient to find an ubiquitous $\mathcal{K}$-morphism $h : A \longrightarrow X$. Define h as $f^* \circ \langle id_A, id_A \rangle$ and let $g : A \longrightarrow X$. Then

$$h \circ a_g = f^* \circ \langle id_A, id_A \rangle \circ a_g = f^* \circ \langle a_g, a_g \rangle = f \circ \pi_1 \langle a_g, a_g \rangle = f \circ a_g.$$

Hence h is ubiquitous. ❏

Corollaries

(1) Cantor: The set $2^{\mathbb{N}}$ of infinite bit streams is uncountable.

Proof. Let $\mathcal{K} = Set$. $g : 2 \longrightarrow 2$ with $g(0) = 1$ and $g(1) = 0$ does not have a fixpoint. Hence by Lawvere's Fixpoint Theorem (2), there is no surjective morphism $f : \mathbb{N} \longrightarrow 2^{\mathbb{N}}$ and thus $2^{\mathbb{N}}$ is uncountable. ❏

(2) For all sets A with $|A| \neq 2$, $|A| < |2^A|$.

Proof. Same argument as in the proof of (1). ❏

(3) Russell: The collection C of all sets that do not contain themselves is not a set.

Proof. Let $\mathcal{K}$ be the category of classes, A be the class of all sets and $f : A \longrightarrow 2^A$ be the function that maps each set B to its characteristic function $\chi_B : B \longrightarrow 2$, i.e., for all sets C, $\chi_B(C) = 1$ iff $C \in B$. Let $g : 2 \longrightarrow 2$ with $g(0) = 1$ and $g(1) = 0$. Assume that C is a set. Then C is the pre-image of $h = g \circ f^* \circ \langle id_A, id_A \rangle$ in A under f, i.e., $f(C) = \chi_C = h$. This leads to a contradiction:

$$f(C)(C) = h(C) = g(f^*(C, C)) = g(f(C)(C)).$$

Hence $C \notin A$, i.e., C is not a set (and thus f is not surjective). This proof uses Lawvere's Fixpoint Theorem (2) only insofar as its conjecture is derived from the fact that 2 does not have the fixpoint property. ❏

[59], Section 3.1, and [70], §3 and §5, employ the same line of argument for re-establishing well-known "negative" results, such as the unsolvability of the halting problem (Turing), the incompleteness of arithmetic theories (Gödel) or the undefinability of truth (Tarski).

Functors and natural transformations

Functors are mappings between categories.

Natural transformations are mappings between functors.

Let $\mathcal{K}$ and $\mathcal{L}$ be two categories. A **functor** $F : \mathcal{K} \to \mathcal{L}$ maps each $\mathcal{K}$-Objekt to an $\mathcal{L}$-object and each $\mathcal{K}$-morphism $f : A \to B$ to an $\mathcal{L}$-morphism $F(f) : F(A) \to F(B)$ such that

- for all $\mathcal{K}$-objects A, $F(id_A) = id_{F(A)}$,
- for all $\mathcal{K}$-morphisms $f : A \to B$ and $g : B \to C$, $F(g \circ f) = F(g) \circ F(f)$.

If $\mathcal{K} = \mathcal{L}$, then F is called an **endofunctor**.

Example

The Haskell function $map\colon(a \to b) \to [a] \to [b]$ is a functor from Set to the category of monoids and monoid homomorphisms: for all $A \in Set$ and all functions $f : A \to B$,

$$map(A) \;=\; (A^*, ++, []),$$
$$map(f)([a_1, \ldots, a_n]) \;=\; [f(a_1), \ldots, f(a_n)].$$

$Id_{\mathcal{K}} : \mathcal{K} \to \mathcal{K}$ denotes the **identity functor** that maps each object or morphism of $\mathcal{K}$ to itself.

The **Hom functor** $Hom : \mathcal{K}^{op} \times \mathcal{K} \to Set$ maps $(A, B) \in \mathcal{K}^{op} \times \mathcal{K}$ to $\mathcal{K}(A, B)$ and $(f : C \to A, g : B \to D) \in \mathcal{K}^{op}(A, C) \times \mathcal{K}(B, D)$ to $\lambda h : A \to B.(g \circ h \circ f : C \to D)$.

The category **Cat** has categories $\mathcal{K}$ as objects and functors $F : \mathcal{K} \to \mathcal{L}$ as morphisms.

Given two functors $F, G : \mathcal{K} \to \mathcal{L}$, a **natural transformation** $\tau : F \to G$ assigns to each object $A \in \mathcal{K}$ an $\mathcal{L}$-morphism $\tau_A : F(A) \to G(A)$ such that for all $\mathcal{K}$-morphisms $f : A \to B$ the following diagram commutes:

$$\begin{array}{ccc} F(A) & \xrightarrow{\tau_A} & G(A) \\ {\scriptstyle F(f)}\downarrow & & \downarrow{\scriptstyle G(f)} \\ F(B) & \xrightarrow{\tau_B} & G(B) \end{array}$$

If for all $A \in \mathcal{K}$, τ_A is an isomorphism, then $\tau : F \to G$ is a **natural equivalence** and F and G are **naturally equivalent**.

Compositions of functors and/or natural transformations

- Let $F : \mathcal{K} \to \mathcal{L}$ and $G : \mathcal{L} \to \mathcal{M}$.
  Then $FG : \mathcal{K} \to \mathcal{M}$ and for all $A \in \mathcal{K}$, $GF(A) = G(F(A))$.
- Let $F, G : \mathcal{K} \to \mathcal{L}$, $\tau : F \to G$ and $H : \mathcal{L} \to \mathcal{M}$.
  Then $H\tau : HF \to HG$ and for all $A \in \mathcal{K}$, $(H\tau)_A = H\tau_A$.
- Let $F : \mathcal{K} \to \mathcal{L}$, $G, H : \mathcal{L} \to \mathcal{M}$ and $\tau : G \to H$.
  Then $\tau F : GF \to HF$ and for all $A \in \mathcal{K}$, $(\tau F)_A = \tau_{F(A)}$.
- *Vertical Composition.* Let $F, G, H : \mathcal{K} \to \mathcal{L}$, $\tau : F \to G$ and $\eta : G \to H$.
  Then $\eta\tau : F \to H$ and for all $A \in \mathcal{K}$, $(\eta\tau)_A = \eta_A \circ \tau_A$.
- *Horizontal Composition.* Let $F, G : \mathcal{K} \to \mathcal{L}$, $\tau : F \to G$, $F', G' : \mathcal{L} \to \mathcal{M}$ and $\tau' : F' \to G'$. Then

$$F'F \xrightarrow{\tau'\tau} G'G \;=\; F'F \xrightarrow{F'\tau} F'G \xrightarrow{\tau'G} G'G \;=\; F'F \xrightarrow{\tau'F} G'F \xrightarrow{G'\tau} G'G.$$

Given two categories $\mathcal{K}$ and $\mathcal{L}$, the category $\mathbf{Fun}(\mathcal{K},\mathcal{L})$ has all functors $F : \mathcal{K} \to \mathcal{L}$ as objects and all natural transformations between such functors and their vertical compositions as morphisms.

Let $T : Set \to Set$ be a functor and A be a set.

The **strength**

$$st^{T,A} : T(-)^A \to (-)^A T$$

of T and A is defined as follows (see [32], p. 380): For all sets B, $g \in T(B^A)$ and $a \in A$,

$$st_B^{T,A}(g)(a) = T((\lambda f \colon B^A.f(a)) \colon B^A \to B)(g) \colon T(B).$$

$st^{T,A}$ is a natural transformation, i.e., for all $h : B \to C$, the following diagram commutes:

$$\begin{array}{ccc} T(B^A) & \xrightarrow{st_B^{T,A}} & T(B)^A \\ {\scriptstyle T(h^A)}\downarrow & & \downarrow{\scriptstyle T(h)^A} \\ T(C^A) & \xrightarrow{st_C^{T,A}} & T(C)^A \end{array}$$

Proof. For all $g \in T(B^A)$ and $a \in A$,

$$(T(h)^A \circ st_B^{T,A}(g))(a) = (T(h)^A \circ \lambda a.T(\lambda f.f(a))(g))(a) = T(h)^A(T(\lambda f.f(a))(g))$$
$$= (T(h) \circ T(\lambda f.f(a)))(g) = T(h \circ \lambda f.f(a))(g) = T(\lambda f.h(f(a)))(g),$$
$$st_C^{T,A}(T(h^A)(g))(a) = T(\lambda f.f(a))(T(h^A)(g)) = (T(\lambda f.f(a)) \circ T(h^A))(g)$$
$$= T((\lambda f.f(a)) \circ h^A)(g) \overset{(*)}{=} T(\lambda f.h(f(a)))(g)$$

❑

Lemma

$$(\lambda f.f(a)) \circ h^A = \lambda f.h(f(a)). \qquad (*)$$

Proof of $(*)$. For all $a \in A$ and $g \in B^A$,

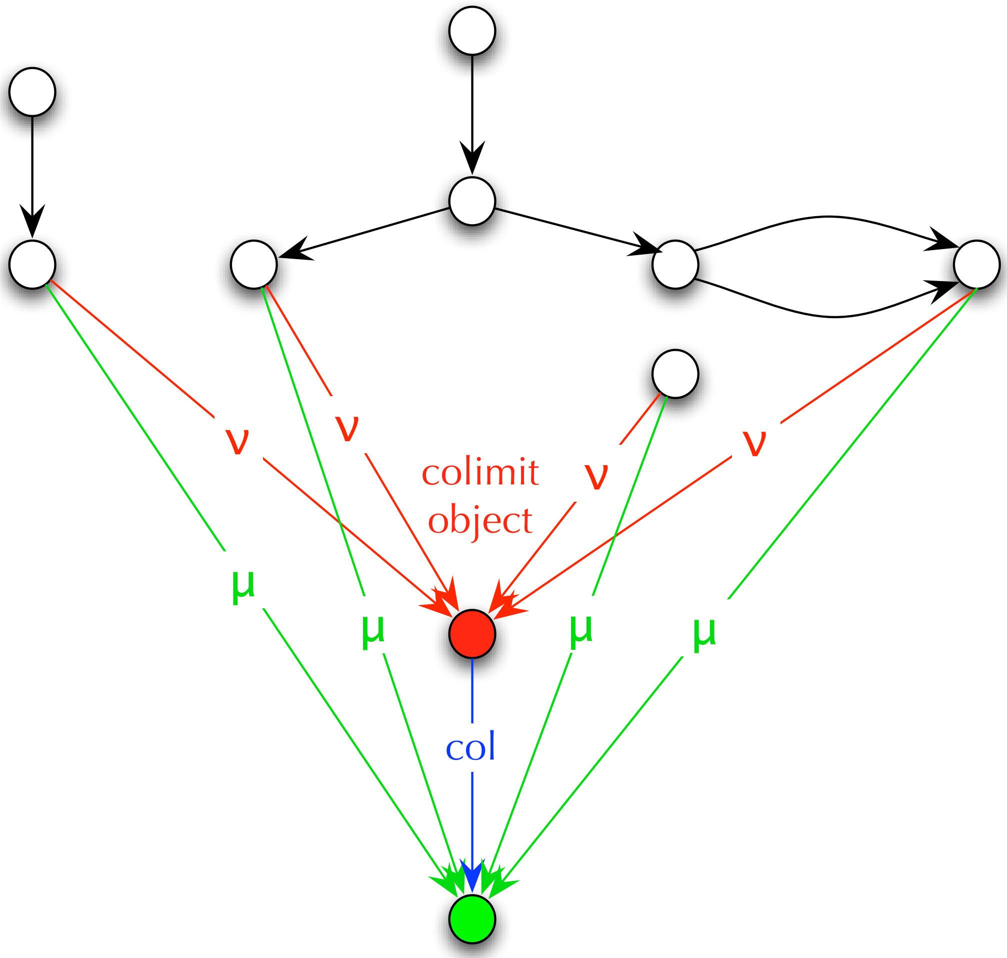
$$(\lambda f.f(a))(h^A(g)) = (\lambda f.f(a))(h \circ g) = h(g(a)) = (\lambda f.h(f(a)))(g).$$

❑

Limits and colimits

Given two categories $\mathcal{I}$ and $\mathcal{K}$, a **diagram of type $\mathcal{I}$ in $\mathcal{K}$** is a functor $\mathcal{D} : \mathcal{I} \to \mathcal{K}$.

The actual objects and morphisms in $\mathcal{I}$ are irrelevant, only the way in which they are interrelated matters. $\mathcal{D}$ is thought of as indexing a collection of objects and morphisms in $\mathcal{K}$ patterned on $\mathcal{I}$. One may also view $\mathcal{D}$ as the node- resp. edge-labelling function of a labelled graph whose nodes and edges are the objects resp. morphisms of $\mathcal{I}$.



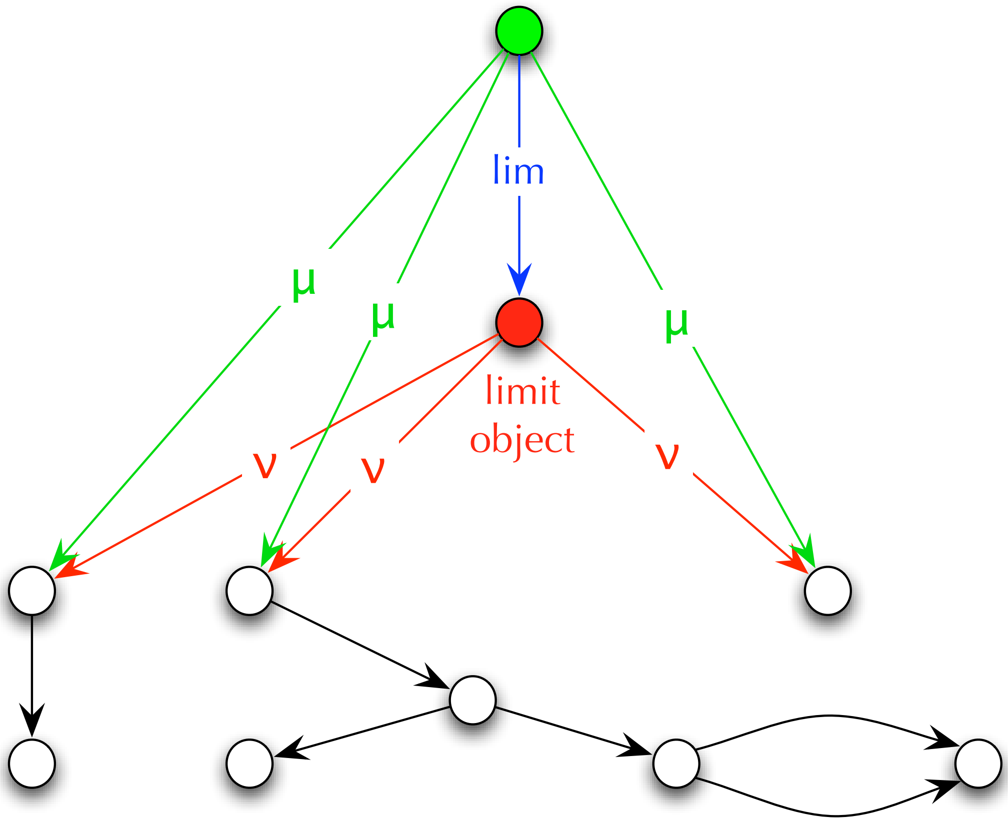
A diagram, its colimit and a further cocone

A set $\mu = \{\mu_n : \mathcal{D}(n) \to C \mid n \in \mathcal{I}\})$ of $\mathcal{K}$-morphisms is a **cocone of** $\mathcal{D}$ if for all $e \in \mathcal{I}(m,n)$, $\mu_m = \mu_n \circ \mathcal{D}(e)$. C is called the **target** of μ.

A cocone ν of $\mathcal{D}$ with target C is a **colimit of** $\mathcal{D}$ if for all $D \in \mathcal{K}$ and cocones μ of $\mathcal{D}$ there is a unique $\mathcal{K}$-morphism $col^D : C \to D$ such that for all $n \in \mathcal{I}$, $col^D \circ \nu_n = \mu_n$.

All colimits of $\mathcal{D}$ are isomorphic.

An object is initial in $\mathcal{K}$ if it is the target object of a colimit of the empty diagram $\emptyset \to \mathcal{K}$.



A diagram, its limit and a further cone

A set $\mu = \{\mu_n : C \to \mathcal{D}(n) \mid n \in \mathcal{I}\}$ of $\mathcal{K}$-morphisms is a **cone of** $\mathcal{D}$ if for all $e \in \mathcal{I}(m, n)$, $\mathcal{D}(e) \circ \mu_m = \mu_n$. C is called the **source** of μ.

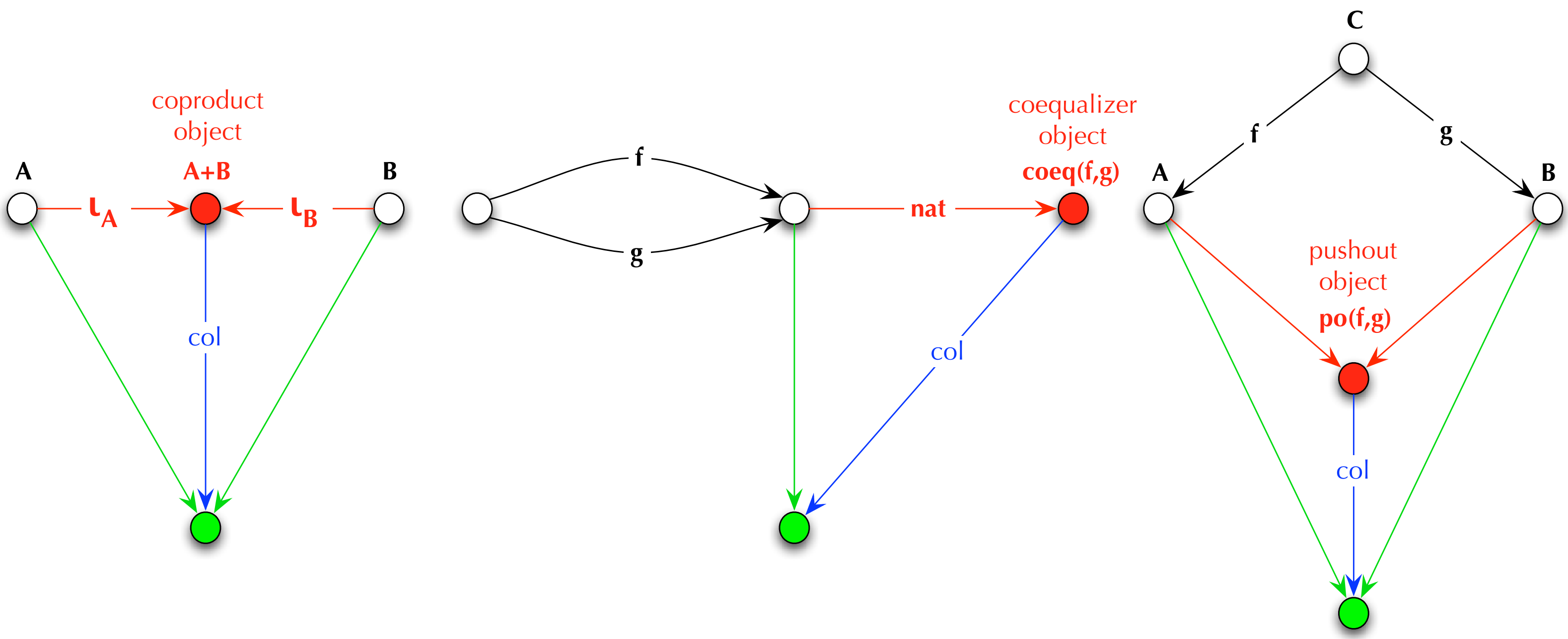
A cone ν of $\mathcal{D}$ with source C is a **limit of** $\mathcal{D}$ if for all $D \in \mathcal{K}$ and cones μ of $\mathcal{D}$ there is a unique $\mathcal{K}$-morphism $lim^D : D \to C$ such that for all $n \in \mathcal{I}$, $\nu_n \circ lim^D = \mu_n$.

All limits of $\mathcal{D}$ are isomorphic.

An object is final in $\mathcal{K}$ if it is the source object of a limit of the empty diagram $\emptyset \to \mathcal{K}$.

$\mathcal{K}$ is **cocomplete** if each diagram in $\mathcal{K}$ has a colimit.

$\mathcal{K}$ is **complete** if each diagram in $\mathcal{K}$ has a limit.



The **coproduct** $A+B$, *the* **coequalizer** $coeq(f,g)$
and the **pushout** $po(f,g) = coeq(\iota_A \circ f, \iota_B \circ g)$ *are colimits.*

If C is initial in $\mathcal{K}$, then $po(f,g) = A+B$.

Coequalizers are epimorphisms.

Let $A + B$ be a coproduct (object) of A and B and I be initial in $\mathcal{K}$.

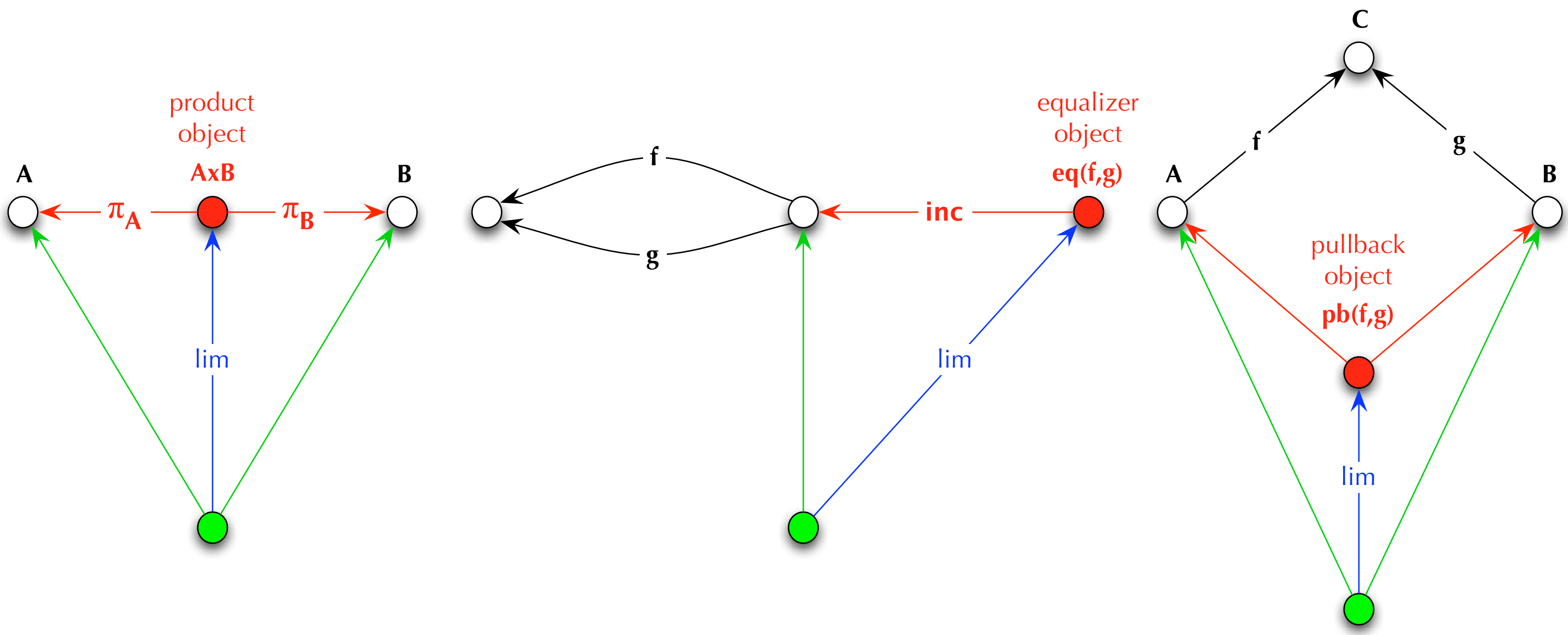
Since all coproducts with the same summands are isomorphic,
$A + (B + C) \cong (A + B) + C \cong A + B + C$, $A + B \cong B + A$ and $I + A \cong A$.

Let $\mathcal{K} = Set^S$.

The coequalizer of $f, g : A \to B$ is the quotient of B by the equivalence closure of $R = \{(f(a), g(a)) \in B \times B \mid a \in A\}$ together with the corresponding natural map that sends an element of B to its equivalence class.

The pushout of $f : A \to B$ and $g : A \to C$ is the quotient of $B \cup C$ by the equivalence closure of $R = \{(f(a), g(a)) \in B \times C \mid a \in A\}$ together with the corresponding natural maps that send an element of B resp. C to its equivalence class.

If f and g are inclusion maps, then the pushout object is isomorphic to $B \cup C$.



The **product** $A \times B$, *the* **equalizer** $eq(f, g)$
and the **pullback** $pb(f, g) = eq(f \circ \pi_A, g \circ \pi_B)$ *are limits.*

If C is final in $\mathcal{K}$, then $pb(f, g) = A \times B$.

Equalizers are monomorphisms.

Let $A \times B$ be a product (object) of A and B, F be final in $\mathcal{K}$ and I be initial in $\mathcal{K}$.

Since all products with the same factors are isomorphic,
$A \times (B \times C) \cong (A \times B) \times C \cong A \times B \times C$, $A \times B \cong B \times A$, $A \times F \cong A$ and $A \times I \cong I$.

Let $\mathcal{K} = Set^S$.

The equalizer of $f, g : A \to B$ is the set of all $a \in A$ such that $f(a) = g(a)$ together with the corresponding inclusion map.

The pullback of $f : A \to C$ and $g : B \to C$ is the set of all $(a, b) \in A \times B$ such that $f(a) = g(b)$ together with the corresponding projections.

If f and g are inclusion maps, then the pullback object is isomorphic to $A \cap B$.

Quotient Theorem (construction of colimits in Set)

A cocone ν of a diagram $\mathcal{D} : \mathcal{I} \to \mathcal{K}$ in Set is the colimit of $\mathcal{D}$ iff the target C of ν is isomorphic to the quotient

$$(\coprod_{n\in\mathcal{I}} \mathcal{D}(n))/\!\sim$$

of the disjunct union over N of all node labels of $\mathcal{D}$ by the equivalence closure $\sim$ of

$$\{(a, \mathcal{D}(e)(a)) \in (\coprod_{n\in\mathcal{I}} \mathcal{D}(n))^2 \mid a \in \mathcal{D}(m),\ e \in \mathcal{I}(m,n),\ m,n \in \mathcal{I}\}.$$

For all $n \in \mathcal{I}$, $\nu_n : \mathcal{D}(n) \to C$ is the composition of the injection

$$\iota_n : \mathcal{D}(n) \to \coprod_{n\in\mathcal{I}} \mathcal{D}(n)$$

with the natural map $nat : \coprod_{n\in\mathcal{I}} \mathcal{D}(n) \to C$.

Subset Theorem (construction of limits in *Set*)

A cone ν of a diagram $\mathcal{D} : \mathcal{I} \to \mathcal{K}$ in Set is the limit of $\mathcal{D}$ iff the source C of ν is isomorphic to the subset

$$\{a \in \prod_{n \in \mathcal{I}} \mathcal{D}(n) \mid \forall\ m, n \in \mathcal{I},\ e \in \mathcal{I}(m,n) : \mathcal{D}(e)(\pi_m(a)) = \pi_n(a)\}$$

of the product over $\mathcal{I}$ of the images under $\mathcal{D}$.

For all $n \in \mathcal{I}$, $\nu_n : C \to \mathcal{D}(n)$ is the composition of the inclusion

$$inc : C \to \prod_{n \in \mathcal{I}} \mathcal{D}(n)$$

with the projection $\pi_n : \prod_{n \in \mathcal{I}} \mathcal{D}(n) \to \mathcal{D}(n)$.

Colimit Theorem
(generalizes the Quotient Theorem to cocomplete categories)

Let $\mathcal{K}$ be a category such that each family of $\mathcal{K}$-objects has a coproduct and each pair $f, g : A \to B$ of $\mathcal{K}$-morphisms has a coequalizer.

A cocone ν of a $\mathcal{K}$-diagram $\mathcal{D} : \mathcal{I} \to \mathcal{K}$ is the colimit of $\mathcal{D}$ if the target C of ν is isomorphic to the coequalizer object of the pair of $\mathcal{K}$-morphisms

$$\psi_1, \psi_2 : \coprod_{m \in \mathcal{I}} \{\mathcal{D}(m) \mid e \in \mathcal{I}(m,n)\} \to \coprod_{n \in \mathcal{I}} \mathcal{D}(n)$$

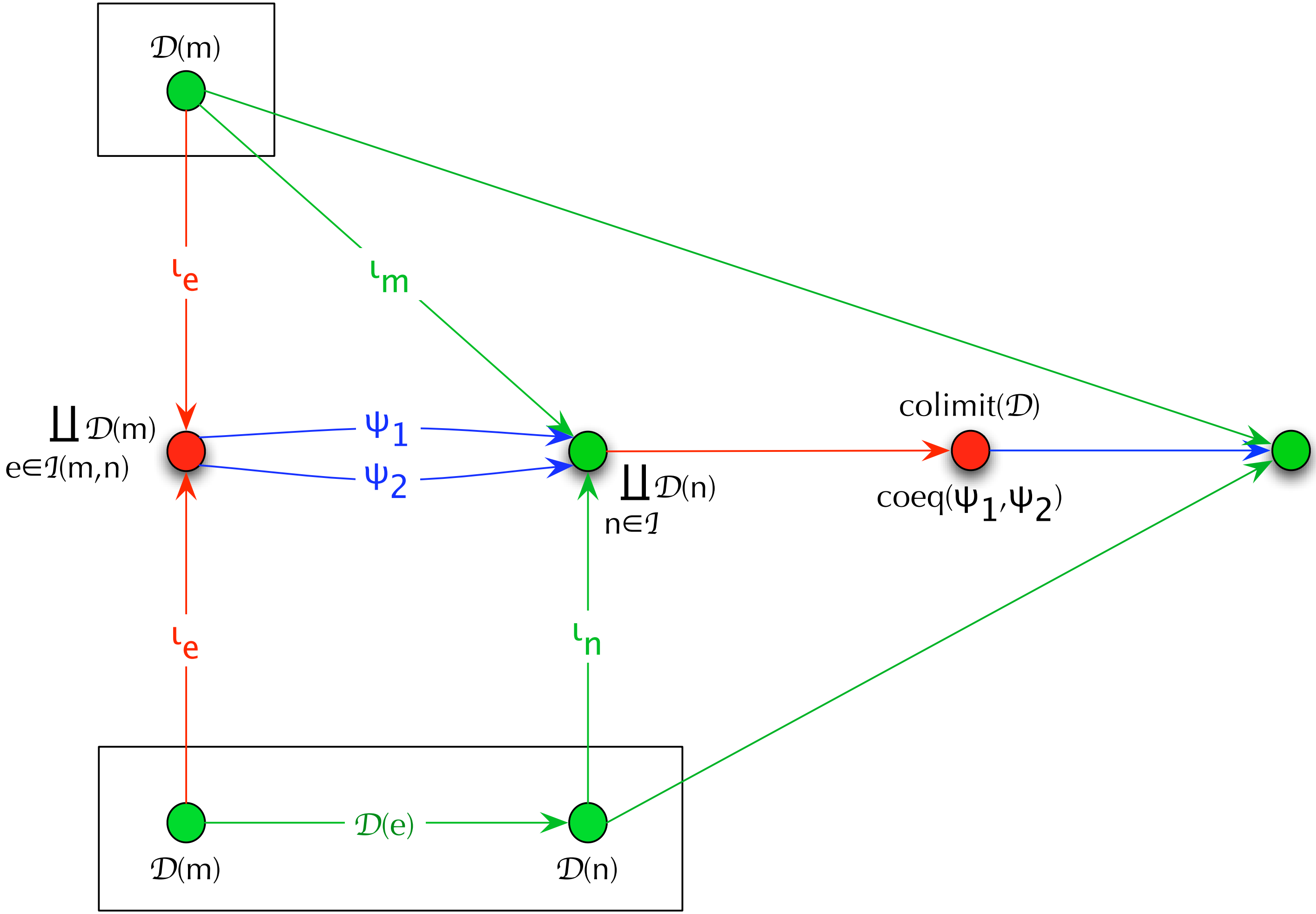
where ψ_1 and ψ_2 are the coproduct extensions of

$$\{\iota_m : \mathcal{D}(m) \to \coprod_{n \in \mathcal{I}} \mathcal{D}(n) \mid m \in \mathcal{I}\}$$

and

$$\{\iota_n \circ \mathcal{D}(e) : \mathcal{D}(m) \to \coprod_{n \in \mathcal{I}} \mathcal{D}(n) \mid e \in \mathcal{I}(m,n)\},$$

respectively.



colimit($\mathcal{D}$) coequalizes the coproduct extensions ψ_1 and ψ_2.

Limit Theorem
(generalizes the Subset Theorem to complete categories)

Let $\mathcal{K}$ be a category such that each family of $\mathcal{K}$-objects has a product and each pair $f, g : A \to B$ of $\mathcal{K}$-morphisms has an equalizer.

A cone ν of a $\mathcal{K}$-diagram $\mathcal{D} : \mathcal{I} \to \mathcal{K}$ is the limit of $\mathcal{D}$ if the source C of ν is isomorphic to the equalizer object of the pair of $\mathcal{K}$-morphisms

$$\psi_1, \psi_2 : \prod_{m \in \mathcal{I}} \mathcal{D}(m) \to \prod_{n \in \mathcal{I}} \{\mathcal{D}(n) \mid e \in \mathcal{I}(m,n)\}$$

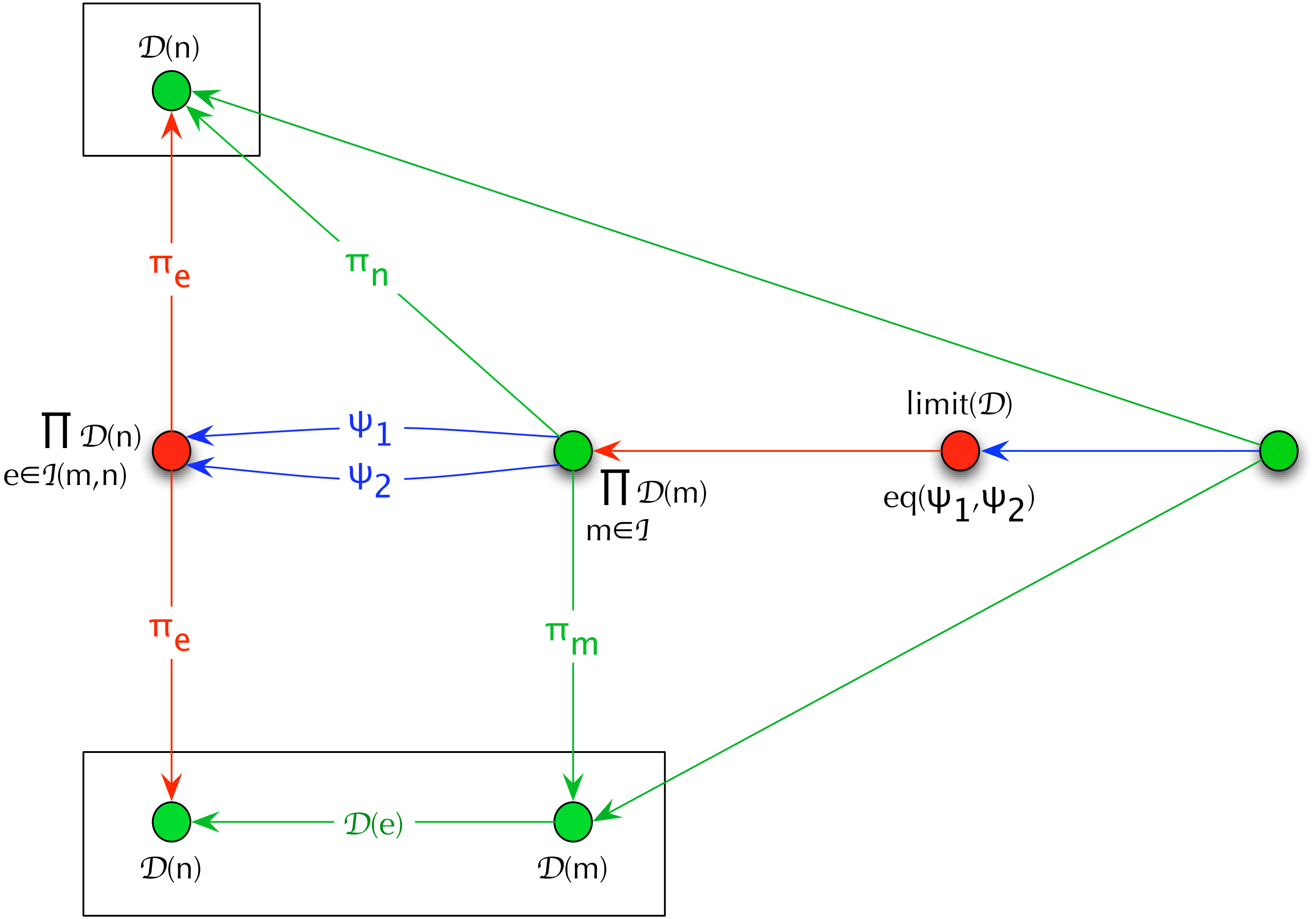
where ψ_1 and ψ_2 are the product extensions of

$$\{\pi_n : \prod_{m \in \mathcal{I}} \mathcal{D}(m) \to \mathcal{D}(n) \mid n \in \mathcal{I}\}$$

and

$$\{\mathcal{D}(e) \circ \pi_m : \prod_{m \in \mathcal{I}} \mathcal{D}(m) \to \mathcal{D}(n) \mid e \in \mathcal{I}(m,n)\},$$

respectively.



limit($\mathcal{D}$) equalizes the product extensions ψ_1 and ψ_2.

Sorted sets, functions and relations

Let S be a finite set of **sorts**.

An S**-sorted** or S**-indexed set** is a tuple $A = (A_s)_{s \in S}$ of sets. A is **nonempty** if for all $s \in S$, $A_s \neq \emptyset$.

An S**-sorted subset** B of A, written as $B \subseteq A$, is an S-sorted set with $B_s \subseteq A_s$ for all $s \in S$.

Given S-sorted sets $A_1, \ldots, A_n$, an S**-sorted relation** $r \subseteq A_1 \times \cdots \times A_n$ is an S-sorted set with $r_s \subseteq A_{1,s} \times \ldots \times A_{n,s}$ for all $s \in S$. If $n = 2$ and $A_1 = A_2$, then r is a **binary relation on** A_1.

Given S-sorted sets A, B, an S**-sorted function** $f : A \to B$ is an S-sorted set such that for all $s \in S$, f_s is a function from A_s to B_s.

B^A denotes the set of S-sorted functions from A to B.

Set^S denotes the product category of S-sorted sets as objects and S-sorted functions as morphisms.

Let f be an S-sorted function.

f is epi iff f is surjective. f is mono iff f is injective. f is iso iff f is bijective.

The **diagonal of A^2** is the S-sorted binary relation Δ_A with $\Delta_{A,s} = \Delta_{A_s}$.

Let BS be a finite set of sets. $\mathbb{T}(S, BS)$ denotes the inductively defined set of **types over** S **and** BS:

$$
\begin{array}{lcll}
s \in S & \Rightarrow & s \in \mathbb{T}(S, BS), & \text{(set variables)} \\
X \in BS & \Rightarrow & X \in \mathbb{T}(S, BS), & \text{(constant types are sets)} \\
e_1, \dots, e_n \in \mathbb{T}(S, BS) & \Rightarrow & e_1 \times \cdots \times e_n, e_1 + \cdots + e_n \in \mathbb{T}(S, BS), & \\
e \in \mathbb{T}(S, BS) & \Rightarrow & word(e), bag(e), set(e) \in \mathbb{T}(S, BS), & \\
 & & & \text{(word, bag and set types)} \\
X \in BS \wedge e \in S & \Rightarrow & e^X \in \mathbb{T}(S, BS). & \text{(power types)}
\end{array}
$$

We regard $e \in \mathbb{T}(S, BS)$ as a finite tree: Each inner node of e is labelled with a **type constructor** ($\times$, $+$, *word*, *bag*, *set* or $\_^X$ for some $X \in BS$) and each leaf is labelled with an element of S or BS.

$e \in \mathbb{T}(S, BS)$ is **flat** if $e \in S \cup BS$ or $e \in \{word(s), bag(s), set(s)\}$ for some $s \in S$. $\mathbb{FT}(S, BS)$ denotes the set of flat types over S and BS.

A **collection type** is a word, bag or set type. A type is **polynomial** if it does not contain set types.

The semantics of e is a functor $F_e : Set^S \to Set$ (also called **predicate lifting**; see [29, 30]) that is inductively defined as follows:

Let A, B be S-sorted sets, $h : A \to B$ be an S-sorted function, $s \in S$, $X \in BS$, $e, e_1, \dots, e_n \in \mathbb{T}(S, BS)$, $a_1, \dots, a_n \in F_e(A)$, $f \in \mathcal{B}_{\mathit{fin}}(F_e(A))$, $g \in \mathcal{P}_{\mathit{fin}}(F_e(A))$, $b \in F_e(B)$ and $g' : X \to F_e(A)$.

$$
\begin{array}{ll}
F_s(A) = A_s, & F_s(h) = h_s, \\
F_X(A) = X, & F_X(h) = id_X, \qquad \text{(constant functors)} \\
F_{e_1 \times \dots \times e_n}(A) = F_{e_1}(A) \times \ldots \times F_{e_n}(A), & F_{e_1 \times \dots \times e_n}(h) = F_{e_1}(h) \times \ldots \times F_{e_n}(h), \\
F_{e_1 + \dots + e_n}(A) = F_{e_1}(A) + \dots + F_{e_n}(A), & F_{e_1 + \dots + e_n}(h) = F_{e_1}(h) + \dots + F_{e_n}(h), \\
F_{word(e)}(A) = F_e(A)^*, & F_{word(e)}(h)(a_1 \dots a_n) = F_e(h)(a_1) \dots F_e(h)(a_n), \\
F_{bag(e)}(A) = \mathcal{B}_{\mathit{fin}}(F_e(A)), & F_{bag(e)}(h)(f)(b) = \sum\{f(a) \mid a \in F_e(A),\ F_e(h)(a) = b\}, \\
F_{set(e)}(A) = \mathcal{P}_{\mathit{fin}}(F_e(A)), & F_{set(e)}(h)(g)(b) = \bigvee\{g(a) \mid a \in F_e(A),\ F_e(h)(a) = b\}, \\
F_{e^X}(A) = F_e(A)^X, & F_{e^X}(h)(g') = F_e(h) \circ g'.
\end{array}
$$

Hence predicate lifting extends S-sorted sets to $\mathbb{T}(S, BS)$-sorted sets.

We often write A_e for the set $F_e(A)$ and h_e for the function $F_e(h)$.

Every function $E : S \to \mathbb{T}(S, BS)$ induces an endofunctor $F_E : Set^S \to Set^S$: For all $s \in S$, $F_E(A)(s) = F_{E(s)}(A)$ and $F_E(h)(s) = F_{E(s)}(h)$.

Given an S-sorted relation $r \subseteq A \times B$, r is extended to an $\mathbb{T}(S, BS)$-sorted relation (also called **relation lifting**; see [29, 30]) inductively as follows:

Let $s \in S$, $e, e_1, \ldots, e_n \in \mathbb{T}(S, BS)$ and $X \in BS$.

$$
\begin{aligned}
r_X &= \Delta_X, \\
r_{e_1 \times \cdots \times e_n} &= \{((a_1, \ldots, a_n), (b_1, \ldots, b_n)) \mid \forall\, 1 \le i \le n : (a_i, b_i) \in r_{e_i}\}, \\
r_{e_1 + \cdots + e_n} &= \{((a, i), (b, i)) \mid (a, b) \in r_{e_i},\ 1 \le i \le n\}, \\
r_{word(e)} &= \{(a_1 \ldots a_n, b_1 \ldots b_n) \mid \forall\, 1 \le i \le n : (a_i, b_i) \in r_e,\ n \in \mathbb{N}\}, \\
r_{bag(e)} &= \{(f, g) \mid \exists\, h : supp(f) \xrightarrow{\sim} supp(g) : (a, h(a)) \in r_e \wedge f(a) = g(h(a))\}, \\
r_{set(e)} &= \{(f, g) \mid \exists\, h : supp(f) \xrightarrow{\sim} supp(g) : (a, h(a)) \in r_e \wedge f(a) = g(h(a))\}, \\
r_{e^X} &= \{(f, g) \mid \forall\, x \in X : (f(x), g(x)) \in r_e\}.
\end{aligned}
$$

Proposition

For all S-sorted sets A, $e \in \mathbb{T}(S, BS)$ and $a \in A_e$, $(a, a) \in \Delta_{A,e}$

Proof. Analogously to the proof of [30], Lemma 4.1.2, or the proposition on page 5 of [66]. ❑

Signatures

A **signature** $\Sigma = (S, BS, F, P)$ consists of

- a finite set S (of **sorts**),
- a finite set BS (of **base sets**),
- a (finite) set F of **function symbols** $f : e \to e'$,
- a (finite) set P of **predicates** $p : e$,

where $e, e' \in \mathbb{T}(S, BS)$.

Given $s \in S$, particular predicates are the binary **s-equality** $=_s: s \times s$ and the unary **s-membership** $\in_s: s$.

For all $f : e \to e' \in F$, $dom(f) = e$ is the **domain** of f and $ran(f) = e'$ is the **range** of f. For all $p : e \in P$, $dom(p) = e$ is the **domain** of p.

For all $s \in S$, $f : e \to s \in F$ is an **s-constructor** and $g : s \to e$ is an **s-destructor**.

Σ is **constructive** resp. **destructive** if F consists of constructors resp. destructors.

Σ is **polynomial** if for all $f : e \to e' \in F$, e and e' are polynomial.

Let $\Sigma' = (S', BS', F', P')$ be a further signature.

A **signature morphism** $\sigma : \Sigma \to \Sigma'$ is a quadruple of maps $\sigma_1 : S \cup BS \to \mathbb{T}(S', BS')$, $\sigma_2 : F \to F'$ and $\sigma_3 : P \to P'$ such that for all $f : e \to e' \in$ and $p : e \in P$, $\sigma_2(f) : \sigma_1^*(e) \to \sigma_1^*(e')$ and $\sigma_3(p) : \sigma_1^*(e)$, where $\sigma_1^*(e)$ denotes the type obtained from e by replacing $s \in S$ with $\sigma_1(s)$.

If σ is an inclusion, then Σ is a **subsignature** of Σ', i.e., $S \subseteq S'$, $BS \subseteq BS'$, $F \subseteq F'$ and $P \subseteq P'$.

Let X and Y be sets.

Constructive signatures

- *Nat* ⇝ natural numbers
  $S = \{nat\}$, $BS = \{1\}$, $F = \{0 : 1 \to nat,\ succ : nat \to nat\}$.

- *Reg*(X) ⇝ regular operators
  $S = \{reg\}$, $BS = \{1, X\}$,
  $$F = \{\ \emptyset,\ \epsilon : 1 \to reg,\ \_ : X \to reg,\ \_|\_,\ \_\cdot\_ : reg \times reg \to reg,\ star : reg \to reg\ \}.$$

- *List*(X) ⇝ finite sequences of elements of X
  $S = \{list\}$, $BS = \{1, X\}$, $F = \{nil : 1 \to list,\ cons : X \times list \to list\}$.

- *Bintree*(X) ⇝ binary trees of finite depth with node labels from X
  $S = \{btree\}$, $BS = \{1, X\}$,
  $F = \{empty : 1 \to btree,\ bjoin : btree \times X \times btree \to btree\}$.

- $Tree(X, Y)$ ⇝ finitely branching trees of finite depth with node labels from X and edge labels from Y
  $S = \{tree, trees\}$, $BS = \{1, X, Y\}$,
  $$F = \{\ join : X \times trees \to tree,\ nil : 1 \to trees,\ cons : Y \times tree \times trees \to trees\ \}$$
  or: $S = \{tree\}$, $BS = \{X, Y\}$, $F = \{join : X \times word(Y \times tree) \to tree\}$.

- $BagTree(X, Y)$ ⇝ finitely branching unordered trees of finite depth with node labels from X and edge labels from Y
  $S = \{tree\}$, $BS = \{X, Y\}$, $F = \{join : X \times bag(Y \times tree) \to tree\}$.

- $FDTree(X, Y)$ ⇝ finitely or infinitely branching trees of finite depth with node labels from X and edge labels from Y
  $S = \{tree\}$, $BS = \{X, Y\}$,
  $F = \{join : X \times ((Y \times tree)^{\mathbb{N}} + word(Y \times tree)) \to tree\}$.

Destructive signatures

- *coNat* ↠ natural numbers with infinity
  $S = \{nat\}$, $BS = \{1\}$, $F = \{pred : nat \to 1 + nat\}$.
- *Stream*(X) ↠ infinite sequences of elements of X
  $S = \{list\}$, $BS = \{X\}$, $F = \{head : list \to X,\ tail : list \to list\}$.
- *coList*(X) ↠ finite or infinite sequences of elements of X $coList(1) \simeq coNat$
  $S = \{list\}$, $BS = \{1, X\}$, $F = \{split : list \to 1 + (X \times list)\}$.
- *Infbintree*(X) ↠ binary trees of infinite depth with node labels from X
  $S = \{btree\}$, $BS = \{X\}$, $F = \{root : btree \to X,\ left, right : btree \to btree\}$.
- *coBintree*(X) ↠ binary trees of finite or infinite depth with node labels from X
  $S = \{btree\}$, $BS = \{1, X\}$, $F = \{split : btree \to 1 + (btree \times X \times btree)\}$.

- *coTree*(X, Y) ⇝ finitely or infinitely branching trees of finite or infinite depth with node labels from X and edge labels from Y
  $S = \{tree, trees\}$, $BS = \{1, X, Y\}$,
  $$F = \{ \; root : tree \to X, \; subtrees : tree \to trees, \\ split : trees \to 1 + (Y \times tree \times trees) \; \}.$$
- *FBTree*(X, Y) ⇝ finitely branching trees of finite or infinite depth with node labels from X and edge labels from Y
  $S = \{tree\}$, $BS = \{X, Y\}$,
  $F = \{root : tree \to X, \; subtrees : tree \to word(Y \times tree)\}$.
- *DAut*(X, Y) ⇝ deterministic Moore automata $DAut(1, Y) \simeq Stream(Y)$
  $S = \{state\}$, $BS = \{X, Y\}$, $F = \{\delta : state \to state^X, \; \beta : state \to Y\}$.
- *NDAut*(X, Y) ⇝ non-deterministic Moore automata, image finite labelled transition systems
  $S = \{state\}$, $BS = \{X, Y\}$, $F = \{\delta : state \to set(state)^X, \; \beta : state \to Y\}$.

- XML documents
  ➾ finitely branching trees of finite or infinite depth with one of n *element types* $s_1, \dots, s_n$ such that each tree t with element type s_i has a node label from X_i and a tuple of subtrees of type $s'_i = s_{i1} + \dots + s_{in_i}$, i.e., for all $1 \leq i \leq n$ and $1 \leq j \leq n_i$ there are $s_{ij1}, \dots, s_{ijn_{ij}} \in S \cup BS$ with $s_{ij} = s_{ij1} \times \ldots \times s_{ijn_{ij}}$

$$
\begin{aligned}
S &= \{s_1, \dots, s_n\} \cup \\
&\quad \{s_{ij1} \times \ldots \times s_{ijn_{ij}} \mid 1 \leq i \leq n,\ 1 \leq j \leq n_i,\ 1 \leq k \leq n_{ij}\}, \\
BS &= \{1, X_1, \dots, X_n\}, \\
F &= \{attributes_i : s_i \to X_i \mid 1 \leq i \leq n\} \ \cup\ \{subtrees_i : s_i \to s'_i \mid 1 \leq i \leq n\} \cup \\
&\quad \{\pi_{ijk} : s_{ij} \to s_{ijk} \mid 1 \leq i \leq n,\ 1 \leq j \leq n_i,\ 1 \leq k \leq n_{ij}\}.
\end{aligned}
$$

  Trees of infinite depth may result from unfolding XML documents by resolving its link attributes.
  Analogously, one may formalize object class diagrams, e.g. those developed as part of an UML design.

Σ-algebras

Let $\Sigma = (S, BS, F, P)$ be a signature.

A **Σ-algebra** A consists of

- for each $s \in S$, a set A_s, the **carrier** of A,
- for each $f : e \rightarrow e' \in F$, a function $f^A : A_e \rightarrow A_{e'}$,
- for each $p : e \in P$, a subset p^A of A_e.

Hence A is an **S-sorted set**, the **carrier** of A, together with interpretations of F and P.

Examples

The regular expressions over X form the reg-carrier of the $Reg(X)$-algebra $T_{Reg(X)}$ of ground $Reg(X)$-terms.

The usual interpretation of regular expressions over X as languages (= sets of words) over X yields the $Reg(X)$-algebra *Lang*:

$Lang_{reg} = \mathcal{P}(X^*)$. For all $x \in X$ and $L, L' \in \mathcal{P}(X^*)$,

$$\begin{array}{l} \emptyset^{Lang} = \emptyset, \ \epsilon^{Lang} = \{\epsilon\}, \ \_^{Lang}(x) = \{x\}, \\ L|^{Lang}L' = L \cup L', \ L \cdot^{Lang} L' = \{vw \mid v \in L, \ w \in L'\}, \\ star^{Lang}(L) = \{w_1 \dots w_n \mid n \in \mathbb{N}, \ \forall\, 1 \leq i \leq n : w_i \in L\}. \end{array}$$

The $Reg(X)$-Algebra *Bool* interprets the regular operators as Boolean functions:

$Bool_{reg} = 2$. For all $x \in X$ and $b, b' \in 2$,

$$\begin{array}{l} \emptyset^{Bool} = 0, \ \epsilon^{Bool} = 1, \ \_^{Bool}(x) = 0, \\ b|^{Bool}b' = b \vee b', \ b \cdot^{Bool} b' = b \wedge b', \ star^{Bool}(b) = 1. \end{array}$$ ❏

Let A and B be Σ-algebras, $h : A \to B$ be an S-sorted function.

h is **compatible** with $f : e \to e' \in F$ if $h_{e'} \circ f^A = f^B \circ h_e$.

h is **compatible** with $p : e \in P$ if $h_e(p^A) \subseteq p^B$.

h is **cocompatible** with $p : e \in P$ if $h_e(A_e \setminus p^A) \subseteq B_e \setminus p^B$.

h **reflects predicates** if for all $p : e \in P$, $p^B \subseteq h_e(p^A)$.

h is a Σ**-homomorphism** or Σ**-homomorphic** if for all $f \in F \cup P$, h is compatible with f.

h is a Σ**-cohomomorphism** or Σ**-cohomomorphic** if for all $f \in F$, h is compatible with f, and for all $p \in P$, h is cocompatible with p.

Alg_Σ denotes the category of Σ-algebras and Σ-homomorphisms.

h is a Σ**-isomorphism** if h is iso in Alg_Σ.

For all Σ-homomorphisms h,

h is epi in Alg_Σ iff h is surjective.
h is mono in Alg_Σ iff h is injective.
h is iso in Alg_Σ iff h is bijective and reflects predicates.

Lemma EMH

Let $g : A \to B$ and $h : B \to C$ be S-sorted functions such that $h \circ g$ is a Σ-homomorphism.

(1) If g is epi in Alg_Σ and reflects predicates, then h is Σ-homomorphic.

(2) If h is mono in Alg_Σ and reflects predicates, then g is Σ-homomorphic.

Proof. (1) Compatibility of h with all $f \in F$ can be shown by diagram chasing. Moreover, for all $p : e \in P$, $p^B \subseteq g_e(p^A)$ implies $h_e(p^B) \subseteq h_e(g_e(p^A)) \subseteq p^A$ because $h \circ g$ is homomorphic.

(2) Compatibility of g with all $f \in F$ can be shown by diagram chasing. Moreover, for all $p : e \in P$, $p^C \subseteq h_e(p^B)$ implies $h_e(g_e(p^A)) \subseteq p^C \subseteq h_e(p^B)$ and thus $g_e(p^A) \subseteq p^B$ because $h \circ g$ is homomorphic and h is injective. ❑

let U_S be the forgetful functor from Alg_Σ to Set^S.

For all $f : e \to e' \in F$,

$$\{f^A : A_e \to A_{e'} \mid A \in Alg_\Sigma\}$$

is a natural transformation from $F_e U_S$ to $F_{e'} U_S$ because morphisms in Alg_Σ are Σ-homomorphisms.

Conversely, we use a notion introduced in [54, 34] and call every natural transformation from $F_e U_S$ to $F_{e'} U_S$ an **(implicit) Σ-operation** of type $e \to e'$. We write $t : e \to e'$ and denote the set of Σ-operations by Op_Σ.

In particular, given base sets X and Y, any function $f : X \to Y$ is a Σ-operation of type $X \to Y$ because for all $A \in Alg_\Sigma$, $F_X(U_S(A)) = X$ and $F_Y(U_S(A)) = Y$.

Σ-formulas

Let V be a set of **variables**. The set Fo_Σ of **Σ-formulas** is inductively defined as follows:

$$
\begin{array}{lcll}
p \in P & \Rightarrow & p \in Fo_\Sigma, & \\
t : e \to e' \in Op_\Sigma,\ p : e' \in P \cup BS & \Rightarrow & pt : e \in Fo_\Sigma, & \textbf{Σ-atoms} \\
\varphi : e, \psi : e \in Fo_\Sigma & \Rightarrow & \neg\varphi : e,\ \varphi \wedge \psi : e,\ \varphi \vee \psi : e,\ \varphi \Rightarrow \psi : e, & \\
 & & \varphi \Leftarrow \psi : e,\ \varphi \Leftrightarrow \psi : e \in Fo_\Sigma, & \\
e = \prod_{x \in V} e_x,\ \varphi : e \in Fo_\Sigma,\ x \in V & \Rightarrow & \forall x\varphi : e,\ \exists x\varphi : e \in Fo_\Sigma. &
\end{array}
$$

A Σ-algebra A interprets a Σ-formula $\varphi : e \in Fo_\Sigma$ by the set of its solutions, i.e., $\varphi^A \subseteq A_e$ is inductively defined as follows:

For all $p : e' \in P \cup BS$ and $t : e \to e' \in Op_\Sigma$, $\varphi, \psi : e \in Fo_\Sigma$ and $x \in V$,

$$\begin{array}{lcl}
(pt)^A & = & \{a \in A_e \mid t^A(a) \in p^A\}, \\
(\neg\varphi)^A & = & A_e \setminus \varphi^A, \\
(\varphi \wedge \psi)^A & = & \varphi^A \cap \psi^A, \\
(\varphi \vee \psi)^A & = & \varphi^A \cup \psi^A, \\
(\varphi \Rightarrow \psi)^A & = & (\psi \Leftarrow \varphi)^A = (\neg\varphi \vee \psi)^A, \\
(\psi \Leftrightarrow \varphi)^A & = & (\varphi \Rightarrow \psi)^A \cap (\varphi \Leftarrow \psi)^A, \\
(\forall x \varphi)^A & = & \{a \in A_e \mid \forall\, b \in A_{e_x} : a[b/x] \in \varphi^A\} \;\text{ if } e = \prod_{x \in V} e_x, \\
(\exists x \varphi)^A & = & \{a \in A_e \mid \exists\, b \in A_{e_x} : a[b/x] \in \varphi^A\} \;\text{ if } e = \prod_{x \in V} e_x.
\end{array}$$

Lemma NEGFREE

Let φ be a negation-free Σ-formula.

(1) For all Σ-homomorphisms $h : A \to B$, $h(\varphi^A) \subseteq \varphi^B$.

(2) For all Σ-cohomomorphisms $h : A \to B$, $h((\neg\varphi)^A) \subseteq (\neg\varphi)^B$. ❑

A **satisfies** $\varphi : e \in Fo_\Sigma$, written as $A \models \varphi$, if $\varphi^A = A_e$.

Given a set AX of Σ-formulas, A is a (Σ, AX)**-algebra** if A satisfies (all formulas of) AX.

$Alg_{\Sigma,AX}$ denotes the full subcategory of Alg_Σ whose objects are all (Σ, AX)-algebras.

Let $\sigma : \Sigma \to \Sigma'$ be a signature morphism, A be a Σ'-algebra and $h : A \to B$ be a Σ'-homomorphism.

The σ**-reduct of** A, $A|_\sigma$, is the Σ-algebra defined as follows:

- For all $s \in S$, $(A|_\sigma)_s = F_{\sigma(s)}(A)$.
- For all $f \in F \cup P$, $f^{A|_\sigma} = \sigma(f)^A$.

The σ**-reduct of** h, $h|_\sigma$, is the Σ-homomorphism defined as follows:

- For all $s \in S$, $(h|_\sigma)_s = h_{\sigma(s)}$.

σ-reducts are the images of the **reduct functor** $\_|_\sigma$ from $Alg_{\Sigma'}$ to Alg_Σ.

Let Σ be a subsignature of Σ', A be a Σ'-algebra and $h : A \longrightarrow B$ be a Σ'-homomorphism.

The **Σ-reduct** $A|_\Sigma$ **of** A is the Σ-algebra defined as follows:

- For all $s \in S$, $(A|_\Sigma)_s = A_s$.
- For all $f \in F \cup P$, $f^{A|_\Sigma} = f^A$.

The **Σ-reduct** $h|_\Sigma$ **of** h is the Σ-homomorphism defined as follows:

- For all $s \in S$, $(h|_\Sigma)_s = h_{\Sigma(s)}$.

Σ-reducts are the images of the **forgetful functor** U_Σ from $Alg_{\Sigma'}$ to Alg_Σ.

An **institution** (see [22]) consists of

- a category $Sign$ of signatures,
- a functor
$$\begin{array}{rcl} Sen : Sign & \rightarrow & Set \\ \Sigma & \mapsto & \text{set of } \Sigma\text{-sentences} \\ \sigma : \Sigma \rightarrow \Sigma' & \mapsto & Sen(\sigma) : Sen(\Sigma) \rightarrow Sen(\Sigma'), \end{array}$$
- a functor
$$\begin{array}{rcl} Mod : Sign^{op} & \rightarrow & Set \\ \Sigma & \mapsto & \text{set of } \Sigma\text{-models} \\ \sigma : \Sigma \rightarrow \Sigma' & \mapsto & Mod(\sigma) : Mod(\Sigma') \rightarrow Mod(\Sigma), \end{array}$$
- for each $\Sigma \in Sign$, a satisfaction relation
$$\models_\Sigma \subseteq Mod(\Sigma) \times Sen(\Sigma)$$
such that for all $Sign$-morphisms $\sigma : \Sigma \rightarrow \Sigma'$, $A \in Mod(\Sigma')$ and $\varphi \in Sen(\Sigma)$.
$$Mod(\sigma)(A) \models_\Sigma \varphi \iff A \models_{\Sigma'} Sen(\sigma)(\varphi). \tag{1}$$

Suppose that

- $Sign$ is the category of signatures and signature morphisms as defined above,
- for all signatures Σ, $Sen(\Sigma)$ is the set of Σ-formulas over a fixed set of co/variables,
- for all signature morphisms $\sigma : \Sigma \to \Sigma'$ and Σ-formulas φ, $Sen(\sigma)$ maps φ to $\sigma(\varphi)$ where $\sigma(\varphi)$ is obtained from φ by replacing all function symbols or predicates of Σ by their σ-images,
- for all signatures Σ, $Mod(\Sigma) = Alg_\Sigma$,
- for all signature morphisms $\sigma : \Sigma \to \Sigma'$ and Σ'-algebras A, $Mod(\sigma)$ maps A to $A|_\sigma$,
- $\models$ is the satisfaction relation defined above.

$(Sign, Sen, Mod, \models)$ is an institution.

Proof. (1) amounts to:

$$A|_\sigma \models_\Sigma \varphi \iff A \models_{\Sigma'} \sigma(\varphi). \tag{2}$$

The proof of (2) is straightforward (induction on the size of φ). ❏

Horn and co-Horn clauses

Let $\Sigma = (S, BS, F, P)$ and $\Sigma' = (S, BS, F, P + P')$ be signatures and C be a Σ-algebra.

$Alg_{\Sigma',C}$ denotes the full subcategory of Alg_Σ consisting of all Σ'-algebras A with $A|_\Sigma = C$.

$Alg_{\Sigma',C}$ is a complete lattice with the following partial order, suprema and infima:

For all $A, B \in Alg_{\Sigma',C}$,

$$A \leq B \iff \forall\, p \in P : p^A \subseteq p^B.$$

For all $\mathcal{A} \subseteq Alg_{\Sigma',C}$ and $p : e \in P$,

$$p^\perp = \emptyset, \quad p^\top = A_e, \quad p^{\sqcup\mathcal{A}} = \bigcup_{A \in \mathcal{A}} p^A \;\text{ and }\; p^{\sqcap\mathcal{A}} = \bigcap_{A \in \mathcal{A}} p^A.$$

Given a set AX of Σ'-formulas, $Alg_{\Sigma',AX}$ denotes the category of all Σ-algebras A that satisfy AX.

$Alg_{\Sigma',C,AX} = Alg_{\Sigma',AX} \cap Alg_{\Sigma',C}$.

A **Horn clause for** $p \in P'$ is a Σ'-formula of the form $pt \Leftarrow \varphi$ such that $\vee$, $\wedge$ and $\forall$ are the only logical operators of φ.

A **co-Horn clause for** $p \in P'$ is a Σ'-formulas of the form $pt \Rightarrow \varphi$ such that $\vee$, $\wedge$ and $\exists$ are the only logical operators of φ.

Let $A, B \in Alg_{\Sigma',C}$ and $pt \Leftarrow \varphi$ resp. $pt \Rightarrow \varphi$ be a Horn resp. co-Horn clause. Since φ is negation-free,

$$A \leq B \quad \text{implies} \quad \varphi^A \subseteq \varphi^B. \tag{3}$$

A Σ'-formula φ is **membership compatible** if for all subformulas $\exists x\psi : e$ and $\forall x\psi : e$ of φ there is a Σ'-formula ρ such that $\psi = (\in_{e_x}\pi_x \wedge \rho)$ or $\psi = (\in_{e_x}\pi_x \Rightarrow \rho)$, respectively.

A Σ'-formula φ is **finitely branching** if for all subformulas $\exists x\psi : e$ or $\forall x\psi : e$ of φ, $A \in Alg_{\Sigma',C}$ and $a \in A_e$, the set $\{b \in A_{e_x} \mid a[b/x] \in \psi^A\}$ is finite.

Lemma FB

Let φ be a finitely branching negation-free Σ'-formula.

(i) For all ω-chains $\{A_i \in Alg_{\Sigma',C} \mid i < \omega\}$ of $Alg_{\Sigma',C}$, $\varphi^{\sqcup_{i\in\mathbb{N}}A_i} \subseteq \bigcup_{i\in\mathbb{N}}\varphi^{A_i}$.

(ii) For all ω-cochains $\{A_i \in Alg_{\Sigma',C} \mid i < \omega\}$ of $Alg_{\Sigma',C}$, $\bigcap_{i\in\mathbb{N}}\varphi^{A_i} \subseteq \varphi^{\sqcap_{i\in\mathbb{N}}A_i}$.

Proof by induction on the size of φ. (i) For all Σ'-atoms $pt : e$,

$$(pt)^{\sqcup_{i\in\mathbb{N}}A_i} = \{a \in C_e \mid t^A(a) \in p^{\sqcup_{i\in\mathbb{N}}A_i}\} = \bigcup_{i\in\mathbb{N}}\{a \in C_e \mid t^A(a) \in p^{A_i}\} = \bigcup_{i\in\mathbb{N}}(pt)^{A_i}.$$

For all Σ'-formulas $\varphi, \psi : e$,

$$(\varphi \vee \psi)^{\sqcup_{i\in\mathbb{N}}A_i} = \varphi^{\sqcup_{i\in\mathbb{N}}A_i} \cup \psi^{\sqcup_{i\in\mathbb{N}}A_i} \overset{i.h.}{\subseteq} (\bigcup_{i\in\mathbb{N}}\varphi^{A_i}) \cup (\bigcup_{i\in\mathbb{N}}\psi^{A_i}) = \bigcup_{i\in\mathbb{N}}(\varphi^{A_i} \cup \psi^{A_i})$$
$$= \bigcup_{i\in\mathbb{N}}(\varphi \vee \psi)^{A_i},$$
$$(\varphi \wedge \psi)^{\sqcup_{i\in\mathbb{N}}A_i} = \varphi^{\sqcup_{i\in\mathbb{N}}A_i} \cap \psi^{\sqcup_{i\in\mathbb{N}}A_i} \overset{i.h.}{\subseteq} (\bigcup_{i\in\mathbb{N}}\varphi^{A_i}) \cap (\bigcup_{i\in\mathbb{N}}\psi^{A_i}) = \bigcup_{i,j\in\mathbb{N}}(\varphi^{A_i} \cap \psi^{A_j})$$
$$\subseteq \bigcup_{i,j\in\mathbb{N}}(\varphi^{A_{max(i,j)}} \cap \psi^{A_{max(i,j)}}) = \bigcup_{i\in\mathbb{N}}(\varphi^{A_i} \cap \psi^{A_i}) = \bigcup_{i\in\mathbb{N}}(\varphi \wedge \psi)^{A_i}.$$

For all Σ'-formulas $\varphi : e = \prod_{x\in V} e_x$ and $x \in V$,

$$(\exists x\varphi)^{\sqcup_{i\in\mathbb{N}}A_i} = \{a \in A_e \mid \exists\ b \in A_{e_x} : a[b/x] \in \varphi^{\sqcup_{i\in\mathbb{N}}A_i}\}$$
$$\stackrel{i.h.}{\subseteq} \{a \in A_e \mid \exists\ b \in A_{e_x} : a[b/x] \in \bigcup_{i\in\mathbb{N}} \varphi^{A_i}\} = \bigcup_{i\in\mathbb{N}}\{a \in A_e \mid \exists\ b \in A_{e_x} : a[b/x] \in \varphi^{A_i}\}$$
$$= \bigcup_{i\in\mathbb{N}}(\exists x\varphi)^{A_i},$$
$$(\forall x\varphi)^{\sqcup_{i\in\mathbb{N}}A_i} = \{a \in A_e \mid \forall\ b \in A_{e_x} : a[b/x] \in \varphi^{\sqcup_{i\in\mathbb{N}}A_i}\}$$
$$= (\forall x\varphi)^{\sqcup_{i\in\mathbb{N}}A_i} = \{a \in A_e \mid \forall\ b \in B_a : a[b_k/x] \in \varphi^{\sqcup_{i\in\mathbb{N}}A_i}\}$$
$$\stackrel{i.h.}{\subseteq} \{a \in A_e \mid \forall\ b \in B_a : a[b/x] \in \bigcup_{i\in\mathbb{N}} \varphi^{A_i}\} = \{a \in A_e \mid \forall\ b \in B_a : a[b/x] \in \varphi^{A_{n_a}}\}$$
$$= (\forall x\varphi)^{A_{n_a}} \subseteq \bigcup_{i\in\mathbb{N}}(\forall x\varphi)^{A_i}$$

where the finiteness of $B_a = \{b \in A_{e_x} \mid a[b/x] \in \varphi^{\sqcup_{i\in\mathbb{N}}A_i}\}$ and thus the existence of n_a with $\{a[b/x] \mid b \in B_a\} \subseteq \varphi^{A_{n_a}}$ follow from the assumption that $\forall x\varphi$ is finitely branching.

(ii) Analogously. ❏

For all $p \in P'$, let AX_p be a set of Horn clauses for p. Then $AX = \cup_{p \in P'} AX_p$ is a **Horn specification for** P' and the elements of P' are called **least predicates**.

The **step function** $\Phi = \Phi_{\Sigma',C,AX} : Alg_{\Sigma',C} \to Alg_{\Sigma',C}$ is defined as follows: For all $A \in Alg_{\Sigma',C}$ and $p : e \in P'$,

$$p^{\Phi(A)} \;=\; \{t^C(a) \mid pt \Leftarrow \varphi \in AX,\; a \in \varphi^A\}.$$

By (3), Φ is monotone and thus by the Fixpoint Theorem of Knaster and Tarski, Φ has the least fixpoint

$$lfp(\Phi) \;=\; \sqcap \{A \in Alg_{\Sigma',C} \mid \Phi(A) \leq A\}.$$

Lemma IND

$$Alg_{\Sigma',C,AX} = \{A \in Alg_{\Sigma',C} \mid \Phi(A) \leq A\} \tag{4}$$

and thus for all $A \in Alg_{\Sigma',C,AX}$,

$$lfp(\Phi) \leq A, \tag{5}$$

Moreover, if C is initial in Alg_{Σ}, then $lfp(\Phi)$ is initial in $Alg_{\Sigma',C,AX}$.

Proof. Let $A \in Alg_{\Sigma',C,AX}$ and $b \in p^{\Phi(A)}$. Then $b = t^A(a)$ for some $pt \Leftarrow \varphi \in AX$ and $a \in \varphi^A$. Since A satisfies $pt \Leftarrow \varphi$, $a \in (pt)^A$ and thus $b = t^C(a) \in p^A$. Hence A is Φ-closed.

Conversely, let A be Φ-closed, $pt \Leftarrow \varphi \in AX$ and $a \in \varphi^A$. Then $t^C(a) \in p^{\Phi(A)}$. Since A is Φ-closed, $t^C(a) \in p^A$ and thus $a \in (pt)^A$. Hence A satisfies $pt \Leftarrow \varphi$.

The initiality of $lfp(\Phi)$ in $Alg_{\Sigma',C,AX}$ follows from the compatibility with P' of id_C as the unique Σ-homomorphism from $lfp(\Phi)$ to every $A \in Alg_{\Sigma',C,AX}$: For all $p \in P'$,

$$id_C(p^{lfp(\Phi)}) = p^{lfp(\Phi)} = \cap\{p^B \mid B \in Alg_{\Sigma',C,AX},\ \Phi(B) \leq B\} \subseteq p^A. \qquad \square$$

For all $p \in P'$, let AX_p be a set of co-Horn clauses for p. Then $AX = \cup_{p \in P'} AX_p$ is a **co-Horn specification for** P' and the elements of P' are called **greatest predicates**.

The **step function** $\Phi = \Phi_{\Sigma',C,AX} : Alg_{\Sigma',C} \to Alg_{\Sigma',C}$ is defined as follows: For all $A \in Alg_{\Sigma',C}$ and $p : e \in P'$,

$$p^{\Phi(A)} = C_e \setminus \{t^C(a) \mid pt \Rightarrow \varphi : e' \in AX,\ a \in C_{e'} \setminus \varphi^A\}.$$

By (3), Φ is monotone and thus by the Fixpoint Theorem of Knaster and Tarski, Φ has the greatest fixpoint

$$gfp(\Phi) \;=\; \sqcup\{A \in Alg_{\Sigma',C} \mid A \leq \Phi(A)\}.$$

Lemma COIND

$$Alg_{\Sigma',C,AX} = \{A \in Alg_{\Sigma',C} \mid A \leq \Phi(A)\} \tag{6}$$

and thus for all $A \in Alg_{\Sigma',C,AX}$,

$$A \leq gfp(\Phi). \tag{7}$$

Moreover, if C is final in Alg_Σ, then $gfp(\Phi)$ is final in $Alg_{\Sigma',C,AX}$.

Proof. Let $A \in Alg_{\Sigma',C,AX}$ and $b \notin p^{\Phi(A)}$. Then $b = t^C(a)$ for some $pt \Rightarrow \varphi \in AX$ and $a \notin \varphi^A$. Since A satisfies $pt \Rightarrow \varphi$, $a \notin (pt)^A$ and thus $b = t^C(a) \notin p^A$. Hence A is Φ-dense.

Conversely, let A be Φ-dense, $pt \Rightarrow \varphi \in AX$ and $a \notin \varphi^A$. Then $t^C(a) \notin p^{\Phi(A)}$. Since A is Φ-dense, $t^C(a) \notin p^A$ and thus $a \notin (pt)^A$. Hence A satisfies $pt \Rightarrow \varphi$.

The finality of $gfp(\Phi)$ in $Alg_{\Sigma',C,AX}$ follows from the compatibility with P' of id_C as the unique Σ-homomorphism from every $A \in Alg_{\Sigma',C,AX}$ to $gfp(\Phi)$: For all $p \in P'$,

$$id_C(p^A) = p^A \subseteq \cup\{p^B \mid B \in Alg_{\Sigma',C,AX},\ B \leq \Phi(B)\} = p^{gfp(\Phi)}. \qquad ❑.$$

Lemma MUPRED

Let C be a Σ-algebra, $\Sigma' = (S, BS, F, P+P')$ be a signature, AX be a Horn specification for P' and $A \in Alg_{\Sigma',AX}$ such that $\Phi = \Phi_{\Sigma',C,AX}$ is ω-continuous.

Every Σ-homomorphism $h : C \to B = A|_\Sigma$ is a Σ'-homomorphism from the (Σ', AX)-algebra $lfp(\Phi)$ to A.

In particular, if C is initial in Alg_Σ, then $lfp(\Phi)$ is initial in $Alg_{\Sigma',AX}$.

Proof. It remains to show that for all $p \in P'$,

$$h(p^{lfp(\Phi)}) \subseteq p^A. \tag{1}$$

Let $p : e \in P'$ and $a \in p^{lfp(\Phi)}$. Hence by Kleene's Fixpoint Theorem (1), $a \in p^{\Phi^i(\bot)}$ for some $i \in \mathbb{N}$. Since $p^\bot = \emptyset$, $i > 0$.

Case 1: $a \in p^{\Phi(\bot)}$. Then $a = t^C(c)$ for some $pt \Leftarrow \varphi : e' \in AX$ and $c \in \varphi^\bot$. Since $\varphi^\bot = \emptyset$, $\varphi = True$. Since A satisfies $pt \Leftarrow \varphi$,

$$B_{e'} = (pt \Leftarrow True)^A = (pt)^A = \{b \in B_{e'} \mid t^B(b) \in p^A\}.$$

Hence for all $b \in B_{e'}$, $t^B(b) \in p^A$, and thus $h(a) = h(t^C(c)) = t^B(h(c)) \in p^A$. We conclude (1).

Case 2: $a \in p^{\Phi^i(\perp)}$ for some $i > 1$. Then $a = t^C(c)$ for some $pt \Leftarrow \varphi : e' \in AX$ and $c \in \varphi^{\Phi^{i-1}(\perp)}$. By induction hypothesis, h is a Σ'-homomorphism from $\Phi^{i-1}(\perp)$ to A. Hence by Lemma NEGFREE (1),

$$h(\varphi^{\Phi^{i-1}(\perp)}) \subseteq \varphi^A. \tag{2}$$

Since $c \in \varphi^{\Phi^{i-1}(\perp)}$, (2) implies $h(c) \in \varphi^A$. Since A satisfies $pt \Leftarrow \varphi$, $(pt \Leftarrow \varphi)^A = B_{e'}$. Hence $h(c) \in \varphi^A$ implies $h(c) \in (pt)^A = \{b \in B_{e'} \mid t^B(b) \in p^A\}$ and thus

$$h(a) = h(t^C(c)) = t^B(h(c)) \in p^A.$$

Again, we conclude (1). ❏

Lemma NUPRED

Let C be a Σ-algebra, $\Sigma' = (S, BS, F, P + P')$ be a signature, AX be a co-Horn specification for P' and $A \in Alg_{\Sigma',AX}$ such that $\Phi = \Phi_{\Sigma',C,AX}$ is ω-cocontinuous.

Every Σ-homomorphism $h : C \to B = A|_\Sigma$ is a Σ'-cohomomorphism from the (Σ', AX)-algebra $gfp(\Phi)$ to A.

Proof. It remains to show that for all $p \in P'$,

$$h(C_e \setminus p^{gfp(\Phi)}) \subseteq B_e \setminus p^A. \tag{1}$$

Let $p : e \in P'$ and $a \in C_e \setminus p^{gfp(\Phi)}$. Hence by Kleene's Fixpoint Theorem (2), $a \in C_e \setminus p^{\Phi^i(\top)}$ for some $i \in \mathbb{N}$. Since $p^\top = C_e$, $i > 0$.

Case 1: $a \in C_e \setminus p^{\Phi(\top)}$. Then $a = t^C(c)$ for some $pt \Rightarrow \varphi : e' \in AX$ and $c \in C_{e'} \setminus \varphi^\top$. Since $\varphi^\top = C_{e'}$, $\varphi = False$. Since A satisfies $pt \Rightarrow \varphi$,

$$B_{e'} = (pt \Rightarrow \varphi)^A = (\neg pt)^A = B_{e'} \setminus \{b \in B_{e'} \mid t^B(b) \in p^A\}.$$

Hence for all $b \in B_{e'}$, $t^B(b) \notin p^A$, and thus $h(a) = h(t^C(c)) = t^B(h(c)) \notin p^A$. We conclude (1).

Case 2: $a \in C_e \setminus p^{\Phi^i(\top)}$ for some $i > 1$. Then $a = t^C(c)$ for some $pt \Rightarrow \varphi : e' \in AX$ and $c \notin \varphi^{\Phi^{i-1}(\top)}$. By induction hypothesis, h is a Σ'-cohomomorphism from A to $\Phi^{i-1}(\top)$. Hence by Lemma NEGFREE (2),

$$h(C_e \setminus \varphi^{\Phi^{i-1}(\top)}) = h((\neg\varphi)^{\Phi^{i-1}(\top)}) \subseteq (\neg\varphi)^A = B_e \setminus \varphi^A. \tag{2}$$

Since $c \notin \varphi^{\Phi^{i-1}(\top)}$, (2) implies $h(c) \notin \varphi^A$. Since A satisfies $pt \Rightarrow \varphi$, $(pt \Rightarrow \varphi)^A = B_{e'}$. Hence $h(c) \notin \varphi^A$ implies $h(c) \notin (pt)^A = \{b \in B_{e'} \mid t^B(b) \notin p^A\}$ and thus

$$h(a) = h(t^C(c)) = t^B(h(c)) \notin p^A.$$

Again, we conclude (1). ❏

A Horn specification is **finitely branching** if the premises of all Horn clauses of AX are finitely branching.

A co-Horn specification is **finitely branching** if the conclusions of all co-Horn clauses of AX are finitely branching.

Theorem CONSTEP

(i) Let AX be a finitely branching Horn specification. Then $\Phi = \Phi_{\Sigma',C,AX}$ is ω-continuous.

(ii) Let AX be a finitely branching co-Horn specification. Then $\Phi = \Phi_{\Sigma',C,AX}$ is ω-cocontinuous.

Proof. (i) Let $\{A_i \in Alg_{\Sigma',C} \mid i < \omega\}$ be an ω-chain of $Alg_{\Sigma',C}$. Since Φ is monotone, it remains to show:

$$\Phi(\sqcup_{i\in\mathbb{N}} A_i) \leq \sqcup_{i\in\mathbb{N}} \Phi(A_i). \tag{8}$$

Let $p \in P'$. Then by Lemma FB,

$$p^{\Phi(\sqcup_{i\in\mathbb{N}} A_i)} = \{t^C(a) \mid pt \Leftarrow \varphi \in AX,\ a \in \varphi^{\sqcup_{i\in\mathbb{N}} A_i}\}$$
$$\subseteq \{t^C(a) \mid pt \Leftarrow \varphi \in AX,\ a \in \bigcup_{i\in\mathbb{N}} \varphi^{A_i}\} = \bigcup_{i\in\mathbb{N}}\{t^C(a) \mid pt \Leftarrow \varphi \in AX,\ a \in \varphi^{A_i}\}$$
$$= \bigcup_{i\in\mathbb{N}} p^{\Phi(A_i)} = p^{\sqcup_{i\in\mathbb{N}}\Phi(A_i)}.$$

Hence (8) holds true.

(ii) Let $\{A_i \in Alg_{\Sigma',C} \mid i < \omega\}$ be an ω-cochain of $Alg_{\Sigma',C}$. Since Φ is monotone, it remains to show:

$$\sqcap_{i\in\mathbb{N}} \Phi(A_i) \leq \Phi(\sqcap_{i\in\mathbb{N}} A_i). \tag{9}$$

Let $p : e \in P'$. Then by Lemma FB,

$$p^{\sqcap_{i\in\mathbb{N}}\Phi(A_i)} = \bigcap_{i\in\mathbb{N}} p^{\Phi(A_i)} = \bigcap_{i\in\mathbb{N}}(C_e \setminus \{t^C(a) \mid pt \Rightarrow \varphi : e' \in AX,\ a \in C_{e'} \setminus \varphi^{A_i}\})$$
$$= C_e \setminus \bigcup_{i\in\mathbb{N}}\{t^C(a) \mid pt \Rightarrow \varphi : e' \in AX,\ a \in C_{e'} \setminus \varphi^{A_i}\}$$
$$= C_e \setminus \{t^C(a) \mid pt \Rightarrow \varphi : e' \in AX,\ a \in \bigcup_{i\in\mathbb{N}}(C_{e'} \setminus \varphi^{A_i})\}$$
$$= C_e \setminus \{t^C(a) \mid pt \Rightarrow \varphi : e' \in AX,\ a \in C_{e'} \setminus \bigcap_{i\in\mathbb{N}} \varphi^{A_i}\}$$
$$\subseteq C_e \setminus \{t^C(a) \mid pt \Rightarrow \varphi : e' \in AX,\ a \in C_{e'} \setminus \varphi^{\sqcap_{i\in\mathbb{N}} A_i}\} = p^{\Phi(\sqcap_{i\in\mathbb{N}} A_i)}.$$

Hence (9) holds true. ❑

Theorem COMPLAX

Let $coP' = \{\overline{p} : e \mid p : e \in P'\}$, $co\Sigma' = (S, F, P + coP')$ and

$$coAX = \begin{cases} \{\overline{p}t \Rightarrow \overline{\varphi} \mid pt \Leftarrow \varphi \in AX\} & \text{if } AX \text{ is a Horn specification,} \\ \{\overline{p}t \Leftarrow \overline{\varphi} \mid pt \Rightarrow \varphi \in AX\} & \text{if } AX \text{ is a co-Horn specification} \end{cases}$$

where the formula $\overline{\varphi}$ is obtained from $\neg\varphi$ by moving $\neg$ to the atoms of φ and replacing each literal $\neg pt$, $p \in P'$, of the resulting formula with $\overline{p}t$.

Let C be a Σ-algebra, $\Phi = \Phi_{\Sigma',C,AX}$ and $\Psi = \Phi_{co\Sigma',C,coAX}$.

(1) Let AX be a finitely branching Horn specification. Then $coAX$ is a finitely branching co-Horn specification and for all $p : e \in P$,

$$\overline{p}^{gfp(\Psi)} = C_e \setminus p^{lfp(\Phi)}.$$

(2) Let AX be a finitely branching co-Horn specification. Then $coAX$ is a finitely branching Horn specification and for all $p : e \in P$,

$$\overline{p}^{lfp(\Psi)} = C_e \setminus p^{gfp(\Phi)}.$$

Proof. (1) Suppose that for all negation-free Σ-formulas $\varphi : e$ and $i \in \mathbb{N}$,

$$\overline{\varphi}^{\Psi^i(\top)} = (\neg\varphi)^{\Phi^i(\bot)}. \tag{3}$$

By Theorem CONSTEP, Φ is ω-continuous and Ψ is ω-cocontinuous. Hence by Kleene's Fixpoint Theorem, (3) implies (1):

$$\overline{p}^{gfp(\Psi)} = \bigcap_{i\in\mathbb{N}} \overline{p}^{\Psi^i(\top)} = \bigcap_{i\in\mathbb{N}} (\neg p)^{\Phi^i(\bot)} = \bigcap_{i\in\mathbb{N}} (C_e \setminus p^{\Phi^i(\bot)}) = C_e \setminus \bigcup_{i\in\mathbb{N}} p^{\Phi^i(\bot)}$$
$$= C_e \setminus p^{lfp(\Phi)}.$$

It remains to show (3). Let $i = 0$. Then

$$\overline{p}^{\Psi^i(\top)} = \overline{p}^{\top} = C_e = C_e \setminus \emptyset = C_e \setminus p^{\bot} = (\neg p)^{\bot} = (\neg p)^{\Phi^i(\bot)}. \tag{4}$$

By induction on the size of φ, (3) follows from (4). Let $i > 0$. Then

$$\begin{aligned}
&\overline{p}^{\Psi^i(\top)} = C_e \setminus \{t^C(a) \mid \overline{p}t \Rightarrow \overline{\varphi} : e' \in coAX,\ a \in C_{e'} \setminus \overline{\varphi}^{\Psi^{i-1}(\top)}\} \\
&= C_e \setminus \{t^C(a) \mid pt \Leftarrow \varphi : e' \in AX,\ a \in C_{e'} \setminus \overline{\varphi}^{\Psi^{i-1}(\top)}\} \\
&\stackrel{i.h.}{=} C_e \setminus \{t^C(a) \mid pt \Leftarrow \varphi : e' \in AX,\ a \in C_{e'} \setminus (\neg\varphi)^{\Phi^{i-1}(\bot)}\} \\
&= C_e \setminus \{t^C(a) \mid pt \Leftarrow \varphi : e' \in AX,\ a \in C_{e'} \setminus (C_{e'} \setminus \varphi^{\Phi^{i-1}(\bot)})\} \\
&= C_e \setminus \{t^C(a) \mid pt \Leftarrow \varphi : e' \in AX,\ a \in \varphi^{\Phi^{i-1}(\bot)}\} = C_e \setminus p^{\Phi^i(\bot)}.
\end{aligned} \tag{5}$$

By induction on the size of φ, (3) follows from (5).

(2) Analogously. ❏

Co/Resolution and narrowing in $lfp(\Phi)$ resp. $gfp(\Phi)$

- **Resolution** Let $p \neq \to$ be a least predicate. AX_p is applied to an atom pt:

$$\frac{pt}{\bigvee_{i=1}^{k} \exists Z_i : (\varphi_i\sigma_i \wedge \vec{x} = \vec{x}\sigma_i)} \Updownarrow$$

where $AX_p = \{\gamma_1 \Rightarrow (pt_1 \Longleftarrow \varphi_1), \dots, \gamma_n \Rightarrow (pt_n \Longleftarrow \varphi_n)\}$,

(∗) $\vec{x}$ is a list of the variables of t,
for all $1 \leq i \leq k$, $t\sigma_i = t_i\sigma_i$, $\gamma_i\sigma_i \vdash \mathit{True}$ and $Z_i = var(t_i, \varphi_i)$,
for all $k < i \leq n$, t is not unifiable with t_i.

- **Coresolution** Let p be a greatest predicate. AX_p is applied to a Σ-atom pt:

$$\frac{pt}{\bigwedge_{i=1}^{k} \forall Z_i : (\varphi_i\sigma_i \vee \vec{x} \neq \vec{x}\sigma_i)} \Updownarrow$$

where $AX_p = \{\gamma_1 \Rightarrow (pt_1 \Longrightarrow \varphi_1), \dots, \gamma_n \Rightarrow (pt_n \Longrightarrow \varphi_n)\}$ and (∗) holds true.

- **Deterministic narrowing**

  Let f be a defined function. AX_f is applied to a Σ-operation ft:

$$\frac{r(\dots,ft,\dots)}{\bigvee_{i=1}^{k}\exists Z_i : (r(\dots,u_i,\dots)\sigma_i \wedge \varphi_i\sigma_i \wedge \vec{x}=\vec{x}\sigma_i) \vee \\ \bigvee_{i=k+1}^{l}(r(\dots,ft,\dots)\sigma_i \wedge \vec{x}=\vec{x}\sigma_i)}$$

  where r is a predicate,

  $AX_f = \{\gamma_1 \Rightarrow (ft_1 = u_1 \Longleftarrow \varphi_1),\dots,\gamma_n \Rightarrow (ft_n = u_n \Longleftarrow \varphi_n)\}$,

  (**) $\vec{x}$ is a list of the variables of t,
  for all $1 \le i \le k$, $t\sigma_i = t_i\sigma_i$, $\gamma_i\sigma_i \vdash \mathit{True}$ and $Z_i = var(t_i,u_i,\varphi_i)$,
  for all $k < i \le l$, σ_i is a partial unifier of t and t_i,
  for all $l < i \le n$, t is not partially unifiable with t_i.

- **Nondeterministic narrowing**

  Let $\longrightarrow$ be a transition predicate. $AX_{\longrightarrow}$ is applied to an atom $t\ {}^\wedge v \longrightarrow t'$:

$$\frac{t\ {}^\wedge v \longrightarrow t'}{\begin{array}{c}\bigvee_{i=1}^{k} \exists Z_i : ((u_i\ {}^\wedge v)\sigma_i = t'\sigma_i \wedge \varphi_i\sigma_i \wedge \vec{x} = \vec{x}\sigma_i) \vee \\ \bigvee_{i=k+1}^{l}((t\ {}^\wedge v)\sigma_i \longrightarrow t'\sigma_i \wedge \vec{x} = \vec{x}\sigma_i)\end{array}}$$

  where $AX_{\longrightarrow} = \{\gamma_1 \Rightarrow (t_1 \longrightarrow u_1 \Longleftarrow \varphi_1), \dots, \gamma_n \Rightarrow (t_n \longrightarrow u_n \Longleftarrow \varphi_n)\}$, $(**)$ holds true and σ_i is a unifier *modulo associativity and commutativity of* ${}^\wedge$.

- **Elimination of irreducible atoms and operations** ("negation as failure")

$$\frac{pt}{False} \qquad \frac{qt}{True} \qquad \frac{r(\dots, ft, \dots)}{r(\dots, (), \dots)} \qquad \frac{t \longrightarrow t'}{() \longrightarrow t'}$$

  where $p \neq \longrightarrow$ is a least predicate, q is a greatest predicate, f is a defined function and pt, qt, ft and $t \longrightarrow t'$ are irreducible, i.e., none of the above rules is applicable.

Congruences and invariants

Let $\Sigma = (S, BS, F, P)$ be a signature, A be a Σ-algebra and $\sim$ be an S-sorted binary relation on A.

$\sim$ is **compatible** with $f : e \longrightarrow e' \in F$ if for all $a, b \in A_e$,

$$a \sim_e b \quad \text{implies} \quad f^A(a) \sim_{e'} f^A(b).$$

By the definition of relation lifting, $\sim$ is always compatible with f if $e \in BS$ because, in this case, $a \sim_e b$ implies $a = b$.

If $\sim$ is compatible with every $f \in F$, then $\sim$ is a **Σ-congruence on** A. If Σ is destructive, then a Σ-congruence is also called a **Σ-bisimulation** and the greatest one is called **Σ-bisimilarity**.

Let $\sim$ be a Σ-congruence on A.

$\sim^{eq}$ denotes the equivalence closure of $\sim$, which is also a Σ-congruence.

A_{eq} denotes the Σ-algebra that agrees with A except for the interpretation of all $p : e \in R$:

$$p^{A_{eq}} \;=\; \{a \in A_e \mid \exists\, b \in p^A : a \sim_e^{eq} b\}.$$

$nat_\sim : A \to A/{\sim}$ denotes the S-sorted **natural function** that maps $a \in A_s$ to $[a]_\sim = \{b \in A_s : a \sim_s^{eq} b\}$.

The **Σ-quotient of A by $\sim$**, $A/{\sim}$, is the Σ-algebra defined as follows:

- For all $s \in S$, $(A/{\sim})_s = \{[a]_\sim \mid a \in A_s\}$.
- For all $f : e \to e' \in F$ and $a \in A_e$, $f^{A/\sim}(nat_{\sim,e}(a)) =_{def} nat_{\sim,e'}(f^A(a))$.
- For all $p : e \in P$, $p^{A/\sim} =_{def} \{nat_{\sim,e}(a) \mid a \in p^{A_{eq}}\}$.

$nat_\sim : A \to A/{\sim}$ is epi in Alg_Σ.

Let $h : A \to B$ be an S-sorted function. The S-sorted binary relation

$$ker(h) \;=\; \{(a,b) \in A^2 \mid h(a) = h(b)\}$$

is called the **kernel of h.**

h is injective iff $ker(h) = \Delta_A$.

Lemma KER

(1) Let A be a Σ-algebra. B is a Σ-algebra and h is Σ-homomorphic iff $ker(h)$ is a Σ-congruence.

(2) h is Σ-homomorphic iff there is a unique Σ-monomorphism $h' : A/ker(h) \rightarrow B$ with $h' \circ nat_{ker(h)} = h$.

Proof. (1) If h is Σ-homomorphic, then $ker(h)$ is a Σ-congruence. Let $ker(h)$ be a Σ-congruence. For all $f : e \rightarrow e' \in F$, define $f^B : B_e \rightarrow B_{e'}$ such that for all $a \in A_e$, $f^B(h(a)) = h(f^A(a))$ and for all $p : e \in P$, define $p^B = h(p^A)$. Then B is a Σ-algebra and h is Σ-homomorphic.

(2) h' is defined by $h'([a]_{ker(h)}) = h(a)$ for all $a \in A$. Hence, if h is epi, then by Lemma EPIMON, h' is epi and thus $A/ker(h)$ and B are Σ-isomorphic. ❏

Lemma CONG

Let h be Σ-homomorphic and $\sim$ be a Σ-congruence on A. Then $\approx = \{(h(a), h(b)) \mid a \sim b\}$ is a Σ-congruence on B.

Proof. Let $f : e \to e' \in F$ and $c \approx_e d$. Then $c = h(a)$ and $d = h(b)$ for some $a, b \in A_e$ with $a \sim b$. Hence $f^A(a) \sim f^A(b)$. Since h is Σ-homomorphic, $f^B(c) = f^B(h(a)) = h(f^A(a))$ and $f^B(d) = f^B(h(b)) = h(f^A(b))$. Hence $f^B(c) \approx f^B(d)$. ❏

Lemma MIN

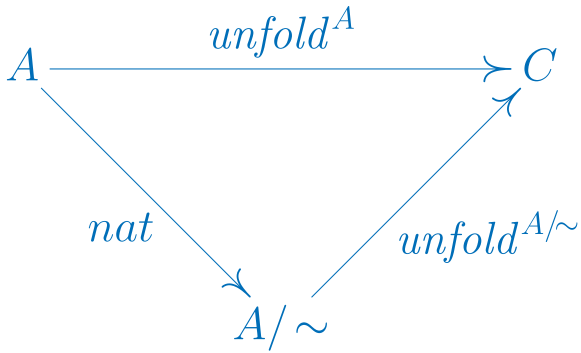
Let C be final in a full subcategory $\mathcal{K}$ of Alg_Σ.

(1) Δ_C is the only Σ-congruence on C.

(2) For all Σ-algebras A, $ker(unfold^A : A \to C)$ is the greatest Σ-congruence on A ([57], Prop. 2.7).

Proof. (1) A Σ-congruence $\sim$ on C induces the Σ-epimorphism $nat : C \to C/{\sim}$. Since C is final in $\mathcal{K}$, $unfold^{C/\sim} \circ nat = id_{C/\sim}$. Hence by Lemma EPIMON, nat is mono and thus iso in $\mathcal{K}$.

(2) Let $\sim$ be a Σ-congruence on A. Since C is final in $\mathcal{K}$, the following diagram commutes:



Hence for all $a, b \in A$,

$a \sim b \;\Rightarrow\; [a]_\sim = [b]_\sim \;\Rightarrow\; unfold^A(a) = unfold^{A/\sim}([a]_\sim) = unfold^{A/\sim}([b]_\sim) = unfold^A(b).$

We conclude that $ker(unfold^A)$ contains $\sim$.

Alternative proof of (2):

Let $\sim$ be a Σ-congruence on A. By Lemma CONG,

$$\approx \;=\; \{(unfold^A(a), unfold^A(b)) \mid a \sim b\}$$

is a Σ-congruence on C. By Lemma MIN (1), Δ_C is the only Σ-congruence on C. Hence $\approx = \Delta_C$ and thus for all $a, b \in A$, $a \sim b$ implies $unfold^A(a) = unfold^A(b)$, i.e., $(a, b) \in ker(unfold^A)$. ❏

Let $\sim$ be an S-sorted binary relation on A and $\sim^{eq}$ denote the equivalence closure of $\sim$.

$\sim$ is a **weak Σ-congruence** if for all $f : e \to e' \in F$ and $a, b \in A_e$, $a \sim b$ implies $f^A(a) \sim^{eq} f^A(b)$.

The equivalence closure $\sim^{eq}$ of a weak Σ-congruence $\sim$ is a Σ-congruence.

Proof by induction on the structure of $\sim^{eq}$. ❏

Let $\Sigma = (S, F, P)$ be a destructive signature and $\Sigma' = (S, F + F', P)$ be an extension of Σ such that F' consists of constructors. Let A be a Σ'-algebra and $\sim^{eq}_{F'}$ be the least S-sorted binary relation on A such that the following conditions hold true:

- $\sim \cup \Delta_A \subseteq \sim^{eq}_{F'}$.
- For all $a, b \in A$, $a \sim^{eq}_{F'} b$ implies $b \sim^{eq}_{F'} a$.
- For all $a, b, c \in A$, $a \sim^{eq}_{F'} b$ and $b \sim^{eq}_{F'} c$ imply $a \sim^{eq}_{F'} c$.
- For all $f : e \to s \in F'$ and $a, b \in A_e$, $a \sim^{eq}_{F'} b$ implies $f^A(a) \sim^{eq}_{F'} f^A(b)$.

$\sim$ is a **weak** (Σ, F')**-congruence** if for all $f : s \longrightarrow e \in F$ and $a, b \in A_s$, $a \sim b$ implies $f^A(a) \sim_{F'}^{eq} f^A(b)$.

Let $\sim$ be a weak (Σ, F')-congruence such that for all $f : e \longrightarrow s \in F'$, $g : s \longrightarrow e' \in F$ and $a, b \in A_e$,

$$F_e(g^A)(a) \sim_{F'}^{eq} F_e(g^A)(b) \quad \text{implies} \quad g^A(f^A(a)) \sim_{F'}^{eq} g^A(f^A(a)).$$

Then $\sim_{F'}^{eq}$ is a Σ-congruence.

Proof by induction on the structure of $\sim_{F'}^{eq}$. ❏

Lemma NAT

Let $e = \prod_{x \in V} e_x \in \mathbb{T}(S, BS)$, $\varphi : e \in Fo_\Sigma$, A be a Σ-algebra and $\sim$ be a Σ-congruence on A.

(1) For all Σ-operations $t : e \to e'$ and $a, b \in A_e$,

$$a \sim_e^{eq} b \quad \text{implies} \quad t^A(a) \sim_{e'}^{eq} t^A(b).$$

(2) For all $a, b \in A_e$, $a \sim_e^{eq} b$ and $a \in \varphi^{A_{eq}}$ imply $b \in \varphi^{A_{eq}}$.

(3) $\varphi^{A/\sim} = nat_{\sim,e}(\varphi^{A_{eq}})$.

(4) $A \models \varphi$ implies $A/\sim \ \models \varphi$.

Proof of (1). Let $a \sim_e^{eq} b$. Then

$$nat_{\sim,e'}(t^A(a)) = t^{A/\sim}(nat_{\sim,e}(a)) = t^{A/\sim}(nat_{\sim,e}(b)) = nat_{\sim,e'}(t^A(b)).$$

Hence $t^A(a) \sim_{e'}^{eq} t^A(a')$.

Proof of (2) by induction on the size of φ. Let $a \sim_e^{eq} b$.

Let $t : e \rightarrow e' \in Op_\Sigma$ and $p : e' \in P$. Since

$$(pt)^{A_{eq}} = \{c \in A_e \mid t^A(c) \in p^{A_{eq}}\} = \{c \in A_e \mid \exists\, a' \in p^A : t^A(c) \sim_{e'}^{eq} a'\}$$

and by (1), $t^A(a) \sim_{e'}^{eq} t^A(b)$, $a \in (pt)^{A_{eq}}$ implies $b \in (pt)^{A_{eq}}$.

Let $\varphi : e, \psi : e \in Fo_\Sigma$ and $x \in V$. Then

$$a \in (\neg\varphi)^{A_{eq}} \Leftrightarrow a \in A_e \setminus \varphi^{A_{eq}} \overset{i.h.}{\Rightarrow} b \in A_e \setminus \varphi^{A_{eq}} \Leftrightarrow b \in (\neg\varphi)^{A_{eq}},$$

$$a \in (\varphi \wedge \psi)^{A_{eq}} \Leftrightarrow a \in \varphi^{A_{eq}} \cap \varphi^{A_{eq}} \overset{i.h.}{\Rightarrow} b \in \varphi^{A_{eq}} \cap \varphi^{A_{eq}} \Leftrightarrow b \in (\varphi \wedge \psi)^{A_{eq}},$$

$$a \in (\forall x\varphi)^{A_{eq}} \Leftrightarrow \forall\, c \in A_{e_x} : a[c/x] \in \varphi^{A_{eq}} \overset{i.h.}{\Rightarrow} \forall\, c \in A_{e_x} : b[c/x] \in \varphi^{A_{eq}} \Leftrightarrow b \in (\forall x\varphi)^{A_{eq}}.$$

Proof of (3) by induction on the size of φ.

Let $t : e \to e' \in Op_\Sigma$ and $p : e' \in P$. Then

$$
\begin{aligned}
&(pt)^{A/\sim} = \{b \in (A/\sim)_e \mid t^{A/\sim}(b) \in p^{A/\sim}\}\\
&= \{b \in (A/\sim)_e \mid \exists\, a' \in p^{A_{eq}} : t^{A/\sim}(b) = nat_{\sim,e'}(a')\}\\
&= \{nat_{\sim,e}(a) \mid a \in A_e,\ \exists\, a' \in p^{A_{eq}} : t^{A/\sim}(nat_{\sim,e}(a)) = nat_{\sim,e'}(a')\}\\
&= \{nat_{\sim,e}(a) \mid a \in A_e,\ \exists\, a' \in p^{A_{eq}} : nat_{\sim,e'}(t^A(a)) = nat_{\sim,e'}(a')\}\\
&= \{nat_{\sim,e}(a) \mid a \in A_e,\ \exists\, a' \in p^{A_{eq}} : t^A(a) \sim^{eq}_{e'} a'\}\\
&= \{nat_{\sim,e}(a) \mid t^A(a) \in p^{A_{eq}}\} = \{nat_{\sim,e}(a) \mid a \in (pt)^{A_{eq}}\} = nat_{\sim,e}((pt)^{A_{eq}}).
\end{aligned}
$$

Let $\varphi, \psi : e \in Fo_\Sigma$ and $x \in V$. Then

$$(\neg\varphi)^{A/\sim} = (A/\sim)_e \setminus \varphi^{A/\sim} \overset{i.h.}{=} (A/\sim)_e \setminus nat_{\sim,e}(\varphi^{A_{eq}}) = nat_{\sim,e}(A_e \setminus \varphi^{A_{eq}})$$
$$= nat_{\sim,e}((\neg\varphi)^{A_{eq}}),$$
$$(\varphi \wedge \psi)^{A/\sim} = \varphi^{A/\sim} \cap \psi^{A/\sim} \overset{i.h.}{=} nat_{\sim,e}(\varphi^{A_{eq}}) \cap nat_{\sim,e}(\psi^{A_{eq}}) = nat_{\sim,e}(\varphi^{A_{eq}} \cap \psi^{A_{eq}})$$
$$= nat_{\sim,e}((\varphi \wedge \psi)^{A_{eq}}),$$
$$(\forall x\varphi)^{A/\sim} = \{b \in (A/\sim)_e \mid \forall\, d \in (A/\sim)_{e_x} : b[d/x] \in \varphi^{A/\sim}\}$$
$$= \{nat_{\sim,e}(a) \mid a \in A_e,\ \forall\, c \in A_{e_x} : nat_{\sim,e}(a)[nat_{\sim,e}(c)/x] \in \varphi^{A/\sim}\}$$
$$\overset{i.h.}{=} \{nat_{\sim,e}(a) \mid a \in A_e,\ \forall\, c \in A_{e_x} : a[c/x] \in \varphi^{A_{eq}}\}$$
$$= \{nat_{\sim,e}(a) \mid a \in A_e,\ a \in (\forall x\varphi)^{A_{eq}}\} = nat_{\sim,e}((\forall x\varphi)^{A_{eq}}).$$

Proof of (4). Let $A \models \varphi$. Then $\varphi^A = A_e$ and thus by (3),

$$\varphi^{A/\sim} = nat_{\sim,e}(\varphi^{A_{eq}}) = nat_{\sim,e}(A_e) = (A/\sim)_e,$$

i.e., $A/\sim \models \varphi$. ❑

Let $\Sigma = (S, F, P)$ be a signature, A be a Σ-algebra and inv be an S-sorted subset of A.

inv is **compatible with** $f : e \to e' \in F$ if for all $a \in A_e$,

$$a \in inv \quad \text{implies} \quad f^A(a) \in inv.$$

By the definition of predicate lifting, inv is always compatible with f if $e' \in BS$ because, in this case, $inv_{e'} = A_{e'} = e'$.

If inv is compatible with every $f \in F$, then inv is a **Σ-invariant** or **Σ-subalgebra of** A.

Given an S-sorted subset B of A, the least Σ-invariant including B is denoted by $\langle B \rangle$.

Let inv be a Σ-invariant of A.

$inc_{inv} : inv \to A$ denotes the S-sorted **inclusion** that maps each $a \in inv$ to a.

inv is extended to a Σ-algebra as follows:

- For all $f : e \to e' \in F$ and $a \in inv_e$, $\quad f^{inv}(a) =_{def} f^A(a)$.
- For all $p : e \in P$, $\quad p^{inv} =_{def} p^A \cap inv_e$.

$inc_{inv} : inv \to A$ is mono in Alg_Σ.

Let $h : A \to B$ be an S-sorted function.

The S-sorted subset $img(h) =_{def} \{h(a) \mid a \in A\}$ of A is called the **image of** h.

h is surjective iff $img(h) = B$.

Lemma IMG

(1) Let B be a Σ-algebra. A is a Σ-algebra and h is Σ-homomorphic iff $img(h)$ is a Σ-invariant.

(2) h is Σ-homomorphic iff there is a unique Σ-epimorphism $h' : A \to img(h)$ with $inc_{inv} \circ h' = h$.

Proof. (1) If h is Σ-homomorphic, then $img(h)$ is a Σ-invariant. Let $img(h)$ be a Σ-invariant. For all $f : e \to e' \in F$, define $f^A : A_e \to A_{e'}$ such that for all $a \in A_e$, $f^A(a) \in h^{-1}(f^B(h(a)))$, and for all $p \in P$, define $p^A = \{a \in A \mid h(a) \in p^B\}$. Then A is a Σ-algebra and h is Σ-homomorphic.

(2) h' is defined by $h'(a) = h(a)$ for all $a \in A$. Hence, if h is mono, then by Lemma EPIMON, h' is mono and thus A and $img(h)$ are Σ-isomorphic. ❑

Lemma INV

Let h be Σ-homomorphic and inv be a Σ-invariant of B. Then

$$inv_0 \;=\; \{a \in A \mid h(a) \in inv\}$$

is a Σ-invariant of A.

Proof.

Let $f : e \to e' \in F$ and $a \in inv_{0,e}$. Then $h(a) \in inv$ and thus $h(f^A(a)) = f^B(h(a)) \in inv$ because h is Σ-homomorphic and inv is a Σ-invariant. Hence $f^A(a) \in inv_0$. ❑

Lemma MAX

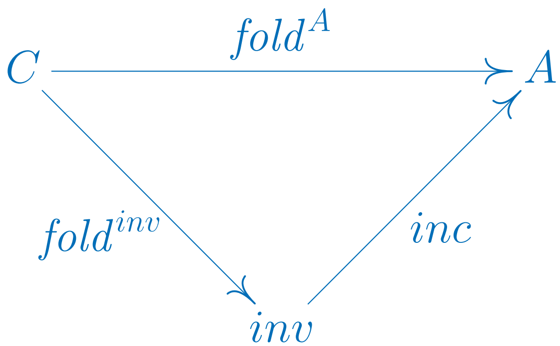
Let C be initial in a full subcategory $\mathcal{K}$ of Alg_Σ.

(1) C is the only Σ-invariant of C.

(2) For all Σ-algebras A, $img(fold^A : C \to A)$ is the least Σ-invariant of A.

Proof. (1) A Σ-invariant inv of C induces the Σ-monomorphism $inc : inv \to C$. Since C is initial in $\mathcal{K}$, $inc \circ fold^{inv} = id_C$. Hence by Lemma EPIMON, inc is epi and thus iso in $\mathcal{K}$.

(2) Let inv be a Σ-invariant of A. Since C is initial in $\mathcal{K}$, the following diagram commutes:



Alternative proof of (2):

Let inv be a Σ-invariant of A. By Lemma INV, $inv_0 = \{c \in C \mid fold^A(c) \in inv\}$ is a Σ-invariant of C. By Lemma MAX (1), C is the only Σ-invariant of C. Hence $inv_0 = C$. Let $a \in img(fold^A)$. Then there is $c \in C$ with $fold^A(c) = a$. Since $C = inv_0$, $c \in inv_0$ and thus $a = fold^A(c) \in inv$. ❏

Hence for all $a \in C$,

$$fold^A(a) = inc(fold^{inv}(a)) = fold^{inv}(a) \in inv.$$

We conclude that inv contains $img(fold^A)$. ❏

Lemma INC

Let $e = \prod_{x \in V} e_x \in \mathbb{T}(S, BS)$, $\varphi : e \in Fo_\Sigma$ be membership compatible, A be a Σ-algebra and inv be a Σ-invariant of A such that for all $s \in S$, $\in_s^A = inv_s$.

(1) For all $t : e \to e' \in Op_\Sigma$ and $a \in inv_e$, $t^{inv}(a) = t^A(a)$.

(2) $\varphi^{inv} = \varphi^A \cap inv_e$.

(3) $A \models \varphi$ implies $inv \models \varphi$.

Proof of (1). Let $a \in inv_e$. Then

$$t^{inv}(a) = F_{e'}(inc_{inv})(t^{inv}(a)) = t^A(F_e(inc_{inv})(a)) = t^A(a).$$

Proof of (2) by induction on the size of φ.

Let $t : e \to e' \in Op_\Sigma$ and $p : e' \in P$. Then by (1),

$$(pt)^{inv} = \{a \in inv_e \mid t^{inv}(a) \in p^A\} = \{a \in inv_e \mid t^A(a) \in p^A\}$$
$$= \{a \in A_e \mid t^A(a) \in p^A\} \cap inv_e.$$

Let $\varphi, \psi : e \in Fo_\Sigma$ and $x \in V$. Then

$$(\neg\varphi)^{inv} = inv_e \setminus \varphi^{inv} \overset{i.h.}{=} inv_e \setminus (\varphi^A \cap inv_e) = (A_e \setminus \varphi^A) \cap inv_e = (\neg\varphi)^A \cap inv_e,$$

$$(\varphi \wedge \psi)^{inv} = \varphi^{inv} \cap \psi^{inv} \overset{i.h.}{=} (\varphi^A \cap inv_e) \cap (\psi^A \cap inv_e) = (\varphi^A \cap \psi^A) \cap inv_e$$
$$= (\varphi \wedge \psi)^A \cap inv_e, \qquad (*)$$

$$(\varphi \vee \psi)^{inv} = (\neg(\neg\varphi \wedge \neg\psi))^{inv} \overset{(*)}{=} (\neg(\neg\varphi \wedge \neg\psi))^A \cap inv_e = \cdots = (\varphi \vee \psi)^A \cap inv_e.$$

Let $x \in V$, $\varphi : e \in Fo_\Sigma$ such that $\varphi = (\in_{e_x}\pi_x \wedge \psi)$ and $\varphi' = (\in_{e_x}\pi_x \Rightarrow \psi')$. Then

$$
\begin{aligned}
(\exists x\varphi)^{inv} &= \{a \in inv_e \mid \forall\, b \in inv_{e_x} : a[b/x] \in \varphi^{inv}\} \\
&\overset{i.h.}{=} \{a \in inv_e \mid \exists\, b \in inv_{e_x} : a[b/x] \in \varphi^A \cap inv_e\} \\
&= \{a \in inv_e \mid \exists\, b \in inv_{e_x} : a[b/x] \in \varphi^A\} \\
&= \{a \in A_e \mid \exists\, b \in inv_{e_x} : a[b/x] \in \varphi^A\} \cap inv_e \\
&= \{a \in A_e \mid \exists\, b \in A_{e_x} : b \in inv_{e_x} \wedge a[b/x] \in \varphi^A\} \cap inv_e \\
&= \{a \in A_e \mid \exists\, b \in A_{e_x} : b \in \in_{e_x}^A \wedge a[b/x] \in \varphi^A\} \cap inv_e \\
&= \{a \in A_e \mid \exists\, b \in A_{e_x} : a[b/x] \in (\in_{e_x}\pi_x)^A \wedge a[b/x] \in \varphi^A\} \cap inv_e \\
&= \{a \in A_e \mid \exists\, b \in A_{e_x} : a[b/x] \in (\in_{e_x}\pi_x \wedge \varphi)^A\} \cap inv_e \\
&= \{a \in A_e \mid \exists\, b \in A_{e_x} : a[b/x] \in (\in_{e_x}\pi_x \wedge \gamma_x\pi_x \wedge \gamma)^A\} \cap inv_e \\
&= \{a \in A_e \mid \exists\, b \in A_{e_x} : a[b/x] \in (\in_{e_x}\pi_x \wedge \gamma)^A\} \cap inv_e \\
&= \{a \in A_e \mid \exists\, b \in A_{e_x} : a[b/x] \in \varphi^A\} \cap inv_e \\
&= (\exists x\varphi)^A \cap inv_e, \qquad (**)
\end{aligned}
$$

$$(\forall x\varphi')^{inv} = (\neg\exists x\neg\varphi')^{inv} = (\neg(\exists x\neg(\gamma_x\pi_x \Rightarrow \gamma'))^{inv} = (\neg(\exists x\neg(\neg\gamma_x\pi_x \vee \gamma'))^{inv}$$
$$= (\neg(\exists x(\gamma_x\pi_x \wedge \neg\gamma'))^{inv} \stackrel{(**)}{=} (\neg(\exists x(\gamma_x\pi_x \wedge \neg\gamma'))^A \cap inv_e = \cdots = (\forall x\varphi')^A \cap inv_e.$$

Proof of (3). Let $A \models \varphi$. Then $\varphi^A = A_e$ and thus by (2),

$$\varphi^{inv} = \varphi^A \cap inv_e = A_e \cap inv_e = inv_e,$$

i.e., $inv \models \varphi$. ❑

Examples

Given a **behavior** function $f : X^* \to Y$, the **minimal realization** of f coincides with the invariant $\langle f\rangle$ of the $DAut(X,Y)$-algebra $Beh(X,Y)$: $Beh(X,Y)_{state} = (X^* \to Y)$; for all $f : X^* \to Y$ and $x \in X$, $\delta^{Beh(X,Y)}(f)(x) = \lambda w.f(xw)$ and $\beta^{Beh(X,Y)}(f) = f(\epsilon)$.

Let $Y = 2$. Then behaviors $f : X^* \to Y$ coincide with languages over X, i.e. subsets L of X^*: $Beh(X,2)_{state} = \mathcal{P}(X^*)$; for all $L \subseteq X^*$ and $x \in X$, $\delta^{Beh(X,2)}(L) = \{w \in X^* \mid xw \in L\}$ and $\beta^{Beh(X,2)}(L) = 1 \Leftrightarrow \epsilon \in L$.

Hence the *state*-carrier of $Beh(X,2)$ agrees with the *reg*-carrier of *Lang* and for all $L\subseteq X^*$, $\langle L\rangle$ is the minimal acceptor of L whose final states are the languages of $\langle L\rangle$ that contain ϵ.

$T_{Reg(X)}$ also provides acceptors of regular languages, i.e., $T=T_{Reg(X)}$ is a $DAut(X,2)$-algebra. Its transition function $\delta^T:T\to T^X$ is called the *Brzozowski derivative* [16, 36]. It has been shown that for all regular expressions R, $\langle R\rangle\subseteq T_{Reg(X)}$ has only finitely many states ([16], Thm. 4.3 (a); [56], Section 5; [32], Lemma 8).

If combined with coinductive proofs of state equivalence (see Section 4), the stepwise construction of the least invariant $\langle R\rangle$ of $T_{Reg(X)}$ can be lifted to a direct construction of the minimal acceptor $\langle L(R)\rangle$ of $L(R)$, thus avoiding the traditional detour from a given automaton, its determinization (powerset construction) and subsequent minimization (see [60], Section 4).

$Beh(1,Y)$ represents the algebra of streams with elements from Y:

$$Beh(1,Y)_{state}=Y^{1^*}\cong Y^{\mathbb{N}}.$$

For all $s\in Y^{\mathbb{N}}$, $\beta(s)=s(0)$ and $\delta(s)(*)=\lambda n.s(n+1)$.

$Beh(2,Y)$ represents the algebra of infinite binary trees with node labels from Y:

$$Beh(2,Y)_{state} = Y^{2^*}.$$

For all $t \in X^{2^*}$ and $b \in 2$, $\beta(t) = s(\epsilon)$, $\delta(t)(b) = \lambda w.t(bw)$.

A set A with addition and multiplication is a **semiring**, if A contains a zero and a one such that for all $a,b,c \in A$ the following equations hold true:

$a+(b+c) = (a+b)+c$	Assoziativität von $+$
$a+b = b+a$	Kommutativität von $+$
$0+a = a = a+0$	Neutralität von 0 bzgl. $+$
$a*(b*c) = (a*b)*c$	Assoziativität von $*$
$1*a = a = a*1$	Neutralität von 1 bzgl. $*$
$0*a = 0 = a*0$	Annihilierung durch 0
$a*(b+c) = (a*b)+(a*c)$	
$(a+b)*c = (a*c)+(b*c)$	Distribution von $*$ über $+$

A semiring A is a **ring** if, in addition, A has additive inverses, i.e., for all $a \in A$ there is a unique $a' \in A$ such that for $a+a' = 0$.

If Y is a semiring, then the elements of $Beh(X,Y)$ are called **power series** (see [57], Section 9). ❑

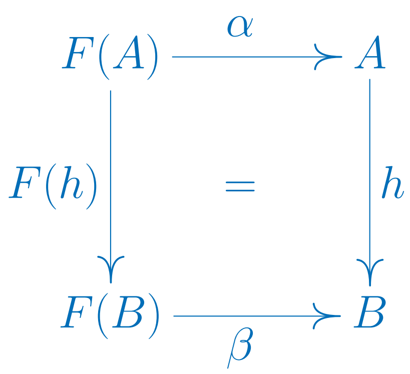
F-algebras and F-coalgebras

Let $\mathcal{K}$ be a category and $F : \mathcal{K} \to \mathcal{K}$ be a functor.

An F**-algebra** or F**-dynamics** is a $\mathcal{K}$-morphism $\alpha : F(A) \to A$.

Alg_F denotes the category of F-algebras.

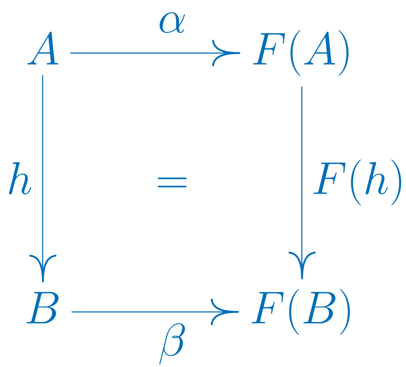
An Alg_F**-morphism** h from an F-algebra $\alpha\colon F(A) \to A$ to an F-algebra $\beta\colon F(B) \to B$ is a $\mathcal{K}$-morphism $h\colon A \to B$ with $h \circ \alpha = \beta \circ F(h)$.



An F-**coalgebra** or F-**codynamics** is a $\mathcal{K}$-morphism $\alpha : A \to F(A)$.

$coAlg_F$ denotes the category of F-coalgebras.

A $coAlg_F$-**morphism** h from an F-coalgebra $\alpha : A \to F(A)$ to an F-coalgebra $\beta : B \to F(B)$ is a $\mathcal{K}$-morphism $h : A \to B$ with $F(h) \circ \alpha = \beta \circ h$.



A $\mathcal{K}$-object A is a **fixpoint of** F if $F(A) \cong A$.

Lambek's Lemma ([39], Lemma 2.2; [14], Prop. 5.12; [6], Section 2; [55], Thm. 9.1)

(1) Suppose that Alg_F has an initial object $\alpha : F(\mu F) \to \mu F$.

α is iso and thus μF is a fixpoint of F.

(2) Suppose that $coAlg_F$ has a final object $\beta : \nu F \to F(\nu F)$.

β is iso and thus νF is a fixpoint of F.

Proof. (1) Since α is initial, there is a unique Alg_F-morphism $f : A \to F(A)$ from α to $F(\alpha)$. Hence $\alpha \circ f$ is an Alg_F-morphism from α to α:

$$\alpha \circ f \circ \alpha = \alpha \circ F(\alpha) \circ F(f) = \alpha \circ F(\alpha \circ f).$$

id_A is also an Alg_F-morphism from α nach α:

$$id_A \circ \alpha = \alpha = \alpha \circ id_{F(A)} = \alpha \circ F(id_A).$$

Hence (3) $id_A = \alpha \circ f$ because α is initial in Alg_F. Since f is an Alg_F-morphism,

$$f \circ \alpha = F(\alpha) \circ F(f) = F(\alpha \circ f) = F(id_A) = id_{F(A)}. \tag{4}$$

By (3) and (4), α is an isomorphism.

(2) Analogously. ❏

Given F-algebras $\alpha : F(A) \longrightarrow A$ and $\beta : F(B) \longrightarrow B$ such that α is initial in Alg_F, the unique Alg_F-morphism from α to β is called a **catamorphism** and denoted by $fold^B$.

Given F-coalgebras $\alpha : A \longrightarrow F(A)$ and $\beta : B \longrightarrow F(B)$ such that β is final in $coAlg_F$, the unique $coAlg_F$-morphism from α to β is called an **anamorphism** and denoted by $unfold^A$.

Co/complete categories and co/continuous functors

Let $\mathbb{O}$ be the category with ordinal numbers as objects and all pairs $(i, j) \in \mathbb{O}^2$ with $i \leq j$ as morphisms.

Let $\mathbb{O}_\lambda$ be the full subcategory of $\mathbb{O}$ with all ordinal numbers less than λ as objects.

A **chain** of $\mathcal{K}$ is a diagram $\mathcal{D} : \mathbb{O} \to \mathcal{K}$. A **cochain** of $\mathcal{K}$ is a diagram $\mathcal{D} : \mathbb{O} \to \mathcal{K}^{op}$.

Let λ be an ordinal number.

A **λ-chain** of $\mathcal{K}$ is a diagram $\mathcal{D} : \mathbb{O}_\lambda \to \mathcal{K}$. A **$\lambda$-cochain** of $\mathcal{K}$ is a diagram $\mathcal{D} : \mathbb{O}_\lambda \to \mathcal{K}^{op}$.

$\mathcal{K}$ is **λ-cocomplete** if $\mathcal{K}$ has an initial object and all λ-chains of $\mathcal{K}$ have colimits.

$\mathcal{K}$ is **λ-complete** if $\mathcal{K}$ has a final object and all λ-cochains of $\mathcal{K}$ have limits.

Set^S is λ-complete and λ-cocomplete.

Let $\mathcal{K}$ and $\mathcal{L}$ be λ-cocomplete. A functor $F : \mathcal{K} \to \mathcal{L}$ is **λ-cocontinuous** if for all λ-chains $\mathcal{D}$ of $\mathcal{K}$, F preserves the colimit $\{\mu_i : \mathcal{D}(i) \to C \mid i < \lambda\}$ of $\mathcal{D}$, i.e., $\{F(\mu_i) \mid i < \lambda\}$ is the colimit of $F \circ \mathcal{D}$.

Let $\mathcal{K}$ and $\mathcal{L}$ be λ-complete. A functor $F : \mathcal{K} \to \mathcal{L}$ is **λ-continuous** if for all λ-cochains $\mathcal{D}$ of $\mathcal{K}$, F preserves the limit $\{\nu_i : C \to \mathcal{D}(i) \mid i < \lambda\}$ of $\mathcal{D}$, i.e., $\{F(\nu_i) \mid i < \lambda\}$ is the limit of $F \circ \mathcal{D}$.

Given index sets I and J, a functor $F : Set^I \to Set^J$ is **permutative** if for all $A \in Set^I$ and $j \in J$ there is $i \in I$ such that $F(A)_j = A_i$.

Theorem CONTYPES

For all polynomial types e over S, $F_e : Set \to Set$ is ω-continuous.

Let e be a type over S, κ be the cardinality of the greatest (base set) exponent occurring in e and λ be the first regular cardinal number $> \kappa$. F_e is λ-cocontinuous.

Proof. By [8], Thms. 1 and 4, or [12], Prop. 2.2 (1) and (2), permutative and constant functors are ω-continuous and ω-cocontinuous, ω-continuous or λ-cocontinuous functors are closed under coproducts, ω-continuous functors are closed under products (and thus under exponentiation; see [55], Thm. 10.1) and λ-cocontinuous functors are closed under finite products.

By [12], Prop. 2.2 (3), ω-continuous or λ-cocontinuous functors are closed under quotients modulo finite equivalence relations. Since for all sets A, $A^* \cong \coprod_{n\in\mathbb{N}} A^n$ and $\mathcal{B}_{fin}(A) \cong \coprod_{n\in\mathbb{N}} A^n/{\sim_n}$ where $a \sim_n b$ iff a is a permutation of b, $\_^*$ and $\mathcal{B}_{fin}$ are ω-continuous and ω-cocontinuous (see [9], Exs. 2.3.14/15). By [9], Ex. 2.2.13, $\mathcal{P}_{fin}$ is ω-cocontinuous. For a proof of the fact that $\mathcal{P}_{fin}$ is not ω-continuous, see [9], Ex. 2.3.11.

Analogously to [9], Thm. 4.1.12, one may show that λ-cocontinuous functors are closed under exponentiation by exponents with a cardinality less than λ.

Moreover, ω-continuous or λ-cocontinuous functors are closed under sequential composition.

Putting all this together, we conclude that for all polynomial types over S and BS, $F_e : Set^S \to Set$ is ω-continuous, and for all $e \in \mathbb{T}(S, BS)$, F_e is λ-cocontinuous. ❑

CPO^E denotes the category of ω-CPOs as objects and pairs

$$(f : A \to B, g : B \to A)$$

of ω-continuous functions with $g \circ f = id_A$ and $f \circ g \leq id_B$ as morphisms.

Theorem CPOE (see, e.g., [48], Section 11.3)

All endofunctors on CPO^E built up from identity and constant functors, coproducts, finite products and Hom functors are cocontinuous. ❑

Construction of initial F-algebras and final F-coalgebras

Theorem LFIX (For $\lambda = \omega$, see [6], Section 2; [42], Thm. 2.1; for any λ, see [2], [3], Thm. 3.19, or [9], Cor. 4.1.5.)

Let λ be an infinite cardinal, Ini be initial in $\mathcal{K}$ and $\mathcal{K}$ be κ-cocomplete for all $\kappa \leq \lambda$.

Given a functor $F : \mathcal{K} \to \mathcal{K}$, define a λ-chain $\mathcal{D}$ of $\mathcal{K}$ as follows:

$$\begin{array}{rcll} \mathcal{D}(0) & = & Ini, & \\ \mathcal{D}(k+1) & = & F(\mathcal{D}(k)) & \text{for all } k < \lambda, \\ \mathcal{D}(k) & = & C_k & \text{for all limit ordinals } k < \lambda, \\ \mathcal{D}(i,k) & = & \mu_{i,k} & \text{for all limit ordinals } k < \lambda \text{ and all } i < k, \\ \mathcal{D}(k,k+1) & = & col_k & \text{for } k = 0 \text{ and all limit ordinals } k < \lambda, \\ \mathcal{D}(i+1,j+1) & = & F(\mathcal{D}(i,j)) & \text{for all } i \leq j < \lambda \end{array}$$

where $\gamma_k = \{\mu_{i,k} : \mathcal{D}(i) \to C_k \mid i < k\}$ is the colimit of the greatest subdiagram $\mathcal{D}_k : \mathbb{O}_k \to \mathcal{K}$ of $\mathcal{D}$ and col_k is the unique $\mathcal{K}$-morphism from C_k to $F(C_k)$ such that for all $i < k$,

$$col_k \circ \mu_{i+1,k} = F(\mu_{i,k}) : \mathcal{D}(i+1) \to F(C_k).$$

col_k exists because $\{F(\mu_{i,k}) \mid i < k\}$ is a cocone of $F \circ \mathcal{D}_k$ and $\gamma_k \setminus \{\mu_{0,k}\}$ is the colimit of $F \circ \mathcal{D}_k$.

Let

$$\mu = \{\mu_i : \mathcal{D}(i) \to C \mid i < \lambda\}$$

be the colimit of $\mathcal{D}$ and F be λ-cocontinuous. Then

$$F(\mu) = \{F(\mu_i) : F(\mathcal{D}(i)) \to F(C) \mid i < \lambda\}$$

is the colimit of $F \circ \mathcal{D}$. Since $\mu \setminus \{\mu_0\}$ is a cocone of $F \circ \mathcal{D}$, there is a unique $\mathcal{K}$-morphism $col^C : F(C) \to C$ – and thus an F-algebra – such that for all $i < \lambda$,

$$col^C \circ F(\mu_i) = \mu_{i+1} : \mathcal{D}(i+1) \to C.$$

col^C is initial in Alg_F.

Proof. Let $\alpha : F(A) \to A$ be an F-algebra. Since A initial in $\mathcal{K}$, $\mathcal{D}$ has the cocone

$$\nu = \{\nu_i : \mathcal{D}(i) \to A \mid i < \lambda\}$$

with $\nu_0 = ini^A$ and $\nu_{i+1} = \alpha \circ F(\nu_i)$ for all $i < \lambda$. Hence there is a unique $\mathcal{K}$-morphism $fold^A : C \to A$ with $fold^A \circ \mu_i = \nu_i$ for all $i < \lambda$. We obtain

$$\begin{aligned} & fold^A \circ col^C \circ F(\mu_i) = fold^A \circ \mu_{i+1} = \nu_{i+1} = \alpha \circ F(\nu_i) = \alpha \circ F(fold^A \circ \mu_i) \\ & = \alpha \circ F(fold^A) \circ F(\mu_i). \end{aligned} \tag{1}$$

Since $\nu \setminus \{\nu_0\}$ is a cocone of $F \circ \mathcal{D}$ and $\mu \setminus \{\mu_0\}$ is the colimit of $F \circ \mathcal{D}$, there is only one $\mathcal{K}$-morphism $h : F(C) \to A$ with $h \circ F(\mu_i) = \nu_{i+1}$ for all $i < \lambda$. Hence (1) implies

$$fold^A \circ col^C = \alpha \circ F(fold^A),$$

i.e., col_A is an Alg_F-morphism from col^C to α.

Let $\theta : C \to A$ be an Alg_F-morphism from col^C to α. Suppose that for all $i < \lambda$,

$$\theta \circ \mu_i = \nu_i : \mathcal{D}(i) \to A. \quad (2)$$

Since $fold^A \circ \mu_i = \nu_i$ and there is only one $\mathcal{K}$-morphism $h : C \to A$ with $h \circ \mu_i = \nu_i$, we conclude $\theta = fold^A$. It remains to show (2) by transfinite induction on i.

Since $\mathcal{D}(0) = I$ is initial in $\mathcal{K}$, $\theta \circ \mu_0 = \nu_0$. Let $0 < k < \lambda$. If k is a successor ordinal, then $k = i + 1$ for some ordinal i and thus

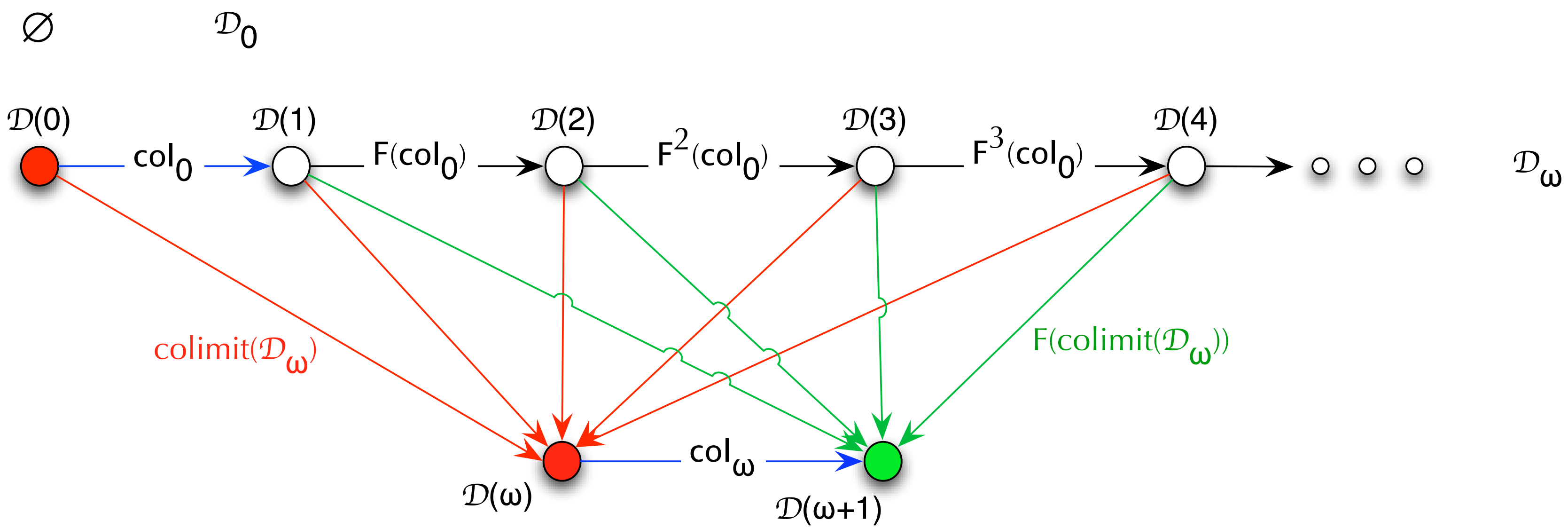
$$\theta \circ \mu_k = \theta \circ \mu_{i+1} = \theta \circ col^C \circ F(\mu_i) \stackrel{\theta \in Alg_F(C,A)}{=} \alpha \circ F(\theta) \circ F(\mu_i) = \alpha \circ F(\theta \circ \mu_i)$$
$$\stackrel{ind.\ hyp.}{=} \alpha \circ F(\nu_i) = \nu_{i+1} = \nu_k.$$

Let k be a limit ordinal. Since μ and ν are cocones of $\mathcal{D}$, $\mu_k \circ \mu_{i,k} = \mu_i$ and $\nu_k \circ \mu_{i,k} = \nu_i$ for all $i < k$.

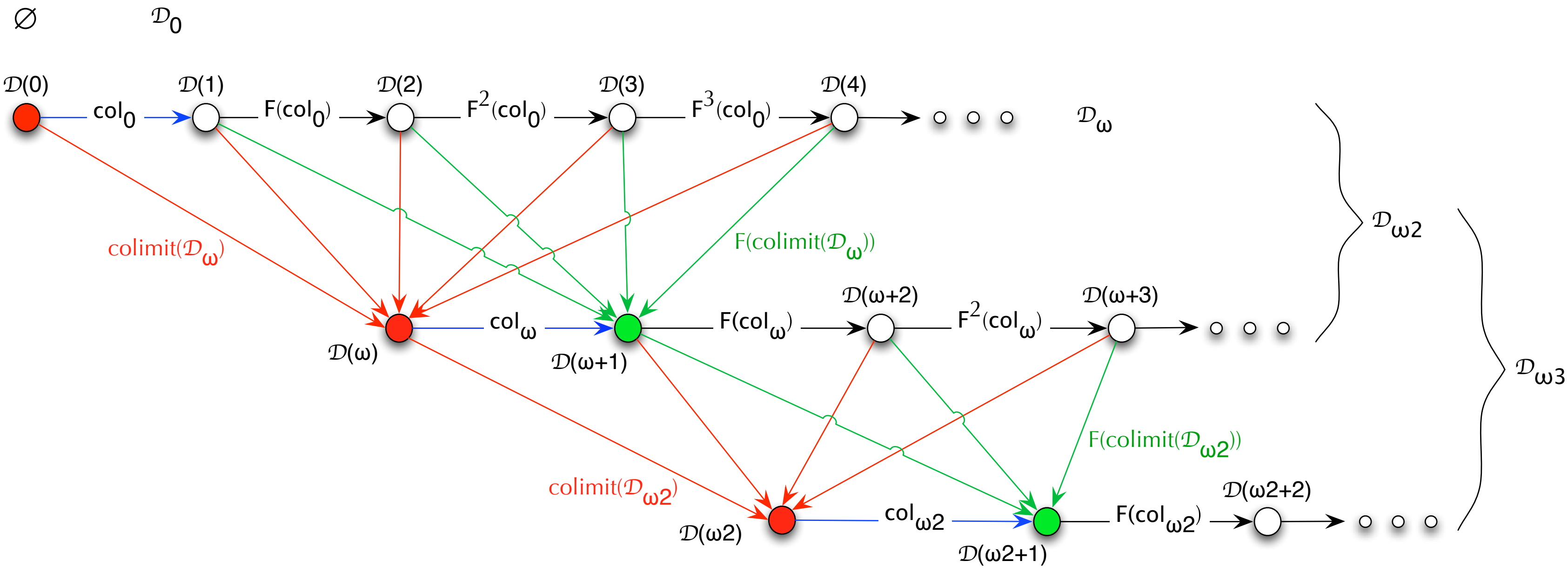
Hence by induction hypothesis,

$$\theta \circ \mu_k \circ \mu_{i,k} = \theta \circ \mu_i = \nu_i = \nu_k \circ \mu_{i,k}. \tag{3}$$

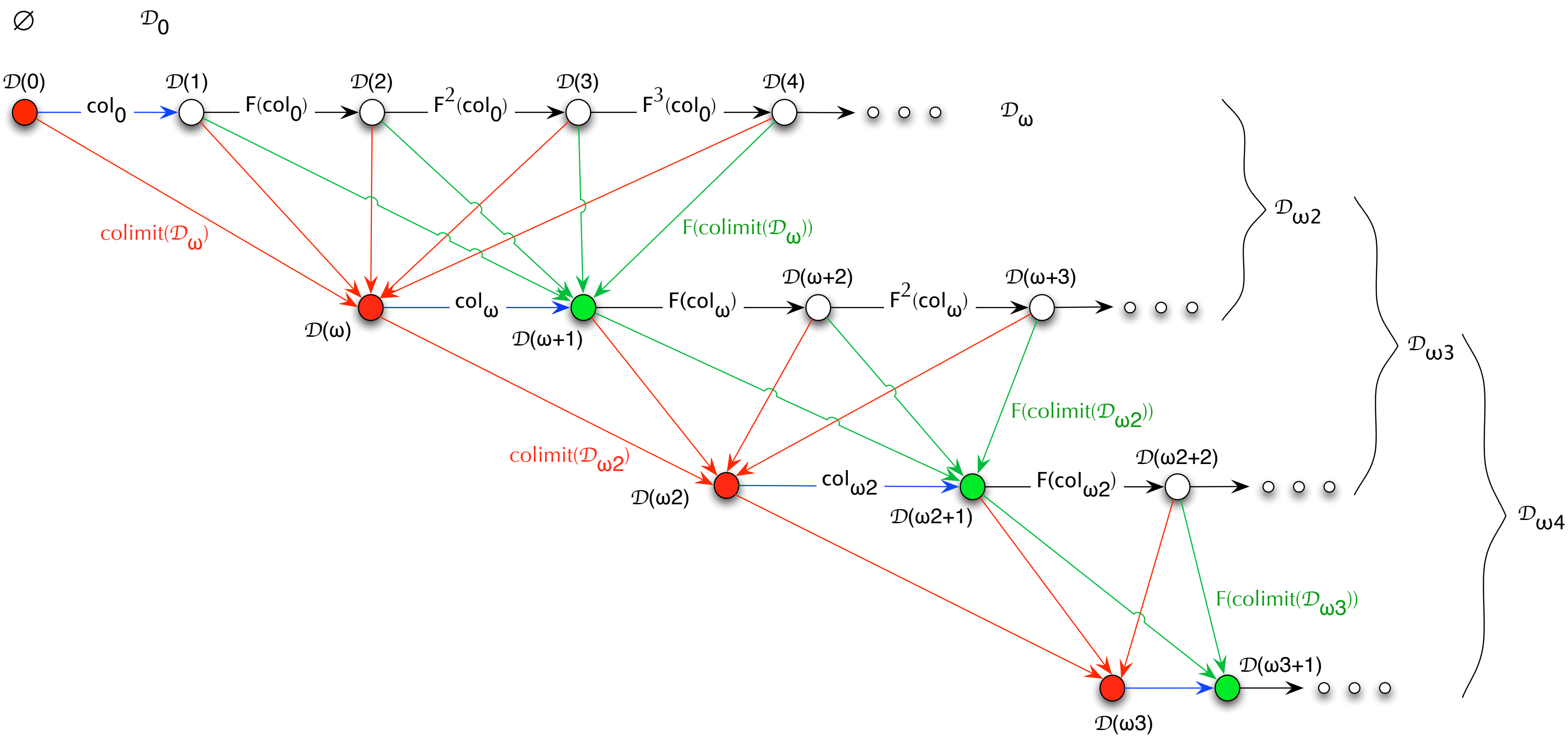
Since $\{\nu_i \mid i < k\}$ is a cocone of $\mathcal{D}_k$ and $\{\mu_{i,k} \mid i < k\}$ is the colimit of $\mathcal{D}_k$, there is only one $\mathcal{K}$-morphism $h : \mathcal{D}(k) \to A$ with $h \circ \mu_{i,k} = \nu_i$ for all $i < k$. Hence (3) implies $\theta \circ \mu_k = \nu_k$, and the proof of (2) is complete. ❑



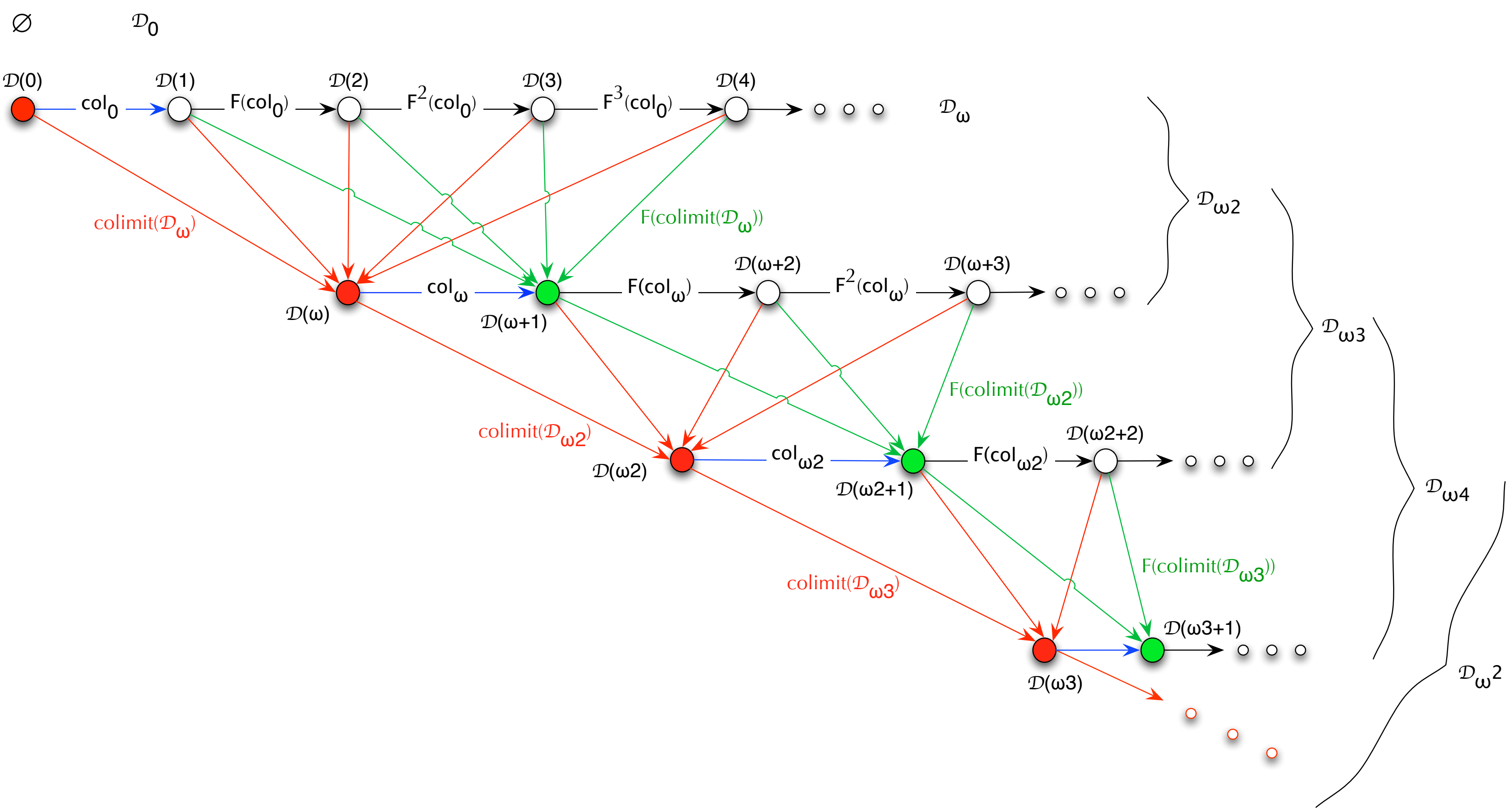
The $\omega + 2$-chain of $\mathcal{K}$ induced by the initial object $\mathcal{D}(0)$ of $\mathcal{K}$



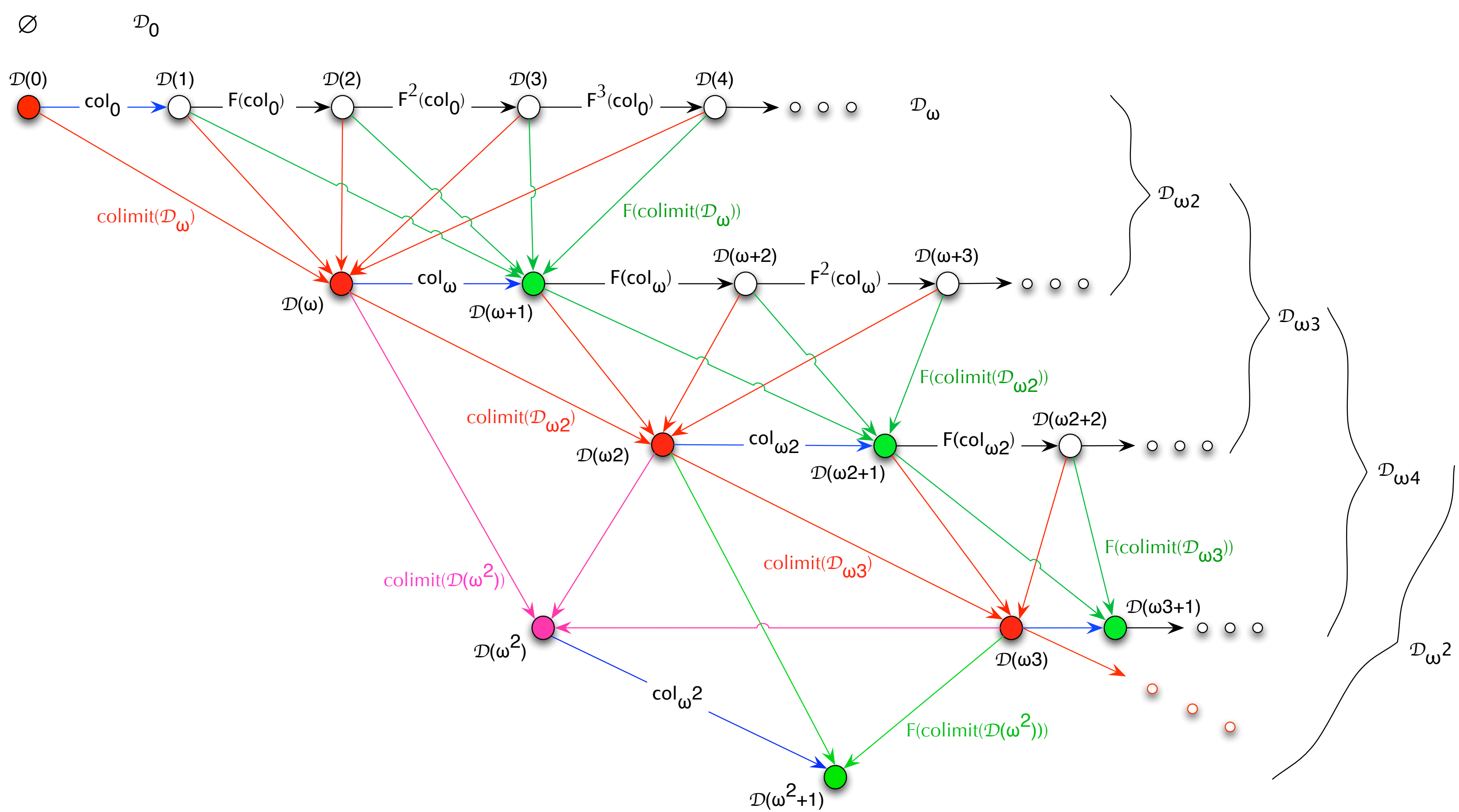
The $(\omega 2 + 2)$-chain of $\mathcal{K}$ induced by the initial object $\mathcal{D}(0)$ of $\mathcal{K}$



The $(\omega 3 + 2)$*-chain of* $\mathcal{K}$ *induced by the initial object* $\mathcal{D}(0)$ *of* $\mathcal{K}$



The ω^2-chain of $\mathcal{K}$ induced by the initial object $\mathcal{D}(0)$ of $\mathcal{K}$



The $(\omega^2 + 2)$-chain of $\mathcal{K}$ induced by the initial object $\mathcal{D}(0)$ of $\mathcal{K}$

Theorem GFIX

Let λ be an infinite cardinal, Fin be final in $\mathcal{K}$ and $\mathcal{K}$ be κ-complete for all $\kappa \leq \lambda$.

Given a functor $F : \mathcal{K} \to \mathcal{K}$, define a λ-cochain $\mathcal{D}$ of $\mathcal{K}$ as follows:

$$\begin{array}{rcll} \mathcal{D}(0) & = & Fin, & \\ \mathcal{D}(k+1) & = & F(\mathcal{D}(k)) & \text{for all } k < \lambda, \\ \mathcal{D}(k) & = & L_k & \text{for all limit ordinals } k < \lambda, \\ \mathcal{D}(k,i) & = & \mu_{k,i} & \text{for all limit ordinals } k < \lambda \text{ and all } i < k, \\ \mathcal{D}(k+1,k) & = & lim_k & \text{for } k = 0 \text{ and all limit ordinals } k < \lambda, \\ \mathcal{D}(i+1,j+1) & = & F(\mathcal{D}(i,j)) & \text{for all } i \geq j < \lambda \end{array}$$

where $\gamma_k = \{\mu_{k,i} : L_k \to \mathcal{D}(i) \mid i < k\}$ is the limit of the greatest subdiagram $\mathcal{D}_k : \mathbb{O}_k \to \mathcal{K}$ of $\mathcal{D}$ and lim_k is the unique $\mathcal{K}$-morphism from $F(L_k)$ to L_k such that for all $i < k$,

$$\mu_{k,i+1} \circ lim_k = F(\mu_{k,i}) : F(L_k) \to \mathcal{D}(i+1).$$

lim_k exists because $\{F(\mu_{k,i}) \mid i < k\}$ is a cone of $F \circ \mathcal{D}_k$ and $\gamma_k \setminus \{\mu_{k,0}\}$ is the colimit of $F \circ \mathcal{D}_k$.

Let

$$\mu = \{\mu_i : C \to \mathcal{D}(i) \mid i < \lambda\}$$

be the limit of $\mathcal{D}$ and F be F be λ-continuous. Then

$$F(\mu) = \{F(\mu_i) : F(C) \to F(\mathcal{D}(i)) \mid i < \lambda\}$$

is the limit of $F \circ \mathcal{D}$. Since $\mu \setminus \{\mu_0\}$ is a cone of $F \circ \mathcal{D}$, there is a unique $\mathcal{K}$-morphism $lim^C : C \to F(C)$ – and thus an F-coalgebra – such that for all $i < \lambda$,

$$F(\mu_i) \circ lim^C = \mu_{i+1} : C \to \mathcal{D}(i+1).$$

lim^C is final in $coAlg_F$.

Proof. Let $\alpha : A \to F(A)$ be an F-coalgebra. Since A final in $\mathcal{K}$, $\mathcal{D}$ has the cone

$$\nu = \{\nu_i : A \to \mathcal{D}(i) \mid i < \lambda\}$$

with $\nu_0 = fin^A$ and $\nu_{i+1} = F(\nu_i) \circ \alpha$ for all $i < \lambda$. Hence there is a unique $\mathcal{K}$-morphism $unfold^A : A \to L$ with $\mu_i \circ unfold^A = \nu_i$ for all $i < \lambda$. Proceed analogously to the proof of Theorem LFIX. ❏

Corollary

Suppose that all co/chains of $\mathcal{K}$ have co/limits. Then the definition of the λ-co/chain $\mathcal{D}$ in Theorem LFIX resp. GFIX can be extended to the definition of a co/chain.

If $F : \mathcal{K} \to \mathcal{K}$ is λ-co/continuous, then $\mathcal{D}$ **converges in** λ **steps**, i.e., $\mathcal{D}(\lambda) \cong \mathcal{D}(\lambda + 1)$.

Proof. The conjecture follows immediately from Lambek's Lemma and Theorem LFIX resp. GFIX. ❏

Constructive-signature functors

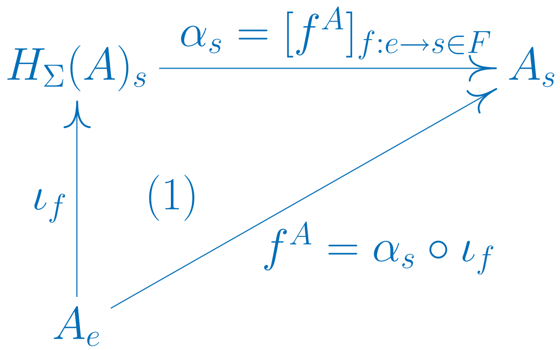
Let $\Sigma = (S, BS, F, P)$ be a constructive signature.

Σ induces the functor $H_\Sigma : Set^S \to Set^S$: For all $A, B \in Set^S$, $h \in Set^S(A, B)$ and $s \in S$,

$$\begin{aligned} H_\Sigma(A)_s &= \coprod_{f:e\to s\in F} A_e, \\ H_\Sigma(h)_s &= \coprod_{f:e\to s\in F} h_e. \end{aligned}$$

A H_Σ-algebra (see F-algebras and F-coalgebras) $\alpha : H_\Sigma(A) \to A$ uniquely corresponds to a Σ-algebra A and vice versa:

For all $s \in S$ and $f : e \to s \in F$,



Hence α_s is the coproduct extension of the interpretations of all constructors of Σ in A.

Moreover, given Σ-algebras A and B and corresponding H_Σ-algebras α resp. β, an S-sorted function $h : A \to B$ is Σ-homomorphic iff h is an Alg_{H_Σ}-morphism from α to β.

Examples

Let A be an S-sorted set.

$$
\begin{aligned}
H_{Nat}(A)_{nat} &= 1 + A_{nat}, \\
H_{Reg(X)}(A)_{reg} &= 1 + 1 + X + A_{reg}^2 + A_{reg}^2 + A_{reg}, \\
H_{List(X)}(A)_{list} &= 1 + (X \times A_{list}), \\
H_{Bintree(X)}(A)_{tree} &= 1 + A_{btree} \times X \times A_{btree}, \\
H_{Tree(X,Y)}(A)_{tree} &= X \times A_{trees}, \\
H_{Tree(X,Y)}(A)_{trees} &= 1 + (Y \times A_{tree} \times A_{trees}), \\
H_{BagTree(X,Y)}(A)_{trees} &= X \times \mathcal{B}_{fin}(Y \times A_{tree}), \\
H_{FDTree(X,Y)}(A)_{trees} &= X \times ((Y \times A_{tree})^{\mathbb{N}} + (Y \times A_{tree})^*). \qquad ❏
\end{aligned}
$$

Let κ be the cardinality of the greatest (base set) exponent occurring in the domain of some $f \in F$ and λ be the first regular cardinal number $> \kappa$.

By Theorem CONTYPES, H_Σ is λ-cocontinuous and thus by Theorem LFIX, Alg_{H_Σ} has an initial object $\alpha : H_\Sigma(\mu\Sigma) \to \mu\Sigma$. In other words, $\mu\Sigma$ is the initial Σ-algebra (see (1)).

Since $\mu\Sigma$ is the colimit of the λ-chain $\mathcal{D}$ of Set^S defined in Theorem LFIX, the Quotient Theorem implies that for all $s \in S$,

$$\mu\Sigma_s \;=\; (\coprod_{i<\lambda} \mathcal{D}(i)_s)/{\sim_s}$$

where $\sim_s$ is the equivalence closure of

$$\{(a, \mathcal{D}(i, i+1)(a)) \mid a \in \mathcal{D}(i)_s,\; i < \lambda\}.$$

Let A be a Σ-algebra. The unique Σ-homomorphism $fold^A : \mu\Sigma \to A$ is the unique S-sorted function such that

$$\coprod_{i<\lambda} \mathcal{D}(i) \stackrel{[\beta_i]_{i<\lambda}}{\longrightarrow} A \quad = \quad \coprod_{i<\lambda} \mathcal{D}(i) \stackrel{nat_\sim}{\longrightarrow} \mu\Sigma \stackrel{fold^A}{\longrightarrow} A$$

where β_0 is the unique S-sorted function from $\mathcal{D}(0)$ to A and for all $i < \lambda$ and $s \in S$,

$$\beta_{i+1,s} = [f^A \circ F_e(\beta_{i,s})]_{f:e\to s\in F} : \mathcal{D}(i+1)_s \to A_s.$$

Flat constructive signatures

Σ is **flat** if the domain of each function symbol of Σ is a finite product of flat types.

If Σ is not flat, Σ can often be transformed into an equivalent flat signature $\Sigma' = (S, BS, F', P)$, i.e., $Alg_\Sigma \cong Alg_{\Sigma'}$. For instance,

- a constructor $f : e \times (e_1 + \cdots + e_n) \longrightarrow s$ is flattened by adding $e_1 + \cdots + e_n$ as a new sort to S and the injections $\iota_i : e_i \longrightarrow e_1 + \cdots + e_n$, $1 \leq i \leq n$, as new constructors to F;
- a constructor $f : e \times e'^B \longrightarrow s$ with finite $B \in BS$ is flattened by adding e'^B as a new sort to S and B-tupling $(\dots, \dots, \dots) : \prod_{b \in B} e' \longrightarrow e'^B$ as a new constructor to F.

The initial model of a flat constructive signature

Let $\Sigma = (S, BS, F, P)$ be flat.

H_Σ is ω-cocontinuous and its object mapping reads as follows: For all S-sorted sets A and $s \in S$,

$$\begin{aligned} H_\Sigma(A)_s &= \coprod_{f:e_1\times\cdots\times e_n\to s\in F} \prod_{i=1}^n A_{e_i} \\ &= \{((a_1,\ldots,a_n),f) \mid f : e_1 \times \cdots \times e_n \to s \in F,\ a_i \in A_{e_i},\ 1 \le i \le n\}. \end{aligned}$$

Moreover, for all $s \in S$, $k \in \mathbb{N}$ and $t \in \mathcal{D}(k)$,

$$\begin{aligned} \mathcal{D}(0)_s &= \emptyset, \\ \mathcal{D}(k+1)_s &= H_\Sigma(\mathcal{D}(k))_s \\ &= \{((t_1,\ldots,t_n),f) \mid f : e_1 \times \cdots \times e_n \to s \in F,\ t_i \in \mathcal{D}(k)_{e_i},\ 1 \le i \le n\}, \\ \mathcal{D}(k,k+1)(t) &= t \end{aligned}$$

and thus by the Quotient Theorem,

$$\mu\Sigma_s \;=\; (\coprod_{k\in\mathbb{N}} \mathcal{D}(k)_s)/{\sim_s} \;\cong\; \bigcup_{k\in\mathbb{N}} \mathcal{D}(k)_s$$

where $\sim_s$ is the equivalence closure of $\{(t, \mathcal{D}(k, k+1)(t)) \mid t \in \mathcal{D}(k)_s,\; k \in \mathbb{N}\} = \Delta_{\mathcal{D}(k),s}$.

By Lambek's Lemma, the H_Σ-algebra α (see (1)) is an isomorphism and thus for all $f : e_1 \times \cdots \times e_n \to s \in F$ and $t_i \in \mu\Sigma_{e_i}$, $1 \le i \le n$,

$$f^{\mu\Sigma}(t_1, \ldots, t_n) = \iota_f(t_1, \ldots, t_n) = ((t_1, \ldots, t_n), f).$$

Since $((t_1, \ldots, t_n), f)$ is represented by a tree with root label f and maximal proper subtrees, we write $f(t_1, \ldots, t_n)$ for $((t_1, \ldots, t_n), f)$.

Hence for all Σ-algebras A,

$$fold^A(f(t_1, \ldots, t_n)) = fold^A(f^{\mu\Sigma}(t_1, \ldots, t_n)) = f^A(fold^A_{e_1}(t_1), \ldots, fold^A_{e_n}(t_n)).$$

The carriers of $\mu\Sigma$ can be represented as equivalence classes of trees:

Let T be the least $\mathbb{FT}(S, BS)$-sorted set of finite trees t such that

- for all $X \in BS$, $t \in T_X$ if t is a leaf labelled with some element of X,
- for all $s \in S$, $t \in T_s$ if the root of t is labelled with some $f : e_1 \times \cdots \times e_n \to s \in F$ and the tuple of maximal proper subtrees of t is in $T_{e_1} \times \ldots \times T_{e_n}$,
- for all collection types $c(s) \in \mathbb{FT}(S, BS)$, $t \in T_{c(s)}$ if the root of t is labelled with c and the tuple of maximal proper subtrees of t is in T_s^*.

Hence for all $t \in T$,

- a node n is a leaf of t iff n is labelled with an element of some $X \in BS$,
- n is an inner node iff n is labelled with a constructor of Σ, *word*, *bag* or *set*.

Let $\sim$ be the least equivalence relation on T such that for all $e \in \mathbb{FT}(S, BS)$, $t, u \in T_e$ and the lists $t_1, \ldots, t_m$ and $u_1, \ldots, u_n$ of maximal proper subtrees of t resp. u, $t \sim u$ if

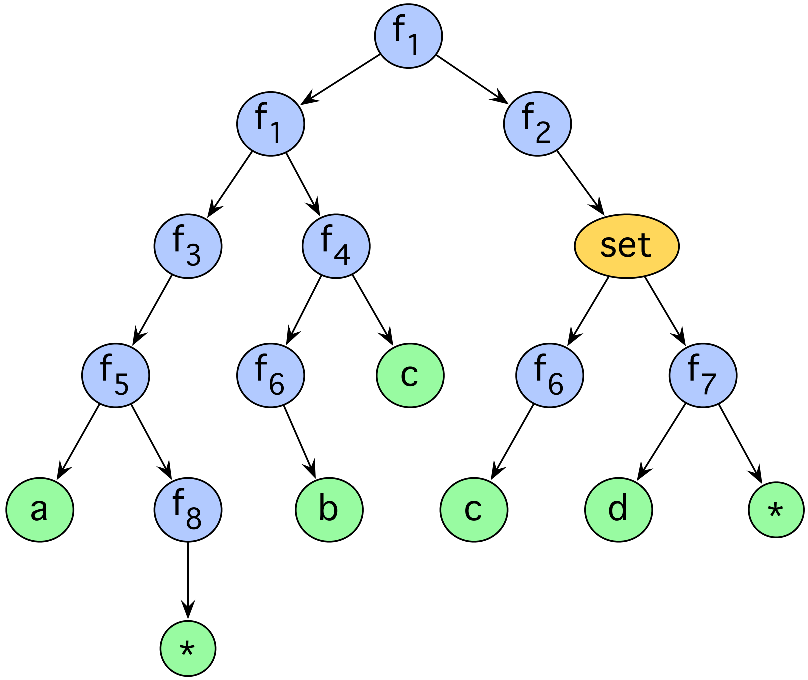
- $e \in S \cup BS$, $m = n$ and for all $1 \leq i \leq n$, $t_i \sim u_i$, or
- e is a word type, $m = n$ and for all $1 \leq i \leq n$, $t_i \sim u_i$, or
- e is a bag type, $m = n$ and there is a bijection h on $\{1, \ldots, n\}$ such that for all $1 \leq i \leq n$, $t_i \sim u_{h(i)}$, or
- e is a set type, for all $1 \leq i \leq m$ there is $1 \leq j \leq n$ with $t_i \sim u_j$ and for all $1 \leq j \leq n$ there is $1 \leq i \leq n$ with $t_i \sim u_j$.

For all $e \in \mathbb{FT}(S, BS)$, $\mu\Sigma_e \cong T_e/\!\sim$.

If F does not contain bag or set types, then $\mu\Sigma_e \cong T_e/\!\sim = T_e$.

The elements of $\mu\Sigma$ are called **finite ground Σ-terms**.

For all $k \in \mathbb{N}$, $\mathcal{D}(k)$ is represented by the (equivalence classes of) finite ground Σ-terms t with $depth(t) \leq k$.



A finite ground Σ*-term with constructors* $f_1, \ldots, f_8$ *and base elements* $a, b, c, d, *$.

Examples

$\mathbb{N}$ is an initial *Nat*-algebra: $0^{\mathbb{N}} = 0$ and for all $n \in \mathbb{N}$, $succ^{\mathbb{N}}(n) = n + 1$.

$T_{Reg(X)}$ is an initial $Reg(X)$-algebra. Hence $T_{Reg(X),reg}$ is the set of regular expressions over X. For all such expressions R, $fold^{Lang}(R)$ is the language of R and $fold^{Bool}(R)$ checks it for inclusion of the empty word.

For $\Sigma \in \{List(X), Tree(X,Y), BagTree(X,Y), FDTree(X,Y)\}$, the elements of the *list*- resp. *tree*-carrier of an initial Σ-algebra can be represented by the sequences resp. trees that we associated in Signatures with Σ. ❏

Predicate induction

Let $\Sigma = (S, BS, F, P)$ be a signature and C be a Σ-algebra.

Predicate induction is an (analytical, top-down) inference rule that allows us to prove properties of the interpretation of P' in $lfp(\Phi_{\Sigma',C,AX})$. The properties are given by Σ-formulas $\psi_p : e$, one for each $p : e \in P'$. The goals $p \Rightarrow \psi_p$, $p \in P'$, are replaced by the axioms for P', which are then coresolved upon the goals:

$$(1) \qquad \frac{p \Rightarrow \psi_p}{\bigwedge_{pt \Leftarrow \varphi \in AX}(\varphi[\psi_p/p \mid p \in P'] \Rightarrow \psi_p t)} \Uparrow$$

If further top-down rules (e.g. resolution and narrowing) transform the succedent of (1) to *True*, then by Lemma IND, C satisfies the antecedent of (1).

Goals can often be proved by induction only after they have been *generalized*: Some formula δ_p must be found such that C satisfies $p \Rightarrow \psi_p \wedge \delta_p$. The generalization strengthens the induction hypothesis in the succedent of (1) from $\varphi[\psi_p/p]$ to $\varphi[\psi_p \wedge \delta_p/p]$.

In order to find δ_p, q_p and $q_p \Rightarrow \psi_p$ are added to Σ resp. AX when (1) is applied. The succedent of (1) is modified accordingly:

$$(2) \qquad \frac{p \Rightarrow \psi_p}{\bigwedge_{pt \Leftarrow \varphi \in AX}(\varphi[q_p/p \mid p \in P'] \Rightarrow \psi_p t)}$$

The demand for generalizing the goal $p \Rightarrow \psi_p$ becomes apparent in the course of proving the succedent of (2) when a subgoal of the form $q_p \Rightarrow \delta_p$ is encountered:

If $\delta_p = \psi_p$, then the subgoal $q_p \Rightarrow \delta_p$ agrees with the added axiom and thus reduces to *True*. Otherwise $q_p \Rightarrow \delta_p$ is added to AX and the proof proceeds with an application of the following rule:

$$(3) \qquad \frac{q_p \Rightarrow \delta_p}{\bigwedge_{pt \Leftarrow \varphi \in AX}(\varphi[q_p/p \mid p \in P'] \Rightarrow \delta_p t)}$$

Between the applications of (2) resp. (3), coresolution steps upon the added axiom $q_p \Rightarrow \psi_p$ must be confined to redex positions with negative polarity, i.e., the number of preceding negation symbols in the entire formula must be odd. Otherwise the axiom added when (3) is applied might violate the soundness of the coresolution steps.

Coresolution upon q_p at any redex position becomes sound as soon as the set of axioms for q_p is not extended any more.

By inferring *True* from the conclusions of (2) and (3) one shows, roughly speaking, that the predicate $\psi_p \wedge \delta_p$ solves the axioms for p. Since p itself represents the least solution, we conclude $p \Rightarrow \psi_p \wedge \delta_p$, in particular the original goal $p \Rightarrow \psi_p$.

Predicate induction allows us to prove properties of *least* predicates. If, however, P' consists of greatest predicates, then proving goals of the form $p \Rightarrow \psi_p$ amounts to coresolving them upon p.

Induction for proving membership

Let $P' = \{inv_s : s \mid s \in S\}$, $\Sigma' = (S, F, P + P')$,

$$AX \;=\; \{inv_{e'}(fx) \Leftarrow inv_e(x) \mid f : e \to e' \in F\},$$

C be initial in a full subcategory of Alg_Σ, R be an S-sorted subset of C and ψ be an S-sorted set of Σ-formulas such that for all $s \in S$, $\psi_s^C = R_s$. By Lemma MAX (1),

$$\begin{array}{rcl} C \subseteq R & \Longleftrightarrow & R \text{ contains some } \Sigma\text{-invariant of } C \\ & \Longleftarrow & R \text{ contains the least } \Sigma\text{-invariant of } C \\ & \Longleftarrow & \text{the succedent of predicate induction is valid} \\ & & \text{for } P',\ AX \text{ and } \psi \text{ defined as above.} \end{array}$$

******

Suppose that for all $s\in S$, s-membership $\in_s$: s belongs to P, and AX is a set of Horn clauses such that for all Σ-algebras A satisfying AX, $\in^A=\{\in_s^A \mid s\in S\}$ is a Σ-invariant.

Let $\mu\Sigma$ be initial in $Alg^{\equiv}_{\Sigma,AX}$ or $obs(Alg^{\in}_{\Sigma,AX})$ (see Thm. ABSINI resp. RESINI). Then $\in^{\mu\Sigma}$ is the least Σ-invariant of $\mu\Sigma$ that satisfies AX.

Let R be an S-sorted subset of $\mu\Sigma$ and for all $s\in S$, $\psi_s : s$ be a Σ-formula that describes R_s, i.e., R_s coincides with $\psi_s^{\mu\Sigma}$. By algebraic induction, $\mu\Sigma\subseteq R$ if for all $s\in S$, $\in_s^{\mu\Sigma}\subseteq\psi_s^{\mu\Sigma}$.

A **context-free grammar** $G = (N, BS, X, R)$ consists of finite sets S of **nonterminals**, BS of **base sets**, X of **terminals**, and $R \subseteq N \times (N \cup BS \cup X)^*$ of **rules**.

The constructive signature $\Sigma(G) = (N, BS, F, \emptyset)$ with

$$F = \{f_r : e_1 \times \cdots \times e_n \to s \mid \begin{array}{l} r = (s, w_0e_1w_1 \ldots e_nw_n) \in R, \\ e_1, \ldots, e_n \in N \cup BS, \ w_0, \ldots, w_n \in X^* \end{array}\}$$

is called the **abstract syntax of** G (see [23], Section 3.1).

$\Sigma(G)$-terms are called **syntax trees** of G.

******

The **word algebra of** G, $Word(G)$, is the $\Sigma(G)$-algebra defined as follows:

- For all $s \in S$, $Word(G)_s = X^*$.
- For all $w_0, \ldots, w_n \in Z^*$, $e_1, \ldots, e_n \in S \cup BS$, $r = (s, w_0s_1w_1 \ldots s_nw_n) \in R$ and $v \in Word(G)_{e_1 \times \cdots \times e_n}$, $f_r^{Word(G)}(v) = w_0v_1w_1 \ldots v_nw_n$.

$L(G) = fold^{Word(G)}(T_{\Sigma(G)})$ is called the **language of** G.

$L(G)$ is also the least solution in S of the set $E(G)$ of equations between the left- and right-hand sides of R. If G is not left-recursive ($\forall\ A \in N, w \in B^* : A \not\xrightarrow{+}_G Aw$), then the solution is unique [49]. This provides a simple method for proving that a given language L agrees with $L(G)$:

$$L = L(G) \iff L \text{ solves } E(G) \text{ in } S.$$

Let $B = Z \cup (\cup BS)$. Every **parser for** G can be presented as a function

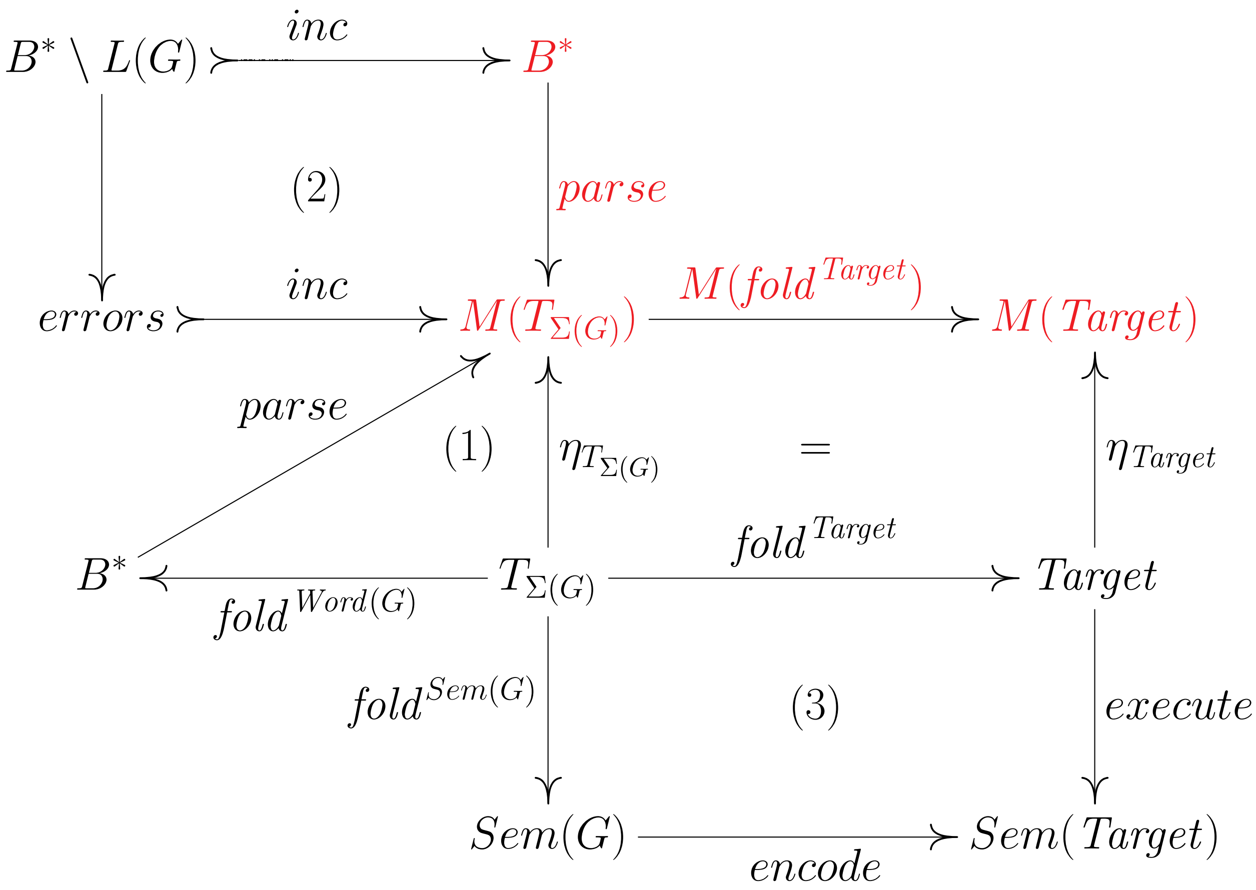
$$parse : B^* \to M(T_{\Sigma(G)})$$

where (M, η, ϵ) is a monad that embeds $T_{\Sigma(G)}$ into a larger set of possible results like syntax errors or sets of syntax trees instead of a single one [49].

$parse$ **is correct** if

- $parse \circ fold^{Word(G)} = \eta_{T_{\Sigma(G)}}$, (1)
- for all $w \in B^* \setminus L(G)$, $parse(w)$ is an error message. (2)

If the target language of a **compiler** *comp* **for** G is presented as a $\Sigma(G)$-algebra *Target*, $comp : B^* \to M(Target)$ is the composition of *parse* and $M(fold^A)$:



The inductive construction of syntax trees by $parse$ can be transformed into an inductive construction of target objects. Consequently, the compiler compiles its input directly without building a syntax tree!

As $fold^{Target}$ is just one instance of a generic function that takes an *arbitrary* $\Sigma(G)$-algebra $Target$ and evaluates the syntax trees of $M(T_{\Sigma(G)})$ in $Target$, so

$$comp \;=\; M(fold^{Target}) \circ parse$$

is just one instance of a generic function that takes $Target$ and compiles input from B^* to elements of $Target$.

Moreover, expressing target languages as $\Sigma(G)$-algebras provides a method for proving the commutativity of (3), i.e. the correctness of $comp$ w.r.t. given semantics $Sem(G)$ and $Sem(Target)$ of G resp. $Target$:

- Suppose that $Sem(Target)$ is a $\Sigma(G)$-algebra and $execute$ and $encode$ are $\Sigma(G)$-homomorphic. Then all functions of (3) are $\Sigma(G)$-homomorphic and thus (3) commutes because $T_{\Sigma(G)}$ is initial in $Alg_{\Sigma(G)}$. (4)

Usually, there is a target signature Σ' such that $T_{\Sigma'} = Target$, each constructor of $\Sigma(G)$ corresponds some Σ'-term, $Sem(Target)$ is a Σ'-algebra and *execute* is Σ'-term evaluation in $Sem(Target)$. Then the correspondence between $\Sigma(G)$-constructors and Σ'-terms may be transferable to their interpretations in $\Sigma(G)$ resp. $Sem(Target)$ such that, indeed, *execute* and *encode* become $\Sigma(G)$-homomorphic. In [64], the method is applied to the translation of imperative programs into data-flow graphs.

To sum up, using algebra in compiler design allows us to

- omit the explicit construction of syntax trees,
- to parameterize the same compiler with different monads that implement different parsing techniques (deterministic, nondeterministic, fine-grain error handling, etc.),
- to parameterize the same compiler with different target languages,
- to employ the fact that abstract syntax trees form an initial algebra when proving the correctness of the compiler.

Example

The grammar SAB $= (N, Z, \emptyset, R)$ consists of $N = \{S, A, B\}$, $Z = \{a, b\}$ and the rules

$$r_1 = S \to aB, \quad r_2 = S \to bA, \quad r_3 = S \to \epsilon,$$
$$r_4 = A \to aS, \quad r_5 = A \to bAA, \quad r_6 = B \to bS, \quad r_7 = B \to aBB.$$

For all $w \in Z^*$ and $x \in Z$ let $w\#x$ be the number of occurrences of x in w. It is easy to see that $g : N \to Lang$ with

$$\begin{aligned} g(S) &= \{w \in Z^* \mid w\#a = w\#b\} \\ g(A) &= \{w \in Z^* \mid w\#a = w\#b + 1\} \\ g(B) &= \{w \in Z^* \mid w\#a = w\#b - 1\} \end{aligned}$$

solves the equations derived from R in $Lang$. Since SAB is not left-recursive, there is only one solution. Hence

$$L(G)_S = g(S), \quad L(G)_A = g(A), \quad L(G)_B = g(B).$$

The abstract syntax of SAB consists of the sorts S, A, B and the function symbols

$$f_{r_1} : B \to S, \quad f_{r_2} : A \to S, \quad f_{r_3} : \epsilon \to S,$$
$$f_{r_4} : S \to A, \quad f_{r_5} : AA \to A,$$
$$f_{r_6} : S \to B, \quad f_{r_7} : BB \to B.$$

The three carriers of the word algebra $\mathcal{A} = \mathit{Word}(\mathrm{SAB})$ are given by $\{a, b\}^*$. The function symbols of $\Sigma(\mathrm{SAB})$ are interpreted in $\mathcal{A}$ as follows: For all $v, w \in \{a, b\}^*$,

$$\begin{array}{lcl} f_{r_1}^{\mathcal{A}}(w) & = & f_{r_4}^{\mathcal{A}}(w) \;=\; aw, \\ f_{r_2}^{\mathcal{A}}(w) & = & f_{r_6}^{\mathcal{A}}(w) \;=\; bw, \\ f_{r_3}^{\mathcal{A}} & = & \epsilon, \\ f_{r_5}^{\mathcal{A}}(v, w) & = & bvw, \\ f_{r_7}^{\mathcal{A}}(v, w) & = & avw. \end{array}$$

Compiler from Z^* into an arbitrary Σ(SAB)-algebra *Target*, written in Haskell:

```
compile_S w = msum ($ w) [try_r1,try_r2,try_r3]
              where try_r1 w = do (x,w) <- compile_a w
                                  (c,w) <- compile_B w
                                  return (f_r1^Target c,w)
                    try_r2 w = do (x,w) <- compile_b w
                                  (c,w) <- compile_A(w)
                                  return (f_r2^Target c,w)
                    try_r3 w = return (f_r3^Target,w)
```

```
compile_A w = msum ($ w) [try_r4,try_r5]
              where try_r4 w = do (x,w) <- compile_a w
                                  (c,w) <- compile_S w
                                  return (f_r4^Target(c),w)
                    try_r5 w = do (x,w) <- compile_b w
                                  (c,w) <- compile_A w
                                  (d,w) <- compile_A w
                                  return (f_r5^Target(c,d),w)
compile_B w = msum ($ w) [try_r6,try_r7]
              where try_r6 w = do (x,w) <- compile_b w
                                  (c,w) <- compile_S w
                                  return (f_r6^Target(c),w)
                    try_r7 w = do (x,w) <- compile_a w
                                  (c,w) <- compile_B w
                                  (d,w) <- compile_B w
                                  return (f_r7^Target(c,d),w)
compile_a w = if null w || head w /= a then error else return (a,tail w)
compile_b w = if null w || head w /= b then error else return (b,tail w)
```

Destructive-signature functors

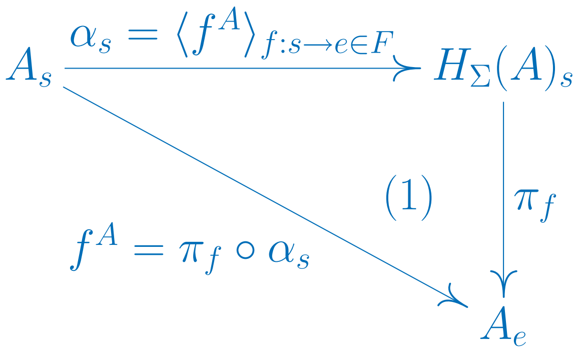
Let $\Sigma = (S, BS, F, P)$ be a destructive signature.

Σ induces the functor $H_\Sigma : Set^S \to Set^S$: For all $A, B \in Set^S$, $h \in Set^S(A, B)$ and $s \in S$,

$$H_\Sigma(A)_s = \prod_{f:s\to e\in F} A_e,$$
$$H_\Sigma(h)_s = \prod_{f:s\to e\in F} h_e.$$

A H_Σ-coalgebra $\alpha : A \to H_\Sigma(A)$ (see F-algebras and F-coalgebras) uniquely corresponds to a Σ-algebra A and vice versa:

For all $s \in S$ and $f : s \to e \in F$,



Hence α_s is the product extension of the interpretations of all destructors of Σ in A.

Moreover, given Σ-algebras A and B and corresponding H_Σ-coalgebras α resp. β, an S-sorted function $h : A \to B$ is Σ-homomorphic iff h is a $coAlg_{H_\Sigma}$-morphism from α to β.

Examples

Let A be an S-sorted set.

$$
\begin{array}{lcl}
H_{coNat}(A)_{nat} & = & 1 + A_{nat}, \\
H_{Stream(X)}(A)_{list} & = & X \times A_{list}, \\
H_{coList(X)}(A)_{list} & = & 1 + (X \times A_{list}), \\
H_{Infbintree(X)}(A)_{btree} & = & A_{btree} \times X \times A_{btree}, \\
H_{coBintree(X)}(A)_{btree} & = & 1 + (A_{btree} \times X \times A_{btree}), \\
H_{coTree(X,Y)}(A)_{tree} & = & X \times A_{trees}, \\
H_{coTree(X,Y)}(A)_{trees} & = & 1 + (Y \times A_{tree} \times A_{trees}), \\
H_{FBTree(X,Y)}(A)_{tree} & = & X \times (Y \times A_{tree})^*, \\
H_{DAut(X,Y)}(A)_{state} & = & A_{state}^X \times Y, \\
H_{NDAut(X,Y)}(A)_{state} & = & \mathcal{P}_{fin}(A_{state})^X \times Y. \qquad ❏
\end{array}
$$

Lemma WEAKFIN

Let $\Sigma = (S, BS, F, P)$ and $\Sigma' = (S, BS', F', P)$ be destructive signatures, $\tau : H_{\Sigma'} \to H_\Sigma$ be a surjective natural transformation, A be final in $Alg_{\Sigma'}$ and

$$\alpha = \langle g^A \rangle_{g:s\to e' \in F'} : A \to H_{\Sigma'}(A)$$

be the corresponding $H_{\Sigma'}$-coalgebra (see (1)).

$\tau_A \circ \alpha : A \to H_\Sigma(A)$ is a H_Σ-coalgebra and thus by (1), the corresponding Σ-algebra has the same carriers as A (why we also denote it by A) and interprets F as follows: For all $f : s \to e \in F$,

$$f^A = \pi_f \circ \tau_{A,s} \circ \alpha_s.$$

$\tau_A \circ \alpha$ is **weakly final** in $coAlg_{H_\Sigma}$, i.e., for all $\beta \in coAlg_{H_\Sigma}$ there is a (not necessarily unique) $coAlg_{H_\Sigma}$-morphism from $\tau_A \circ \alpha$ to β.

In other words, A is weakly final in Alg_Σ, i.e., for all Σ-algebras B there is a (not necessarily unique) Σ-homomorphism from A to B.

Moreover, $A/{\sim}$ is final in Alg_Σ where $\sim$ is the greatest Σ-congruence on B (which is the union of all Σ-congruences on B).

Proof. The lemma generalizes [24], Lemma 2.3 (iv), [26], 4.3.2/3, or [9], 2.4.6/16, from Set to Set^S.

Let $\beta : B \longrightarrow H_\Sigma(B)$ be a H_Σ-coalgebra (see (1)). Since $\tau_B : H_{\Sigma'}(B) \longrightarrow H_\Sigma(B)$ is surjective, there is an S-sorted function $h : H_\Sigma(B) \longrightarrow H_{\Sigma'}(B)$ with $\tau_B \circ h = id_{H_\Sigma(B)}$.

Hence $h \circ \beta : B \longrightarrow H_{\Sigma'}(B)$ is a $H_{\Sigma'}$-coalgebra and thus there is a unique Σ'-homomorphism $unfold^B : B \longrightarrow A$. If F is interpreted in A as above, $unfold^B$ is also Σ-homomorphic:

$$\begin{aligned} H_\Sigma(unfold^B) \circ \beta &= H_\Sigma(unfold^B) \circ \tau_B \circ h \circ \beta = \tau_A \circ H_{\Sigma'}(unfold^B) \circ h \circ \beta \\ &= \tau_A \circ \alpha \circ unfold^B. \end{aligned}$$

Hence $nat_\sim \circ unfold^B$ is a Σ-homomorphism from B to $A/\!\sim$. Let $g, h : B \longrightarrow A/\!\sim$ be Σ-homomorphisms. There is an S-sorted function $m : A/\!\sim \longrightarrow A$ with $nat_\sim \circ m = id_{A/\sim}$. Let $\approx$ be the least Σ-congruence on A that contains all pairs $(m(g(b)), m(h(b)))$ with $b \in B$. Since $\sim$ is the largest Σ-congruence on A, $\approx \subseteq \sim$. Hence for all $b \in B$, $m(g(b)) \approx m(h(b))$ implies $m(g(b)) \sim m(h(b))$ and thus

$$g(b) = nat_\sim(m(g(b))) = nat_\sim(m(h(b))) = h(b).$$

We conclude $g = h$. ❑

Let Σ be polynomial.

By Theorem CONTYPES, H_Σ is ω-continuous and thus by Theorem GFIX, $coAlg_{H_\Sigma}$ has a final object $\alpha : \nu\Sigma \to H_\Sigma(\nu\Sigma)$. In other words, $\nu\Sigma$ is the final Σ-algebra (see (1)).

Since $\nu\Sigma$ is the limit of the ω-cochain $\mathcal{D}$ of Set^S defined in Theorem GFIX, the Subset Theorem implies that for all $s \in S$,

$$\nu\Sigma_s \;=\; \{a \in \prod_{i<\omega} \mathcal{D}(i)_s \mid \forall\, i < \omega : a_i = \mathcal{D}(i+1, i)(a_{i+1})\}.$$

Let A be a Σ-algebra. The unique Σ-homomorphism $unfold^A : A \to \nu\Sigma$ is the unique S-sorted function such that

$$A \stackrel{\langle\beta_i\rangle_{i<\omega}}{\longrightarrow} \prod_{i<\omega} \mathcal{D}(i) \;=\; A \stackrel{unfold^A}{\longrightarrow} \nu\Sigma \stackrel{inc}{\longrightarrow} \prod_{i<\omega} \mathcal{D}(i)$$

where β_0 is the unique S-sorted function from A to $\mathcal{D}(0)$ and for all $i < \omega$ and $s \in S$,

$$\beta_{i+1,s} = \langle F_e(\beta_{i,s}) \circ f^A \rangle_{f:s\to e\in F} : A_s \to \mathcal{D}(i+1)_s.$$

Flat destructive signatures

Σ is **flat** if the range of each function symbol of Σ is a finite or coproduct of flat types.

If Σ is not flat, Σ can often be transformed into an equivalent flat signature $\Sigma' = (S', BS, F', P)$, i.e., $Alg_\Sigma \cong Alg_{\Sigma'}$. For instance,

- a destructor $f : s \rightarrow e + (e_1 \times \cdots \times e_n)$ is flattened by adding $e_1 \times \cdots \times e_n$ as a new sort to S and the projections $\pi_i : e_1 \times \cdots \times e_n \rightarrow e_i$, $1 \leq i \leq n$, as new destructors to F;
- a destructor $f : s \rightarrow e + e'^B$ with $B \in BS$ is flattened by adding e'^B as a new sort to S and the projections $\pi_b : e'^B \rightarrow e'$, $b \in B$, as new destructors to F.

The final model of a flat destructive signature

Let $\Sigma = (S, BS, F, P)$ be flat and $F' = \{f' : s \to e'_1 + \cdots + e'_n \mid f' : s \to e_1 + \cdots + e_n \in F\}$ where for all $s \in S$, $set(s)' = word(s)$, and for all other flat types e, $e' = e$.

$\Sigma' = (S, BS, F', P)$ is flat and polynomial.

$H_{\Sigma'}$ is ω-continuous and its object mapping reads as follows: For all S-sorted sets A and $s \in S$,

$$\begin{aligned} H_\Sigma(A)_s &= \prod_{f:s\to e_1+\cdots+e_n\in F} \coprod_{i=1}^{n} A_{e_i} \\ &= \{g : F \to A \times \mathbb{N} \mid \forall\, f : s \to e_1 + \cdots + e_n \in F : \pi_1(g(f)) \in A_{e'_{\pi_2(g(f))}}\}. \end{aligned}$$

Moreover, for all $s \in S$, $k \in \mathbb{N}$ and $t \in \mathcal{D}(k+1)$,

$$\begin{aligned}
\mathcal{D}(0)_s &= 1 = \{*\} \\
\mathcal{D}(k+1)_s &= H_\Sigma(\mathcal{D}(k))_s = \{t : F \to \mathcal{D}(k) \times \mathbb{N} \mid \forall\, f : s \to e_1 + \cdots + e_n \in F : \\
&\qquad \pi_1(t(f)) \in \mathcal{D}(k)_{e'_{\pi_2(t(f))}}\}, \\
\mathcal{D}(k+1,k)(t) &= \pi_1 \circ t
\end{aligned}$$

and thus by the Subset Theorem,

$$\begin{aligned}
\nu\Sigma'_s &= \{t \in \textstyle\prod_{k\in\mathbb{N}} \mathcal{D}(k)_s \mid \forall\, k \in \mathbb{N} : \mathcal{D}(k+1,k)(\pi_{k+1}(t)) = \pi_k(t)\} \\
&= \{t \in \textstyle\prod_{k\in\mathbb{N}} \mathcal{D}(k)_s \mid \forall\, k \in \mathbb{N} : \pi_1 \circ \pi_{k+1}(t) = \pi_k(t)\}.
\end{aligned}$$

A surjective natural transformation $\tau : H_{\Sigma'} \to H_\Sigma$ is defined as follows: For all S-sorted sets A, $f : s \to e_1 + \cdots + e_n \in F$ and $b = (b_f)_{f \in F} \in H_{\Sigma'}(A) = \prod_{f:s\to e_1+\cdots+e_n \in F} \coprod_{i=1}^n A_{e'_i}$,

$$\pi_f(\tau_A(b)) = \begin{cases} (\{a_1,\dots,a_k\}, i) & \text{if } b_f = (a_1 \dots a_k, i) \text{ and } e_i \text{ is a set type,} \\ b_f & \text{otherwise.} \end{cases}$$

Since $A = \nu\Sigma'$ is final in $Alg_{\Sigma'}$. Lemma WEAKFIN implies that A is weakly final in Alg_Σ if F is interpreted as follows: For all $f : s \to e_1 + \cdots + e_n$,

$$f^A = \pi_f \circ \tau_{A,s} \circ \langle f'^A \rangle_{f \in F},$$

i.e., for all $a \in A_s$,

$$f^A(a) = \begin{cases} (\{a_1,\dots,a_k\}, i) & \text{if } f'^A(a) = (a_1 \dots a_k, i) \text{ and } e_i \text{ is a set type,} \\ f'^A(a) & \text{otherwise.} \end{cases}$$

Moreover, $\nu\Sigma = A/\sim$ is final in Alg_Σ where $\sim$ is the greatest Σ-congruence on A, i.e., the union of all S-sorted binary relations $\sim$ on A such that for all $s \in S$ and $a, b \in A_s$, $a \sim_s b$ implies $f^A(a) \sim_{e_1+\cdots+e_n} f^A(b)$, i.e.,

$$a \sim_s b \;\Rightarrow\; \begin{cases} \{a_1,\dots,a_k\} \sim_{e_i} \{b_1,\dots,b_l\} & \text{if } f'^A(a) = (a_1\dots a_k, i), \\ & f'^A(b) = (b_1\dots b_l, i) \text{ and } e_i \text{ is a set type,} \\ f'^A(a) \sim f'^A(b) & \text{otherwise.} \end{cases}$$

Remember that for all set types $set(s)$, $\{a_1,\dots,a_k\} \sim_{set(s)} \{b_1,\dots,b_l\}$ holds true iff for all $1 \le i \le k$ there is $1 \le j \le l$ with $a_i \sim_s b_j$ and for all $1 \le j \le l$ there is $1 \le i \le k$ with $a_i \sim_s b_j$.

By Lambek's Lemma, the H_Σ-coalgebra $\alpha : \nu\Sigma \to H_\Sigma(\nu\Sigma)$ (see (1)) is an isomorphism and thus for all $f : s \to e_1 + \cdots + e_n \in F$ and $t \in \nu\Sigma_s$,

$$f^{\nu\Sigma}(t) = \pi_f(t) = t(f).$$

Hence for all Σ-algebras A and $a \in A_s$, $f^A(a) = (b, i)$ implies

$$unfold^A(a)(f) = f^{\nu\Sigma}(unfold^A(a)) = unfold^A(f^A(a)) = unfold^A(b, i) = (unfold^A(b), i).$$

The carriers of $\nu\Sigma$ can be represented as equivalence classes of trees:

Let T be the greatest $\mathbb{FT}(S, BS)$-sorted set of finite or infinite trees t such that

- for all $X \in BS$, if $t \in T_X$, then t is a leaf labelled with some element of X,
- for all $s \in S$, if $t \in T_s$, then for all $f : s \to e_1 + \cdots + e_n \in F$ there are $1 \leq i \leq n$, $u \in T_{e_i}$ and a unique outarc of the root r of t that is labelled with (f, i) and points to the root of u and r has no other outarcs,
- for all collection types $c(s) \in \mathbb{FT}(S, BS)$, if $t \in T_{c(s)}$, then the root of t is labelled with c and the tuple of maximal proper subtrees of t is in T_s^*.

Hence for all $t \in T$,

- a node n is a leaf of t iff n is labelled with an element of some $X \in BS$,
- n is an inner node iff n is unlabelled or labelled with *word*, *bag* or *set*.

Let $\sim$ be the greatest equivalence relation on T such that for all $e \in \mathbb{FT}(S, BS)$, $t, u \in T_e$ and lists $t_1, \ldots, t_m$ and $u_1, \ldots, u_n$ of maximal proper subtrees of t resp. u, $t \sim u$ implies

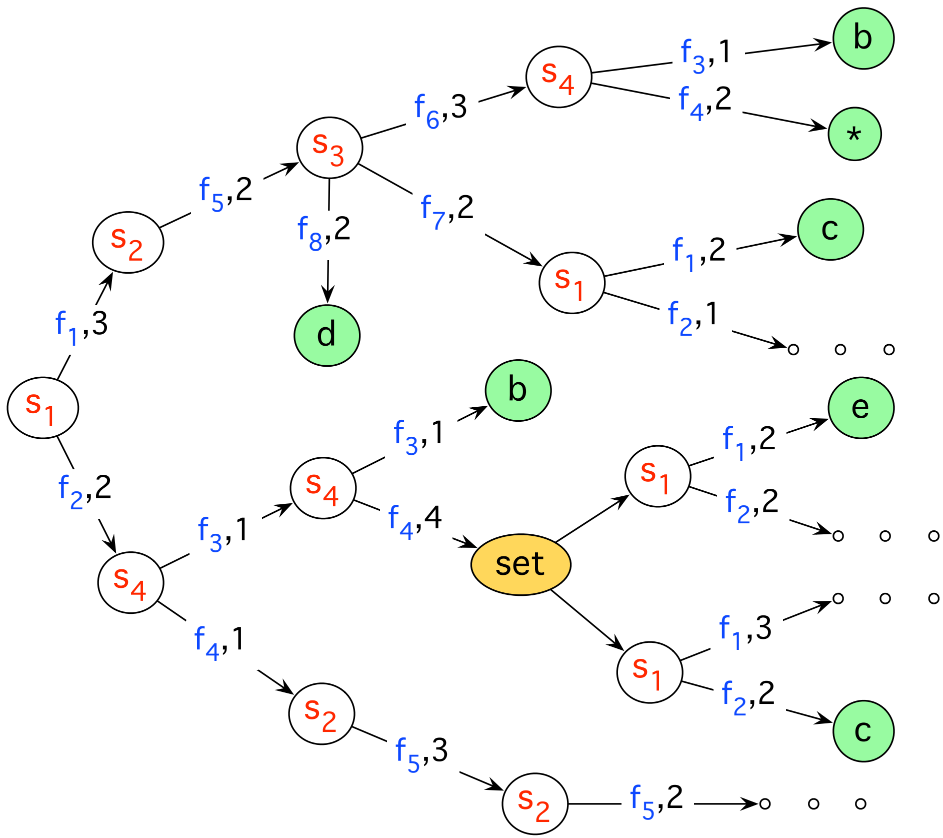
- $e \in S \cup BS$, $m = n$ and for all $1 \leq i \leq n$, $t_i \sim u_i$, or
- e is a word type, $m = n$ and for all $1 \leq i \leq n$, $t_i \sim u_i$, or
- e is a bag type, $m = n$ and there is a bijection h on $\{1, \ldots, n\}$ such that for all $1 \leq i \leq n$, $t_i \sim u_{h(i)}$, or
- e is a set type, for all $1 \leq i \leq m$ there is $1 \leq j \leq n$ with $t_i \sim u_j$ and for all $1 \leq j \leq n$ there is $1 \leq i \leq n$ with $t_i \sim u_j$.

For all $e \in \mathbb{FT}(S, BS)$, $\nu\Sigma_e \cong T_e/\sim$.

If F does not contain bag or set types, then $\nu\Sigma_e \cong T_e/\sim = T_e$.

The elements of $\nu\Sigma$ are called **ground Σ-coterms**.

For all $k \in \mathbb{N}$, $\mathcal{D}(k)$ is represented by the (equivalence classes of) finite ground Σ-coterms t with $depth(t) \leq k$.



A ground Σ-coterm with destructors $f_1, \ldots, f_8$ and base elements $a, b, c, d, e, *$.

Each inner node n is labelled with the sort of the subtree with root n.

Dots indicate infinite subtrees.

Examples

$A = \mathbb{N} \cup \{\infty\}$ is a final *coNat*-algebra: For all $n \in A$,

$$pred^A(n) = \begin{cases} * & \text{if } n = 0, \\ n-1 & \textit{if } n > 0, \\ \infty & \textit{if } n = \infty. \end{cases}$$

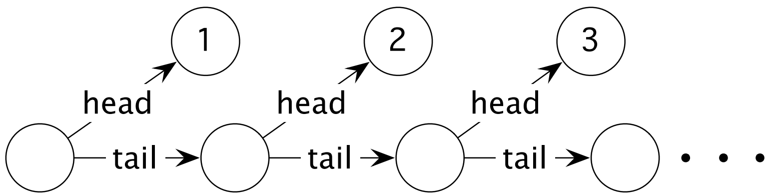
$Beh(X,Y)$ is final in $Alg_{DAut(X,Y)}$. In particular, the $DAut(1,Y)$-algebra of streams with elements from Y Y is final in $Alg_{DAut(1,Y)}$ and the $DAut(2,Y)$-algebra of infinite binary trees with node labels from Y is final in $Alg_{DAut(2,Y)}$.

Since $T = T_{Reg(X)}$ and $Lang$ are $DAut(X,2)$-algebras, $fold^{Lang} : T \to Lang$ is a $DAut(X,2)$-homomorphism (see [49], Section 12) and $Lang$ is a final $DAut(X,2)$-algebra, $fold^{Lang}$ coincides with $unfold^T$. This fact allows us to build a generic parser for all regular languages upon δ^T and β^T and to extend it to a generic parser for all context-free languages by simply incorporating the respective grammar rules (see ????[49], Sections 12 and 14).

For $\Sigma \in \{Stream(X), coList(X), Infbintree(X), coBintree(X,Y), coTree(X,Y), FBTree(X,Y)\}$, the elements of the *list*-, *btree*- resp. *tree*-carrier of $\nu\Sigma$ can be represented by the sequences resp. trees that we associated in Signatures with Σ.

This follows from a simple one-to-one transformation of the tree representation described above: Remove each edge e labelled with an **attribute**, i.e., a destructor $f : s \to B$ with $B \in BS$ and add the label $b \in B$ of the target of e to the label(s) of the source of e. Of course, if s has several attributes, it must be indicated that b was the value produced by f.

For instance, the usual sequence representation of the stream $[1, 2, 3, \dots]$ is obtained from the following tree representation:



Predicate coinduction

Let $\Sigma = (S, BS, F, P)$ be a signature and C be a Σ-algebra.

Predicate coinduction is an (analytical, top-down) inference rule that allows us to show that the interpretation of P' in $gfp(\Phi_{\Sigma',C,AX})$ contains all objects with some properties, given by Σ-formulas $\psi_p : e$, one for each $p : e \in P'$. The goals $psi_p \Rightarrow p$, $p \in P'$, are replaced by the axioms for P', which are then resolved upon the goals:

$$(1) \qquad \frac{\psi_p \Rightarrow p}{\bigwedge_{pt\Rightarrow\varphi\in AX}(\psi_p t \Rightarrow \varphi[\psi_p/p \mid p \in P'])} \quad \Uparrow$$

If further top-down rules (e.g. resolution and narrowing) transform the succedent of (1) to *True*, then by Lemma COIND, C satisfies the antecedent of (1).

Goals can often be proved by coinduction only after they have been *generalized*: Some formula δ_p must be found such that C satisfies $\psi_p \vee \delta_p \Rightarrow p$. The generalization weakens the coinduction conclusion in the succedent of (1) from $\varphi[\psi_p/p]$ to $\varphi[\psi_p \vee \delta_p/p]$.

In order to find δ_p, q_p and $q_p \Leftarrow \psi_p$ are added to Σ resp. AX when (1) is applied. The succedent of (1) is modified accordingly:

$$(2) \qquad \frac{\psi_p \Rightarrow p}{\bigwedge_{pt\Rightarrow\varphi\in AX}(\psi_p t \Rightarrow \varphi[q_p/p \mid p \in P'])}$$

If p is binary and AX includes congruence axioms for p, ψ_p is also binary and we add equivalence axioms for q_p to AX:

$$q_p\langle x,x\rangle, \quad q_p\langle x,y\rangle \Rightarrow q_p\langle y,x\rangle, \quad q_p\langle x,y\rangle \wedge q_p\langle y,z\rangle \Rightarrow q_p\langle x,z\rangle.$$

The demand for generalizing the goal $\psi_p \Rightarrow p$ becomes apparent in the course of proving the succedent of (2) when a subgoal of the form $q_p \Leftarrow \delta_p$ is encountered:

If $\delta_p = \psi_p$, then the subgoal is an axiom and thus reduces to *True*. Otherwise $q_p \Leftarrow \delta_p$ is added to AX and the proof proceeds with an application of the following rule:

$$(3) \qquad \frac{\delta_p \Rightarrow q_p}{\bigwedge_{pt\Rightarrow\varphi\in AX}(\delta_p t \Rightarrow \varphi[q_p/p \mid p \in P'])}$$

Between the applications of (2) resp. (3), resolution steps upon the added axiom $q_p \Leftarrow \psi_p$ must be confined to redex positions with positive polarity, i.e., the number of preceding negation symbols in the entire formula must be even. Otherwise the axiom added when (3) is applied might violate the soundness of the resolution steps.

Resolution upon q_p at any redex position becomes sound as soon as the set of axioms for q_p is not extended any more.

By inferring *True* from the conclusions of (2) and (3) one shows, roughly speaking, that the predicate $\psi_p \vee \delta_p$ solves the axioms for p. Since p itself represents the greatest solution, we conclude $\psi_p \vee \delta_p \Rightarrow p$, in particular the original goal $\psi_p \Rightarrow p$.

Predicate coinduction allows us to prove properties of *greatest* predicates. If, however, P' consists of least predicates, then proving goals of the form $\psi_p \Rightarrow p$ amounts to simply resolving them upon p.

The recent approach called coinductive logic or co-logic programming [27, 61] has not much to do with co/induction. It is rather co/resolution upon least resp. greatest predicates on models consisting of finite or infinite terms. In contrast to the above co/resolution rules, co-logic programming does not only resolve axioms upon (atoms of) the current goal φ, but also compares φ with all predecessors of φ in order to detect circularities in the derivation. We claim that most results obtained due to this – rather inefficient – inspection of the entire derivation would also be accomplished if the above co/induction rules were used instead.

Coinduction for proving equality

Let $P' = \{\sim_s : s \times s \mid s \in S\}$, $\Sigma' = (S, F, P + P')$,

$$AX = \{x \sim_e y \Rightarrow fx \sim_{e'} fy \mid f : e \to e' \in F\},$$

C be final in a full subcategory of Alg_Σ, R be an S-sorted binary relation on C and ψ be an S-sorted set of Σ-formulas such that for all $s \in S$, $\psi_s^C = R_s$. By Lemma MIN (1),

$$\begin{aligned} R \subseteq \Delta_C \iff & \text{ some } \Sigma\text{-congruence} \sim \text{contains } R \\ \Longleftarrow & \text{ the greatest } \Sigma\text{-congruence} \sim \text{contains } R \\ \Longleftarrow & \text{ the succedent of predicate coinduction is valid} \\ & \text{ for } P', AX \text{ and } \psi \text{ defined as above.} \end{aligned}$$

******

Suppose that for all $s \in S$, s-equality $=_s$: $s \times s$ belongs to P, and AX is a set of co-Horn clauses such that for all Σ-algebras A satisfying AX, $=^A = \{=_s^A \mid s \in S\}$ is a Σ-congruence.

Let $\nu\Sigma$ be final in $Alg^{\in}_{\Sigma,AX}$ or $gen(Alg^{=}_{\Sigma,AX})$ (see Thm. RESFIN resp. ABSFIN). Then $=^{\nu\Sigma}$ is the greatest Σ-congruence on $\nu\Sigma$ that satisfies AX.

Let R be an S-sorted binary relation on $\nu\Sigma$ and for all $s \in S$, $\psi_s : s \times s$ be a Σ-formula that describes R_s, i.e., R_s coincides with $\psi_s^{\nu\Sigma}$. By algebraic coinduction, $R \subseteq \Delta_{\nu\Sigma}$ if for all $s \in S$, $\psi_s^{\nu\Sigma} \subseteq =_s^{\nu\Sigma}$.

Bounded functors

Let $\alpha : A \to F(A)$ be an F-coalgebra and B be a subset of A. If the inclusion mapping $inc : B \to A$ is a $coAlg_F$-morphism from an F-coalgebra $\beta : B \to F(B)$ to α then β is an F**-invariant** or F**-subcoalgebra** of α.

Theorem ([30], Prop. 4.2.4 (i)) Every union or intersection of F-invariants is an F-invariant. Hence for all subsets of B of A there is a least F-invariant $\langle B\rangle : C \to F(C)$ such that C includes B. ❑

Let M be an S-sorted set. $F : Set^S \to Set^S$ is M**-bounded** if for all F-coalgebras $\alpha : A \to F(A)$ and $a \in A$, $|\langle a\rangle_s| \leq |M_s|$ (see [24], Section 4).

Let λ be a cardinal number. A category $\mathcal{I}$ is λ**-filtered** if for each class $\mathcal{L}$ of less than λ $\mathcal{I}$-objects there is a cocone $\{i \to j \mid i \in \mathcal{L}\}$ in $\mathcal{I}$ and for all $\mathcal{I}$-objects i, j and each set Φ of less than λ $\mathcal{I}$-morphisms from i to j there is a coequalizing $\mathcal{I}$-morphism $h : j \to k$, i.e., for all $f, g \in \Phi$, $h \circ f = h \circ g$.

A diagram $\mathcal{D} : \mathcal{I} \to \mathcal{K}$ is λ**-filtered** if $\mathcal{I}$ is a λ-filtered category.

A functor $F : \mathcal{K} \to \mathcal{L}$ is λ**-accessible** if F preserves the colimits of all λ-filtered diagrams $\mathcal{D} : \mathcal{I} \to \mathcal{K}$ (see [9], Section 5.2).

Theorem ([10], Thm. 4.1; [11], Thm. V.4)

Let M be an S-sorted set. $F : Set^S \to Set^S$ is M-bounded if F is $|M|$-accessible. Conversely, F is $(|M|+1)$-accessible if F is M-bounded. ❏

By [55], Thm. 10.6, or [24], Cor. 4.9 and Section 5.1, for each destructive signature Σ there is an S-sorted set M such that H_Σ is M-bounded (see Destructive-signature functors).

Examples

By [55], Ex. 6.8.2, or [24], Lemma 4.2, $H_{DAut(X,Y)}$ is X^*-bounded:

For all $DAut(X,Y)$-algebras A and $a \in A_{state}$,

$$\langle st \rangle = \{\delta^{A*}(a)(w),\ w \in X^*\}$$

where $\delta^{A*}(a)(\epsilon) = st$ and for all $x \in X$ and $w \in X^*$, $\delta^{A*}(a)(xw) = \delta^{A*}(\delta^A(a)(x))(w)$. Hence $|\langle st \rangle| \leq |X^*|$.

$H_{NDAut(X,Y)}$ is $(X^* \times \mathbb{N})$-bounded: For all $NDAut$-algebras A and $a \in A_{state}$,

$$\langle st \rangle = \cup\{\delta^{A*}(a)(w),\ w \in X^*\}$$

where $a \in A_{state}$, $\delta^{A*}(a)(\epsilon) = \{st\}$ and $\delta^{A*}(a)(xw) = \cup\{\delta^{A*}(st')(w) \mid st' \in \delta^A(a)(x)\}$ for all $x \in X$ and $w \in X^*$. Since for all $a \in A_{state}$ and $x \in X$, $|\delta^A(a)(x)| \in \mathbb{N}$, $|\langle st \rangle| \leq |X^* \times \mathbb{N}|$. If $X = 1$, then $X^* \times \mathbb{N} \cong \mathbb{N}$ and thus $H_{NDAut(1,Y)}$ is $\mathbb{N}$-bounded (see [55], Ex. 6.8.1; [24], Section 5.1). ❏

A destructive signature $\Sigma = (S, BS, F, P)$ is **Moore-like** if there is an S-sorted set M such that for all $f : s \to e \in F$, $e = s^{M_s}$ or $e \in BS$. Then M is called the **input** of Σ.

Lemma MOORE

Let $\Sigma = (S, BS, F, P)$ be a Moore-like signature with input M and

$$F' = \{f : s \to e \mid e \in BS\}.$$

Σ is polynomial and thus Alg_Σ has a final object A.

Let $Y = \prod_{f:s\to e\in F'} e$. If $|S| = 1$, then Σ agrees with $DAut(M_s, Y)$ and thus $A \cong Beh(M_s, Y)$. Otherwise A can be constructed as a straightforward extension of $Beh(M_s, Y)$ to several sorts: For all $s \in S$ and $h \in A_s$,

$$A_s = M_s^* \to Y,$$

for all $f : s \to e \in F'$, $f^A(h) = \pi_g(h(\epsilon))$ and for all $f : s \to s^{M_s}$, $f^A(h) = \lambda x.\lambda w.h(xw)$.

A can be visualized as the S-sorted set of trees such that for all $s \in S$ and $h \in A_s$, the root r of h has $|M_s|$ outarcs, for all $f : s \to e \in F'$, r is labelled with $f^A(h)$, and for all $f : s \to s^{M_s}$ and $x \in M_s$, $f^A(h)(x) = \lambda w.h(xw)$ is the subtree of h where the x-th outarc of r points to. ❑

Theorem MOORETAU

Let $\Sigma = (S, BS, F, P)$ be a destructive signature, M be an S-sorted set, H_Σ be M-bounded and

$$F' = \{f_s : s \to s^{M_s} \mid s \in S\} \cup \{f' : s \to M_e \mid f : s \to e \in F\}.$$

Let $\Sigma' = (S, BS \cup \{M_e \mid e \in \mathbb{T}(S, BS)\}, F', R)$ and $\tau : H_{\Sigma'} \to H_\Sigma$ be the function defined as follows: For all S-sorted sets A, $a \in H_{\Sigma'}(A)_s$ and $f : s \to e \in F$,

$$\pi_f(\tau_{A,s}(a)) = F_e(\pi_{f_s}(a))(\pi_{f'}(a)).$$

τ is a surjective natural transformation.

Proof. The theorem generalizes [24], Thm. 4.7 (i)⇒(iv), from Set to Set^S. ❑

Theorem BFIN

Let $\Sigma = (S, BS, F, P)$ be a destructive signature, M be an S-sorted set, H_Σ be M-bounded and the Σ-algebra A be defined as follows: For all $s \in S$,

$$A_s \;=\; M_s^* \to \prod_{f:s\to e\in F} M_e,$$

and for all $f : s \to e \in F$ and $h \in A_s$,

$$f^A(h) = F_e(\lambda x.\lambda w.h(xw))(\pi_f(h(\epsilon))).$$

A is weakly final and $A/{\sim}$ is final in Alg_Σ where $\sim$ is the greatest Σ-congruence on A.

Proof. Let Σ' and τ be defined as in Theorem MOORETAU. Let $Y = \prod_{f':s\to M_e\in F'} M_e$. Since Σ' is Moore-like, Lemma MOORE implies that the following Σ'-algebra B is final:

For all $s \in S$, $B_s \;=\; M_s^* \to Y$.

For all $f : s \to e \in F$ and $h \in B_s$, $f_s^B(h) = \lambda x.\lambda w.h(xw)$ and $f'^B(h) = \pi_{f'}(h(\epsilon))$.

Hence by Lemma WEAKFIN, A is weakly final:

For all $s \in S$, $\prod_{f:s \to e \in F} M_e = Y$ and thus $A_s = B_s$.

For all $f : s \to e \in F$ and $h \in A_s$,

$$\begin{aligned} f^A(h) &= F_e(\lambda x.\lambda w.h(xw))(\pi_f(h(\epsilon))) = F_e(\lambda x.\lambda w.h(xw))(\pi_{f'}(h(\epsilon))) \\ &= F_e(f_s^A(h))(f'^A(h)) = F_e(\pi_{f_s}(g_1(h),\dots,g_n(h)))(\pi_{f'}(g_1(h),\dots,g_n(h))) \\ &= \pi_f(\tau_{A,s}(g_1(h),\dots,g_n(h))) = \pi_f(\tau_{A,s}(\langle g_1,\dots,g_n\rangle(h))) = f^B(h) \end{aligned}$$

where $\{g_1,\dots,g_n\} = \{g^A \mid g : s \to e' \in F'\}$.

Hence again by Lemma WEAKFIN, $A/\!\sim$ is final in Alg_Σ where $\sim$ is the greatest Σ-congruence on A.

A direct proof of the existence of a final Σ-algebra is given by [25], Thm. 3.5. ❏

Example

Let $\Sigma = NDAut(X,Y)$, i.e., $S = \{state\}$, $BS = \{X,Y\}$,

$$F = \{\delta : state \to set(state)^X,\ \beta : state \to Y\}$$

and $P = \emptyset$, and $M_{state} = X^* \times \mathbb{N}$. Hence $M_{set(state)^X} = \mathcal{P}_{fin}(M)^X$ and $M_Y = Y$. Since H_Σ is M-bounded, Theorem BFIN implies that the following Σ-algebra A is weakly final:

$$A_{state} \;=\; M^* \to \mathcal{P}_{fin}(M)^X \times Y.$$

For all $h \in A_{state}$ and $x \in X$, $h(\epsilon) = (g,y)$ implies

$$\begin{aligned}
\delta^A(h)(x) &= F_{set(state)^X}(\lambda m.\lambda w.h(mw))(\pi_\delta(h(\epsilon)))(x)\\
&= F_{set(state)^X}(\lambda m.\lambda w.h(mw))(g)(x) = F_{set(state)}(\lambda m.\lambda w.h(mw))(g(x))\\
&= \{F_{state}(\lambda m.\lambda w.h(mw))(m) \mid m \in g(x)\}\\
&= \{\lambda m.\lambda w.h(mw))(m) \mid m \in g(x)\} = \{\lambda w.h(mw) \mid m \in g(x)\},\\
\beta^A(h) &= F_Y(\lambda x.\lambda w.h(xw))(\pi_\beta(h(\epsilon))) = F_Y(\lambda x.\lambda w.h(xw))(y) = id_Y(y) = y.
\end{aligned}$$

Moroever, $A/\!\sim$ is final in Alg_Σ where $\sim$ is the greatest Σ-congruence on A, i.e., the union of all S-sorted binary relations $\sim$ on A such that for all $h,h' \in A_{state}$,

$$h \sim h' \;\text{ implies }\; \delta^A(h) \;\sim_{set(state)^X}\; \delta^A(h') \;\wedge\; \beta^A(h) \;\sim_Y\; \beta^A(h'),$$

i.e., for all $x \in X$, $h \sim h'$, $h(\epsilon) = (g, y)$ and $h'(\epsilon) = (g', y')$ imply

$$\forall\ m \in g(x)\ \exists n \in g'(x) : \lambda w.h(mw) \sim \lambda w.h'(nw)\ \wedge$$
$$\forall\ n \in g'(x)\ \exists m \in g(x) : \lambda w.h(mw) \sim \lambda w.h'(nw)\ \wedge\ y = y'.$$

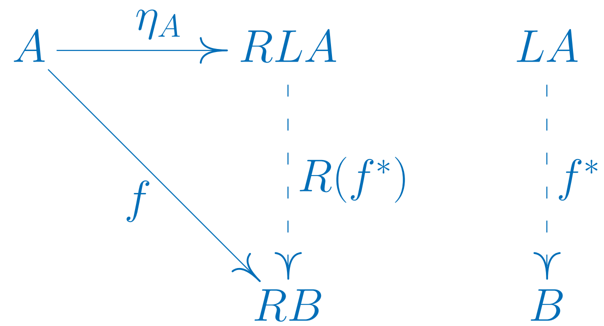
Let $F' = \{f : state \to state^M,\ \delta : state \to \mathcal{P}_{fin}(M)^X,\ \beta : state \to Y\}$ and $\Sigma' = (S, \{X, Y, M, \mathcal{P}_{fin}(M)^X\}, F', P)$.

A is constructed from the following Σ'-algebra B with $B_{state} = A_{state}$ (see the proof of Theorem BFIN): For all $h \in A_{state}$, $f^B_{state}(h) = \lambda m.\lambda w.h(mw)$ and $\langle \delta^B, \beta^B \rangle(h) = h(\epsilon)$.

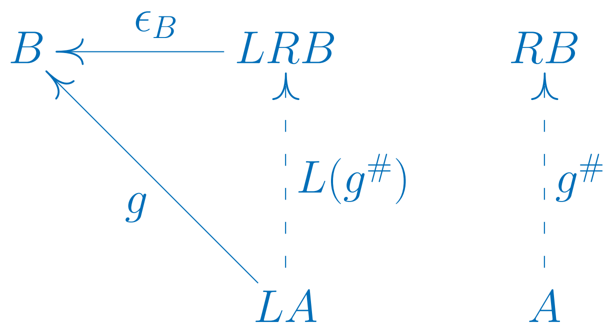
Since Σ' is Moore-like, Lemma MOORE implies that A can be visualized as the set of trees h such that the root r of h has $|M|$ outarcs, r is labelled with $h(\epsilon)$ and for all $m \in M$, $\lambda w.h(mw)$ is the subtree of h where the m-th outarc of r points to. [26], Section 5, shows (for the case $X = Y = 1$) how these trees yield the quotient $A/\sim$. ❏

Adjunctions

An **adjunction** is a quadruple (L, R, η, ϵ) consisting of functors $L : \mathcal{K} \to \mathcal{L}$, $R : \mathcal{L} \to \mathcal{K}$ and natural transformations $\eta : Id_{\mathcal{K}} \to RL$ and $\epsilon : LR \to Id_{\mathcal{L}}$ such that for each $\mathcal{K}$-morphism $f : A \to R(B)$ there is a unique $\mathcal{L}$-morphism $f^* : LA \to B$, called the **$\mathcal{K}$-extension of f**, such that the following diagram commutes:



or for each $\mathcal{L}$-morphism $g : L(A) \to B$ there is a unique $\mathcal{K}$-morphism $g^\# : A \to RB$, called the **$\mathcal{L}$-extension of g**, such that the following diagram commutes:



η is the **unit** (or **inclusion of generators**) and ϵ the **co-unit** (or **evaluation**) of the adjunction.

η exists if and only if ϵ exists.

For all $B \in \mathcal{L}$, $R\epsilon_B \circ \eta_{RB} = id_{RB}$.

Hence by the uniqueness of $\mathcal{K}$-extensions,
$\epsilon_B = id^*_{RB}$ and for all $f \in \mathcal{K}(A, B)$, $Lf = (\eta_B \circ f)^*$.

For all $A \in \mathcal{K}$, $\epsilon_{LA} \circ L\eta_A = id_{LA}$.

Hence by the uniqueness of $\mathcal{L}$-extensions,
$\eta_A = id^\#_{LA}$ and for all $g \in \mathcal{L}(A, B)$, $Rg = (g \circ \epsilon_B)^\#$.

L is the **left adjoint** of R. R is the **right adjoint** of L. We write $L \dashv R$.

Left adjoints preserves colimits. Right adjoints preserves limits.

$L \dashv R$ iff

$\mathcal{K}(\_, R(\_))$ and $\mathcal{L}(\_, L(\_))$ are naturally equivalent functors from $\mathcal{K}^{op} \times \mathcal{L}$ to Set,

i.e.,

- for all $A \in \mathcal{K}$ and $B \in \mathcal{L}$,
$$\mathcal{K}(A, RB) \cong \mathcal{L}(LA, B),$$
- for all $f \in \mathcal{K}(A', A)$ and $g \in \mathcal{L}(B, B')$, the following diagram commutes:

$$\begin{array}{ccc}
\mathcal{K}(A', RB) & \cong & \mathcal{L}(LA', B) \\
\mathcal{K}(f, Rg) =_{def} \Big\downarrow \lambda h.(Rg \circ h \circ f) & & \mathcal{L}(Lf, g) =_{def} \Big\downarrow \lambda h.(g \circ h \circ Lf) \\
\mathcal{K}(A, RB') & \cong & \mathcal{L}(LA, B')
\end{array}$$

Examples of adjunctions $L \dashv R$

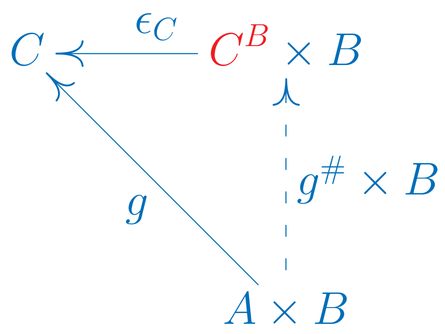
Identity functors are left and right adjoints

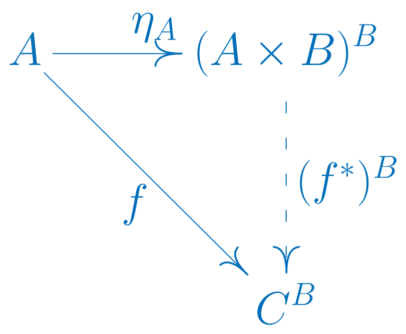
$L = R = Id_{\mathcal{K}}$.

Exponentials are right adjoints

A category $\mathcal{K}$ is **Cartesian closed** if $\mathcal{K}$ has a final object, binary products and for all $B \in \mathcal{K}$ there is an adjunction $(L : \mathcal{K} \to \mathcal{K}, R : \mathcal{K} \to \mathcal{K}, \eta, \epsilon)$ such that for all $A \in \mathcal{K}$ and $\mathcal{K}$-morphisms f, $L(A) = A \times B$ and $L(f) = f \times B$.

For all $A \in \mathcal{K}$, $R(A)$ is denoted by A^B and called an **exponential**.





Set^S is Cartesian closed:

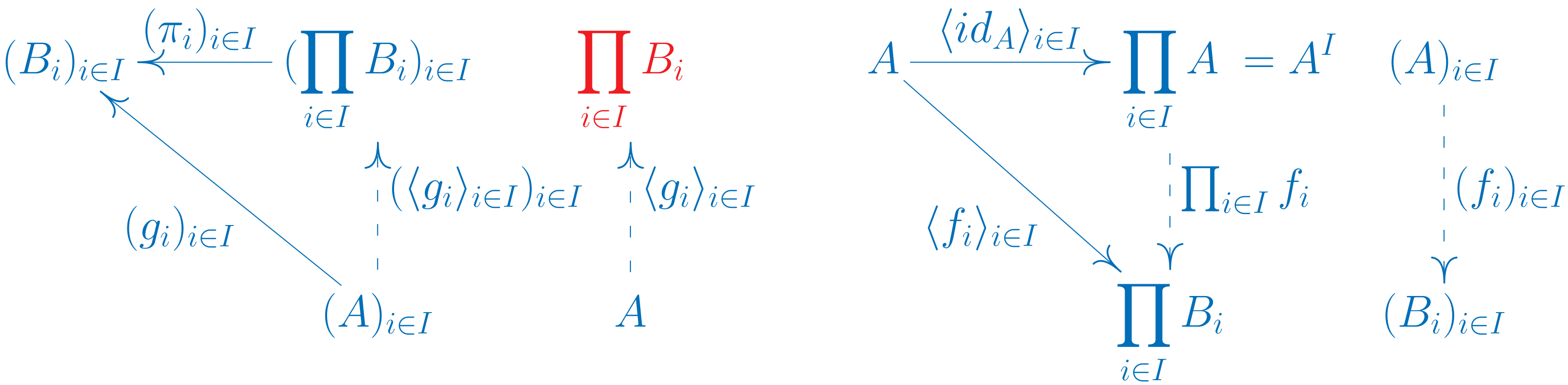
Let B be an S-sorted set.

- For all S-sorted sets A, $R(C) = C^B = Set^S(B, A)$,
- For all S-sorted functions $f : A \to C$ and $g : B \to A$, $R(f)(g) = f^B(g) = f \circ g$.
- For all S-sorted sets C, $\epsilon_C = \lambda(f, b).f(b)$.
- For all S-sorted sets A, $\eta_A = \lambda a.\lambda b.(a, b)$.
- For all S-sorted functions $g : A \times B \to C$, $g^\# = \lambda a.\lambda b.g(a, b)$.
- For all S-sorted functions $f : A \to C^B$, $f^* = f \circ \pi_1$.

Products are right adjoints

Let I be an index set, $\mathcal{K}$ be a category with I-indexed products,

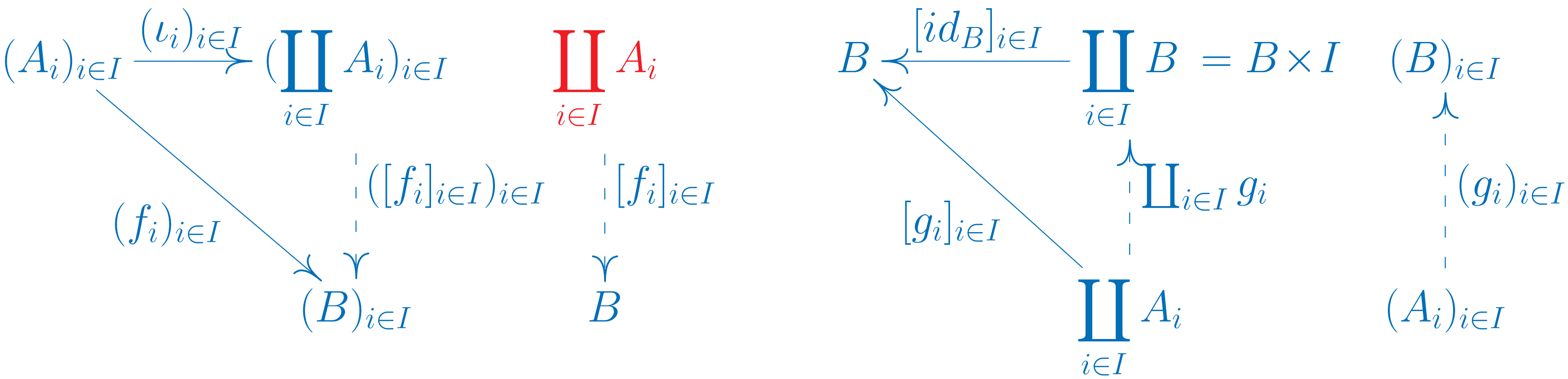
- $L : \mathcal{K} \to \mathcal{K}^I$ be the **diagonal functor** defined by $L(A)_i = A$ for all $\mathcal{K}$-objects and $\mathcal{K}$-morphisms A and $i \in I$,
- $R : \mathcal{K}^I \to \mathcal{K}$ defined by $R((B_i)_{i\in I}) = \prod_{i\in I} B_i$ for all $\mathcal{K}^I$-objects and $\mathcal{K}^I$-morphisms $(B_i)_{i\in I}$.



Coproducts are left adjoints

Let I be an index set, $\mathcal{L}$ be a category with I-indexed coproducts,

- $R : \mathcal{L} \to \mathcal{L}^I$ be the diagonal functor defined by $R(A)_i = A$ for all $\mathcal{L}$-objects and $\mathcal{L}$-morphisms A and $i \in I$,
- $L : \mathcal{L}^I \to \mathcal{L}$ defined by $L((A_i)_{i\in I}) = \coprod_{i\in I} A_i$ for all $\mathcal{L}^I$-objects and $\mathcal{L}^I$-morphisms $(A_i)_{i\in I}$.



Term adjunction (see The initial model of a flat constructive signature)

Let $\Sigma = (S, BS, F, P)$ be a flat constructive signature, $\mu\Sigma$ be initial in Alg_Σ, V be an S-sorted set of **variables**,

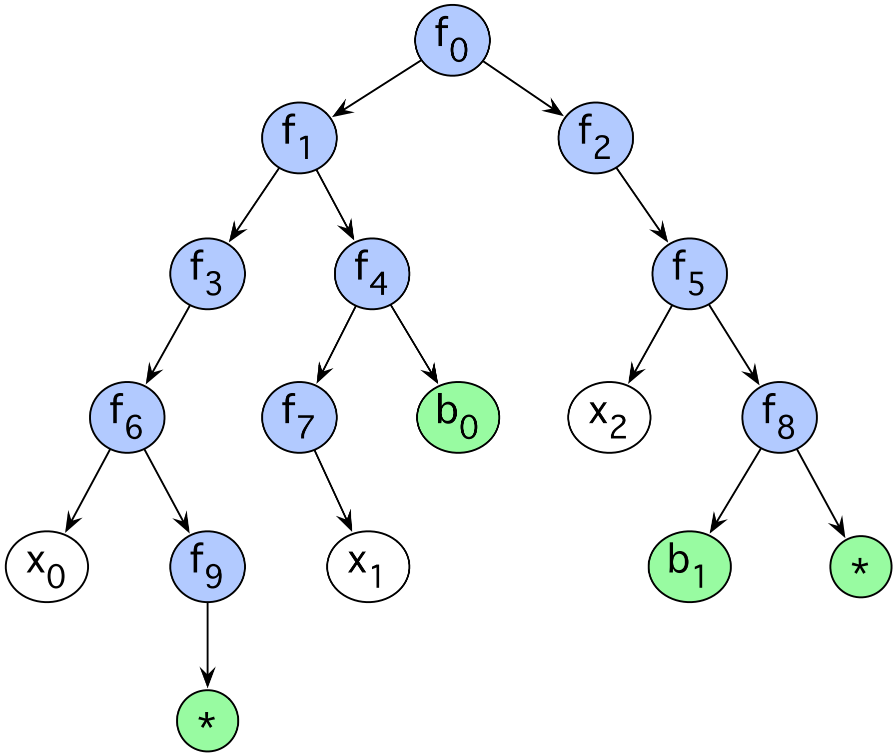
$$F' = \{in_s : V_s \to s \mid s \in S\} \quad and \quad \Sigma(V) = (S, BS \cup V, F \cup F', P).$$

The initial $\Sigma(V)$-algebra $\mu\Sigma(V)$ is called the **free Σ-algebra over V**. The Σ-reduct of $\mu\Sigma(V)$ is denoted by $T_\Sigma(V)$.

$T_\Sigma(\emptyset) \cong \mu\Sigma$ where $\emptyset$ denotes the S-sorted V with $V_s = \emptyset$ for all $s \in S$.

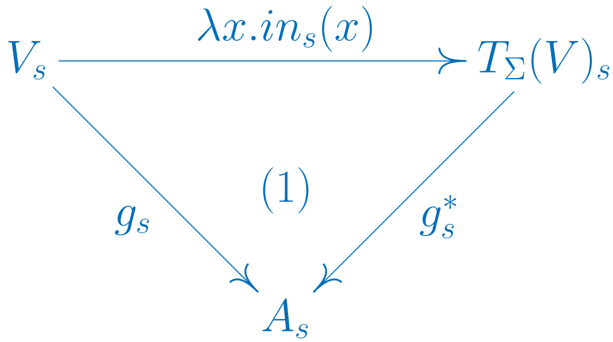
The elements of $T_\Sigma(V)$ are called **Σ-terms over V**.

In the tree representation of a Σ-term, we identify each node labelled with in_s, $s \in S$, and its respective successor.



A Σ*-term over* $\{x_0, x_1, x_2\}$ *with base elements* $b_0, b_1, *$

Let A be a Σ-algebra and g be a **valuation** of V in A, i.e., an S-sorted function from V to A. Then there is a unique Σ-homomorphism $g^* : T_\Sigma(V) \to A$ such that for all $s \in S$ the following diagram commutes:



Proof. A becomes a $\Sigma(V)$-algebra by defining $in^A_s = g_s$ for all $s \in S$. Hence there is a unique $\Sigma(V)$-homomorphism $fold^A$ from $\mu\Sigma(V)$ to A.

Let $h : T_\Sigma(V) \to A$ be a Σ-homomorphism satisfying (1), i.e., for all $s \in S$,

$$h_s \circ \lambda x.in_s(x) = g_s.$$

Then for all $x \in V_s$, $h(in^{\mu\Sigma(V)}_s(x)) = h(in_s(x)) = g(x) = in^A_s(x)$, i.e., h is compatible with F'. We conclude $h = fold^A$. ❏

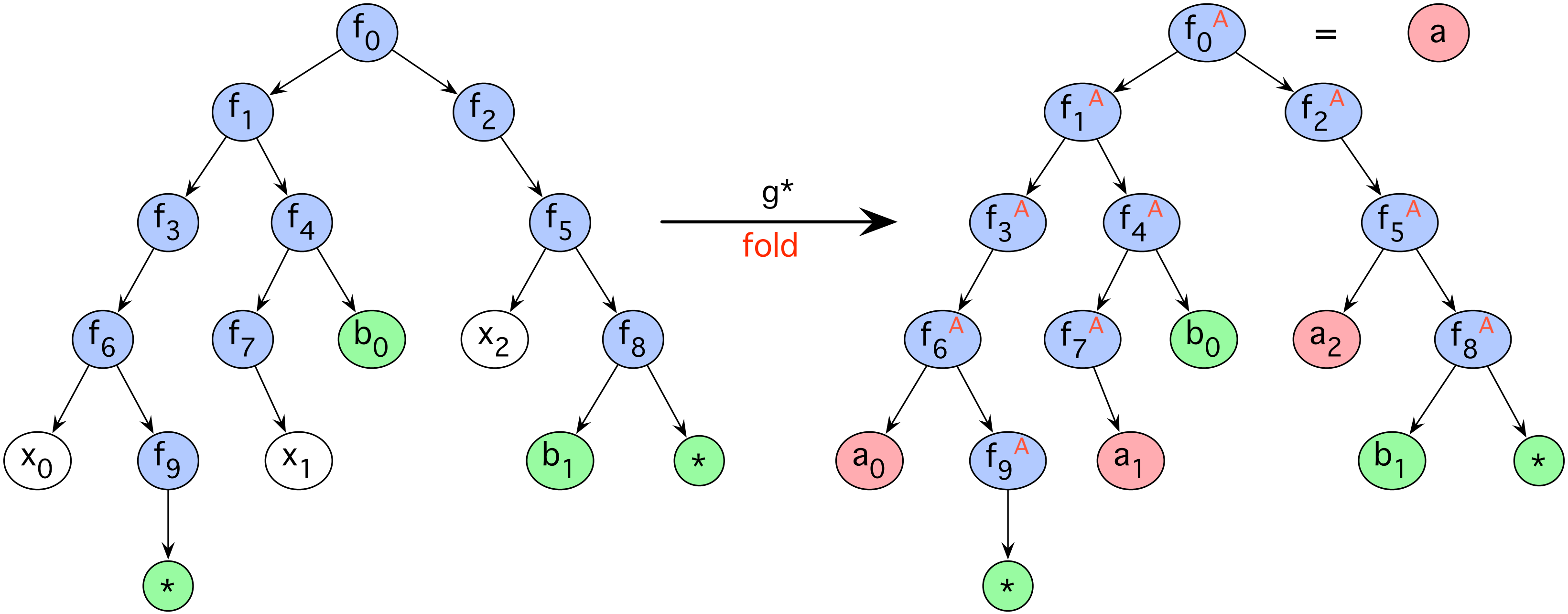
For all $s \in S$ and $x \in V$, $g^*(in_s(x)) = g(x)$.

Since g^* is Σ-homomorphic, for all $f : e \to s \in F$ and $(t_1, \ldots, t_n) \in T_{\Sigma,e}$,

$$g^*(f(t_1, \ldots, t_n)) = g^*(f^{\mu\Sigma}(t_1, \ldots, t_n)) = f^A(g^*(t_1), \ldots, g^*(t_n)).$$

g^* evaluates terms into algebra elements:

$g^* = fold^A : T_\Sigma(V) \to A$ takes a term $t \in \mu\Sigma(V)$, replaces each occurrence of a variable $x \in V$ in t by the value $g(x)$ and folds ("evaluates") the resulting term into an element of A.



Evaluation of a Σ*-term w.r.t. the valuation* $g = \lambda x_i . a_i$

By the uniqueness of g^*, the functor

$$\begin{array}{rcl} T_\Sigma : Set^S & \rightarrow & Alg_\Sigma \\ V & \mapsto & T_\Sigma(V) \\ h : V \rightarrow V' & \mapsto & \{(\lambda x.in_s(x) \circ h_s)^* : T_\Sigma(V)_s \rightarrow T_\Sigma(V')_s \mid s \in S\} \end{array}$$

is the left adjoint of the forgetful functor $U_S : Alg_\Sigma \rightarrow Set^S$

and the following lemma holds true:

Lemma EVAL

For all S-sorted functions $g : V \rightarrow A$ and Σ-homomorphisms $h : A \rightarrow B$,

$$(h \circ g)^* = h \circ g^*.$$

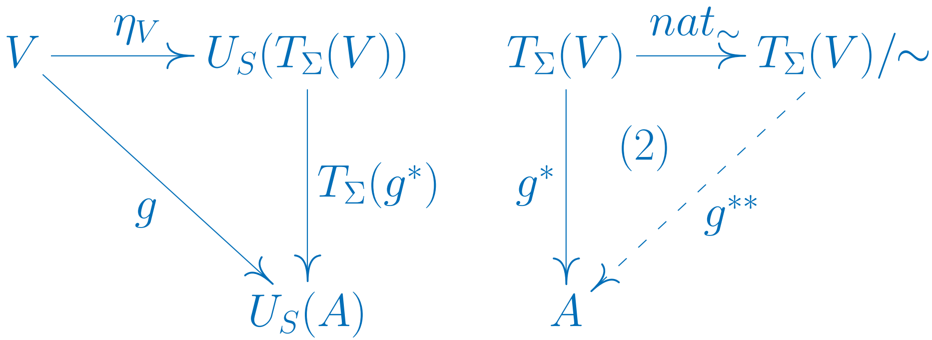
For all S-sorted sets V and $s \in S$, $\eta_{V,s} = \lambda x.in_s(x)$ (see (1)).

Let A be a Σ-algebra. The co-unit $\epsilon_A = id_A^* : T_\Sigma(U_S(A)) \rightarrow A$ takes a term t with "variables" in A and folds ("evaluates") t into an element of A.

Variety

Let $\sim$ be a Σ-congruence on $T_\Sigma(V)$.

A subcategory $\mathcal{K}$ of Alg_Σ is a **Σ-variety** if for all $A \in \mathcal{K}$ and all S-sorted functions $g : V \to U_S(A)$, g^* factorizes through $T_\Sigma(V)/{\sim}$:



Let $A \in \mathcal{K}$ and $g : V \to U_S(A)$ be an S-sorted function. If

$$\sim \text{ is a subset of the kernel of } g^*, \tag{3}$$

then g^{**} is well-defined by $g^{**}([t]_\sim) = g^*(t)$ for all $t \in T_\Sigma(V)$.

Since $nat_\sim$ is epi and predicate preserving, Lemma EMH (1) and the uniqueness of g^* imply that (3) is equivalent to the existence and uniqueness of g^{**} with (2).

Hence, if $T_\Sigma(V)/\!\sim\, \in \mathcal{K}$, then the forgetful functor from $\mathcal{K}$ to Set^S has a left adjoint with unit $nat_\sim \circ \eta_V$ and extension g^{**} of g.

$T_\Sigma(V)/\!\sim$ is called the **free $\mathcal{K}$-object over V**.

In particular, $T_\Sigma(\emptyset)/\!\sim$ is initial in $\mathcal{K}$.

Birkhoff Theorem I

A class of Σ-algebras is a Σ-variety iff it is closed under the formation of subalgebras, homomorphic images and products. ❏

Coterm adjunction (see The final model of a flat destructive signature)

Let $\Sigma = (S, BS, F, P)$ be a flat destructive signature, $\nu\Sigma$ be final in Alg_Σ, V be an S-sorted set of **covariables**,

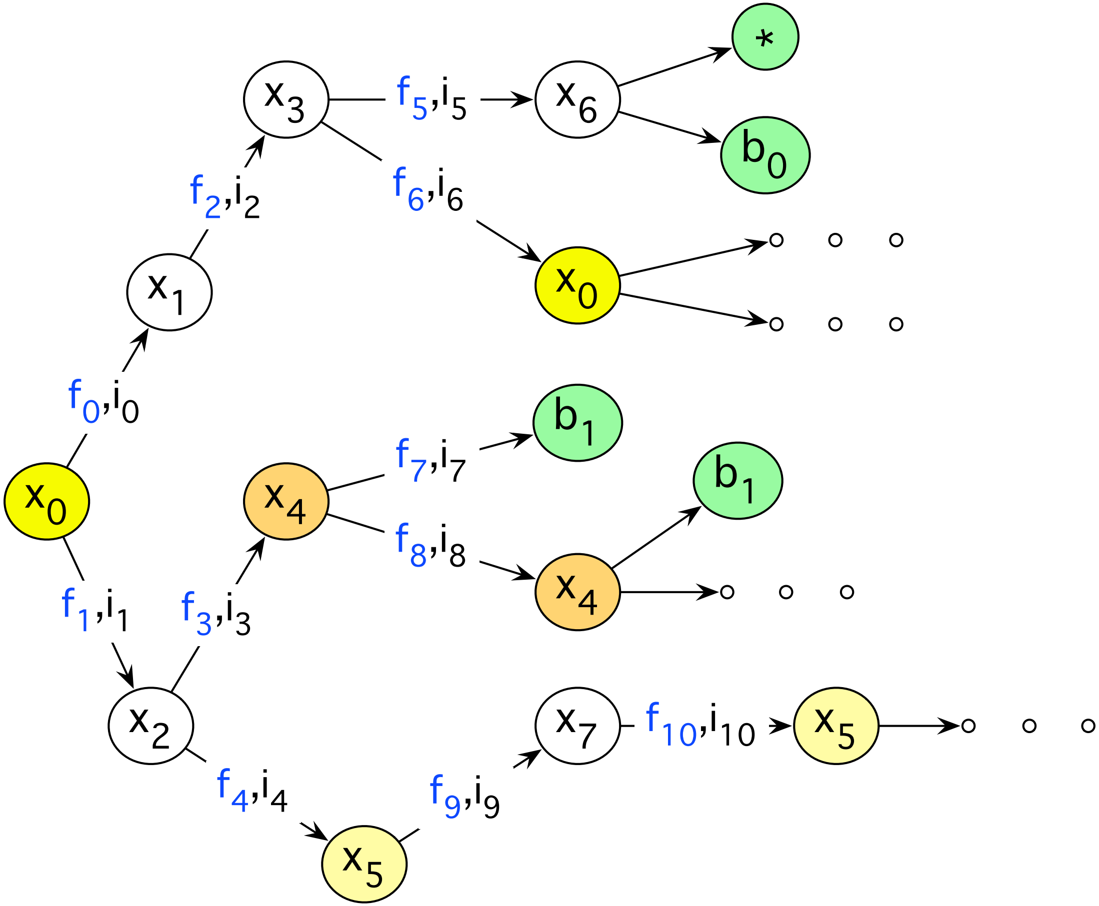
$$F' = \{out_s : s \to V_s \mid s \in S\} \quad and \quad \Sigma(V) = (S, BS \cup V, F \cup F', P).$$

The final $\Sigma(V)$-algebra $\nu\Sigma(V)$ is called the **cofree Σ-algebra over** V. The Σ-reduct of $\nu\Sigma(V)$ is denoted by $coT_\Sigma(V)$.

$coT_\Sigma(1) \cong \nu\Sigma$ where 1 denotes the S-sorted V with $V_s = 1$ for all $s \in S$.

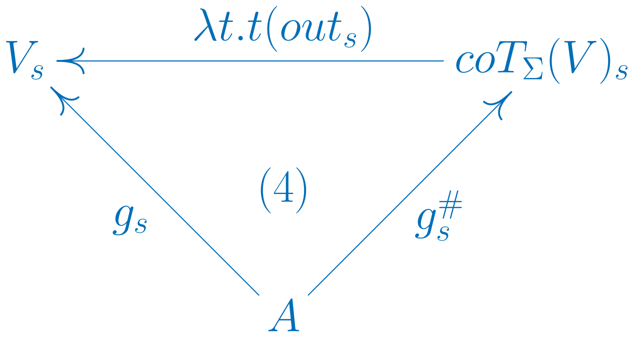
The elements of $T_\Sigma(V)$ are called **Σ-coterms over** V.

In the tree representation of a Σ-coterm, we identify each node labelled with out_s, $s \in S$, and its respective successor.



A Σ-coterm over $\{x_0, \ldots, x_7\}$ with base elements $b_0, b_1, *$

Let A be a Σ-algebra and g be a **coloring** of A by V, i.e., an S-sorted function from A to V. Then there is a unique Σ-homomorphism $g^\# : A \to coT_\Sigma(V)$ such that for all $s \in S$ the following diagram commutes:



Proof. A becomes a $\Sigma(V)$-algebra by defining $out_s^A = g_s$ for all $s \in S$. Hence there is a unique $\Sigma(V)$-homomorphism $unfold^A$ from A to $\nu\Sigma(V)$.

Let $h : A \to coT_\Sigma(V)$ be a Σ-homomorphism satisfying (1), i.e., for all $s \in S$,

$$\lambda t.t(out_s) \circ h_s = g_s.$$

Then for all $a \in A_s$, $out_s^{\nu\Sigma(V)}(h(a)) = h(a)(out_s) = (\lambda t.t(out_s))(h(a)) = g(a) = out_s^A(a)$, i.e., h is compatible with F'. We conclude $h = unfold^A$. ❑

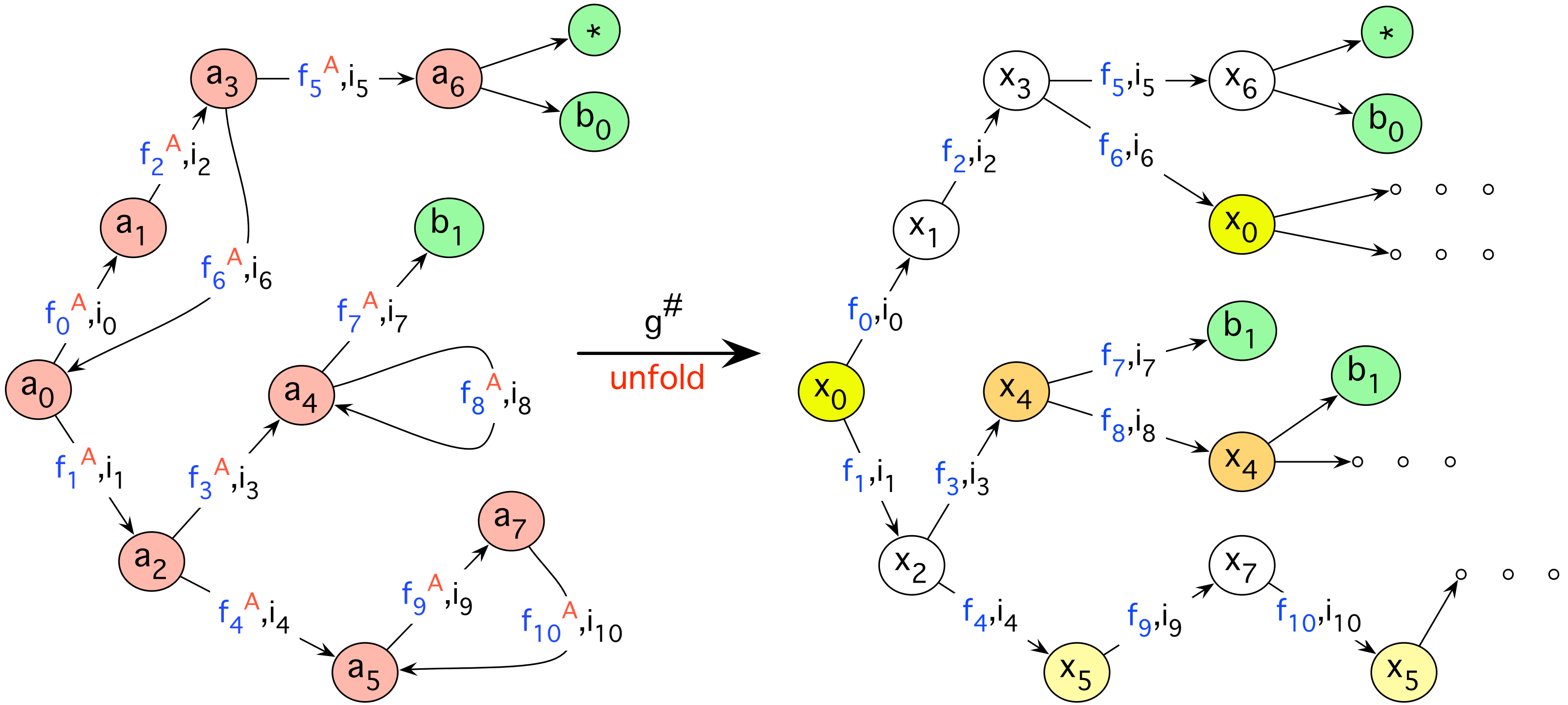
For all $s \in S$ and $a \in A_s$, $g^\#(a)(out_s) = (\lambda t.t(out_s))(g^\#(a)) = g(a)$.

Since $g^\#$ is Σ-homomorphic, for all $f : s \to e \in F$ and $a \in A_s$,

$$g^\#(a)(f) = f^{\nu\Sigma}(g^\#(a)) = g^\#(f^A(a)).$$

$g^\#$ observes the behavior of algebra elements:

$g^\# = unfold^A : A \to coT_\Sigma(V)$ takes $a \in A$, unfolds a into the behavior t of a and labels (the root of each) subtree u of t with the color $g(b)$ of some $b \in A$ with behavior u.



Observation of the behavior of a_0 *w.r.t. the coloring* $g = \lambda a_i.x_i$

By the uniqueness of $g^\#$, the functor

$$\begin{array}{rcl} coT_\Sigma : Set^S & \to & Alg_\Sigma \\ V & \mapsto & coT_\Sigma(V) \\ h : V \to V' & \mapsto & \{(h_s \circ \lambda t.t(out_s))^\# : coT_\Sigma(V)_s \to coT_\Sigma(V')_s \mid s \in S\} \end{array}$$

is the right adjoint of the forgetful functor $U_S : Alg_\Sigma \to Set^S$

and the following lemma holds true:

Lemma COEVAL

For all S-sorted functions $g : A \to V$ and Σ-homomorphisms $h : B \to A$,

$$(g \circ h)^\# = g^\# \circ h.$$

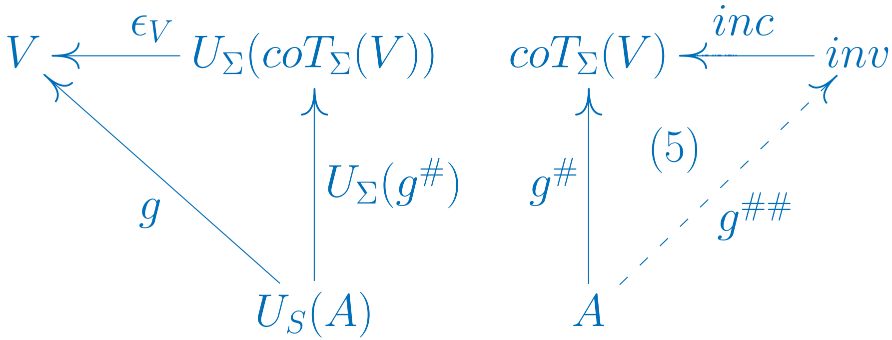
For all S-sorted sets V and $s \in S$, $\epsilon_{V,s} = \lambda t.t(out_s)$ (see (4)).

Let A be a Σ-algebra. The unit $\eta_A = id_A^\# : A \to coT_\Sigma(U_S(A))$ takes $a \in A$ and unfolds a into the behavior (tree) of a.

Covariety

Let inv be a Σ-invariant of $coT_\Sigma(V)$.

A subcategory $\mathcal{K}$ of Alg_Σ is a **Σ-covariety** if for all $A \in \mathcal{K}$ and all S-sorted functions $g : U_S(A) \to V$, $g^\#$ factorizes through inv:



Let $A \in \mathcal{K}$ and $g : U_S(A) \rightarrow V$ be an S-sorted function. If

$$\text{the image of } g^\# \text{ is a subset of } inv, \tag{6}$$

then $g^{\#\#}$ is well-defined by $g^{\#\#}(a) = g^\#(a)$ for all $a \in A$.

Since inc is mono and predicate preserving, Lemma EMH (2) and the uniqueness of $g^\#$ imply that (6) is equivalent to the existence and uniqueness of $g^{\#\#}$ with (5).

Hence, if $inv \in \mathcal{K}$, then the forgetful functor from $\mathcal{K}$ to Alg_Σ has a right adjoint

with co-unit $\epsilon_V \circ inc$ and extension $g^{\#\#}$ of g.

inv is called the **cofree $\mathcal{K}$-object over V**.

In particular, if $V = 1$, then inv is final in $\mathcal{K}$.

Birkhoff Theorem II

A class of Σ-algebras is a Σ-covariety iff it is closed under the formation of subalgebras, homomorphic images and coproducts. ❏

Base extensions: Base algebras as base sets

Let $\Sigma' = (S', BS', F', P')$ be a signature, $\Sigma = (S, BS, F, P)$ be a subsignature of Σ and B be a Σ-algebra.

For all $e \in \mathbb{T}(S, BS)$, $e_B \in \mathbb{T}(S' \setminus S)$ is obtained from e by replacing each sort $s \in S$ with B_s. Let $F_B = \{f_B : e_B \to e'_B \mid f : e \to e' \in F'\}$, $P_B = \{p_B : e_B \mid p : e \in P'\}$,

$$\Sigma_B = (S' \setminus S, BS' \cup B, F_B, P_B)$$

and $\sigma_B : \Sigma' \to \Sigma_B$ be the signature morphism that maps $s \in S$ to B_s, $s \in S' \setminus S$ to s and $f \in F' \cup P'$ to f_B. Then for all Σ_B-algebras A and $s \in S$,

$$(A|_{\sigma_B})_s = F_{\sigma_B(s)}(A) = \begin{cases} F_{B_s}(A) & = B_s \ \text{ if } s \in S, \\ F_s(A) & = A_s \ \text{ otherwise.} \end{cases}$$

Let U_Σ denote the forgetful functor from $Alg_{\Sigma'}$ to Alg_Σ.

Let A be a Σ'-algebra and $B = U_\Sigma(A)$. A yields a Σ_B-algebra $A_{\Sigma,B}$ that is defined as follows: For all $s \in S' \setminus S$, $A_{\Sigma,B,s} = A_s$. For all $f \in F' \cup P'$, $f_B^{A_{\Sigma,B,s}} = f^A$.

The σ_B-reduct of $A_{\Sigma,B}$ agrees with A: $A_{\Sigma_B}|_{\sigma_B} = A$.

Let Σ_B be constructive and $\mu\Sigma_B$ be initial in Alg_{Σ_B}.

U_Σ has a left adjoint $L_{\Sigma'} : Alg_\Sigma \to Alg_{\Sigma'}$:

$L_{\Sigma'}(B)$ is the σ_B-reduct of $\mu\Sigma_B$ and called the **free Σ'-algebra over** B.

The unit $\eta : Id \to U_\Sigma L_{\Sigma'}$ is defined as follows: For all $b \in B$, $\eta_B(b) = b$.

The co-unit $\epsilon : L_{\Sigma'} U_\Sigma \to Id$ is defined as follows: For all Σ'-algebras A,

$$L_{\Sigma'}(C) \xrightarrow{\epsilon_A} A \quad = \quad \mu\Sigma_C|_{\sigma_C} \xrightarrow{fold^{A_{\Sigma,C}}|_{\sigma_C}} A_{\Sigma,C}|_{\sigma_C}$$

where $C = U_\Sigma(A)$ and $fold^{A_{\Sigma,C}}$ is the unique Σ_C-homomorphism from $\mu\Sigma_C$ to $A_{\Sigma,C}$.

Let $\Sigma(B)$ be destructive and $\nu\Sigma_B$ be the final Σ_B-algebra.

U_Σ has a right adjoint $R_{\Sigma'} : Alg_\Sigma \longrightarrow Alg_{\Sigma'}$:

$R_{\Sigma'}(B)$ is the σ_B-reduct of $\nu\Sigma_B$ and called the **cofree Σ'-algebra over B**.

The co-unit $\epsilon : U_\Sigma R_{\Sigma'} \longrightarrow Id$ is defined as follows: For all $b \in B$, $\epsilon_B(b) = b$.

The unit $\eta : Id \longrightarrow R_{\Sigma'} U_\Sigma$ is defined as follows: For all Σ'-algebras A,

$$A \xrightarrow{\eta_A} R_{\Sigma'}(C) \qquad = \qquad A_{\Sigma,C}|_{\sigma_C} \stackrel{unfold^{A_{\Sigma,C}}|_{\sigma_C}}{\longrightarrow} \nu\Sigma_C|_{\sigma_C}$$

where $C = U_\Sigma(A)$ and $unfold^{A_{\Sigma,C}}$ is the unique Σ_B-homomorphism from $A_{\Sigma,C}$ to $\nu\Sigma_B$.

From constructors to destructors

Let $\Sigma = (S, BS, F, P)$ be a constructive signature and A be the initial Σ-algebra.

By Lambek's Lemma, the initial H_Σ-algebra

$$\alpha = \{\alpha_s : H_\Sigma(A)_s \stackrel{[f^A]_{f:e\to s\in F}}{\longrightarrow} A_s \mid s \in S\}$$

(see Constructive-signature functors) is an isomorphism in Set^S. Hence there are the H_Σ-coalgebra

$$\{\alpha_s^{-1} : A_s \stackrel{d_s^A}{\longrightarrow} H_\Sigma(A)_s \mid s \in S\},$$

which corresponds to a $co\Sigma$-algebra where

$$co\Sigma \;=\; (S, \{d_s : s \to \coprod_{f:e\to s\in F} e \mid s \in S\}, P)$$

is a destructive signature and for all $f : e \longrightarrow s$, $d_s^A \circ f^A = \iota_f$.

Suppose that Σ is flat (see Constructive-signature functors). Then $co\Sigma$ is also flat – provided that we regard the (finite-product) domains of the function symbols of F as additional sorts and their projections as additional destructors, i.e.,

$$\begin{array}{rcl} co\Sigma & = & (\; S \cup S', \\ & & \{d_s : s \to \coprod_{f:e\to s\in F} e \mid s \in S\} \cup \\ & & \{\pi_i : e \to e_i \mid e = e_1 \times \cdots \times e_n \in S',\ 1 \leq i \leq n\}, \\ & & P \;) \end{array}$$

where $S' = \{e \mid f : e \to s \in F\}$.

Hence the elements of the final $co\Sigma$-algebra can be represented as ground $co\Sigma$-coterms, i.e., (equivalence classes of) finitely branching trees of finite or infinite depth whose edges are labelled with function symbols of $co\Sigma$.

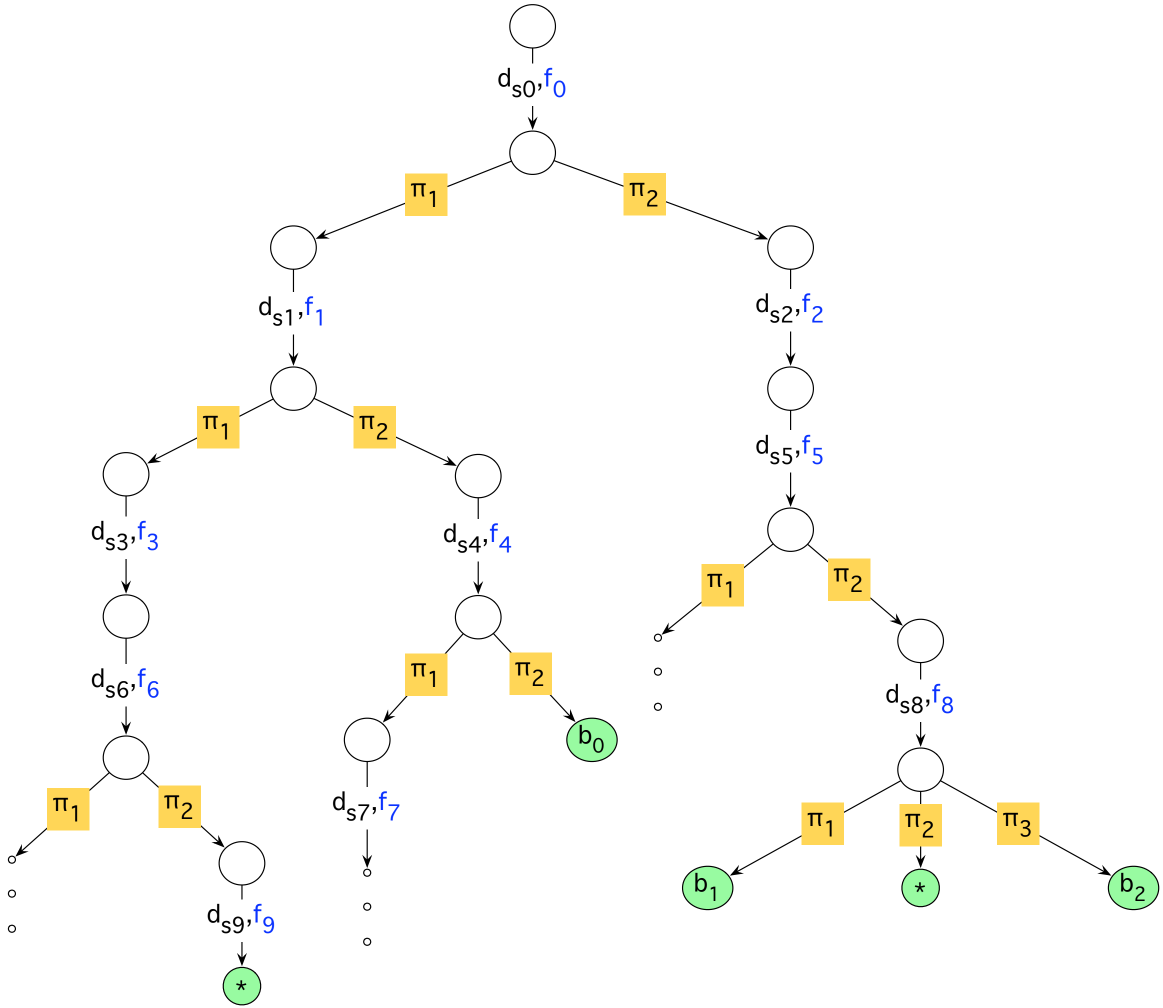


Figure 1. A ground coΣ-coterm t

These trees are in one-to-one correspondence with ground Σ-terms, i.e., trees whose nodes are labelled with function symbols of Σ.

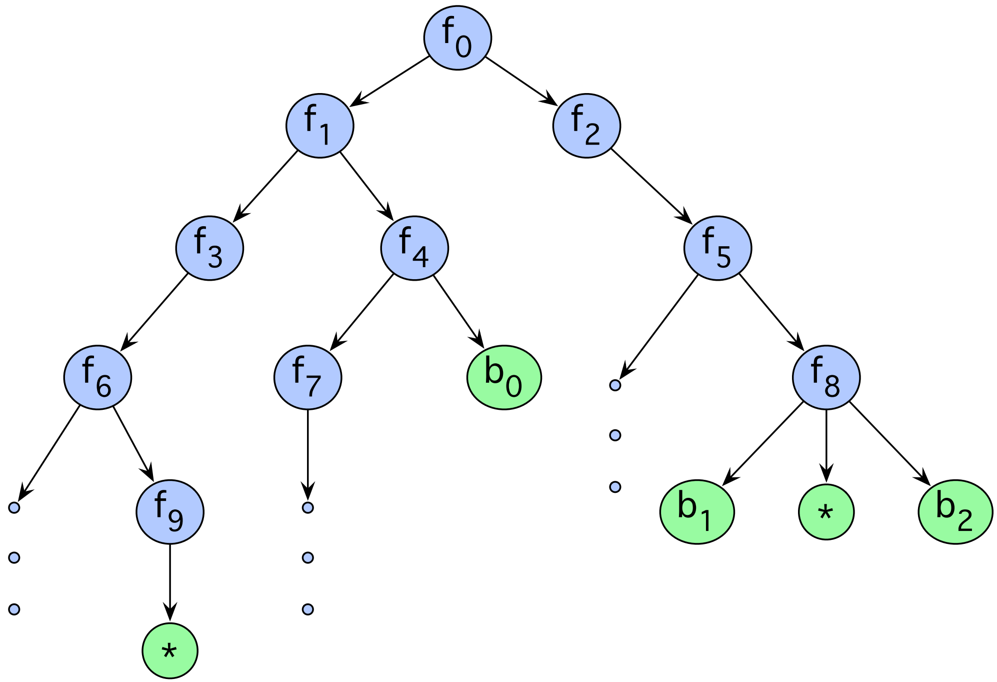


Figure 2. The unique infinite Σ-term obtained from t

Since infinite trees can be formalized as completions of infinite sequences of finite terms, this observation illustrates the following well-known result:

The final $co\Sigma$-algebra is a **completion** of the initial Σ-algebra (see [12], Thm. 3.2; [4], Prop. IV.2). ❑

Instead of presenting infinite terms as infinite sequences of finite terms we define the set of finite or infinite (ground) Σ-terms directly as follows:

Ground Σ-terms

Let T be the greatest $\mathbb{FT}(S,BS)$-sorted set of partial functions

$$t:\mathbb{N}^*\rightarrow F\cup\{word,bag,set\}\cup\bigcup BS$$

such that

- for all $X\in BS$, $T_X=X$,
- for all $s\in S$, if $t\in T_s$, then there is $f:e_1\times\cdots\times e_n\rightarrow s\in F$ such that $t(\epsilon)=f$, for all $0\leq i<n$, $t(wi)\in T_{e_i}$ and for all $i\geq n$, $t(wi)$ is undefined,
- for all collection types $c(s)\in\mathbb{FT}(S,BS)$, if $t\in T_{c(s)}$, then there is $n\in\mathbb{N}$ such that $t(\epsilon)=c$, for all $0\leq i<n$, $t(wi)\in T_s$ and for all $i\geq n$, $t(wi)$ is undefined.

Let $\sim$ be the greatest equivalence relation on T such that for all $e \in \mathbb{FT}(S, BS)$, $t, u \in T_e$ and the lists $t_1, \dots, t_m$ and $u_1, \dots, u_n$ of maximal proper subtrees of t resp. u, $t \sim u$ implies

- $e \in S \cup BS$, $m = n$ and for all $1 \leq i \leq n$, $t_i \sim u_i$, or
- e is a word type, $m = n$ and for all $1 \leq i \leq n$, $t_i \sim u_i$, or
- e is a bag type, $m = n$ and there is a bijection h on $\{1, \dots, n\}$ such that for all $1 \leq i \leq n$, $t_i \sim u_{h(i)}$, or
- e is a set type, for all $1 \leq i \leq m$ there is $1 \leq j \leq n$ with $t_i \sim u_j$ and for all $1 \leq j \leq n$ there is $1 \leq i \leq n$ with $t_i \sim u_j$.

The elements of $CT_\Sigma = T/\!\sim$ are called **ground Σ-terms**.

Of course, finite ground Σ-terms, which represent the elements of the initial Σ-algebra $\mu\Sigma$ (see The initial model of a flat constructive signature), can be embedded into CT_Σ:

Let $h : \mu\Sigma \to CT_\Sigma$ be defined as follows: For all $f : e \to s \in F$ and $(t_1, \dots, t_n) \in \mu\Sigma_e$,

$$h(f(t_1, \dots, t_n)) = f^{CT_\Sigma}(h(t_1), \dots, h(t_n)).$$

h is a Σ-monomorphism. Hence we write $f(t_1, \dots, t_n)$ for $f^{CT_\Sigma}(t_1, \dots, t_n)$.

Suppose that for all $s \in S$, F contains the constructor $\bot_s : 1 \to s$. For all $t \in CT_\Sigma$,

$$def(t) = \{w \in \mathbb{N}^* \mid t(w) \text{ is defined}, t(w) \neq \bot\}.$$

$v \in def(t)$ is the **root position** of the subtree $\lambda w.t(vw)$ of t.

t is **finite** resp. **infinite** if $def(t)$ is finite resp. infinite.

A Σ-algebra A is ω**-continuous** if its carriers are ω-complete posets and if for all $f \in F$, f^A is ω-continuous (see CPOs, lattices and fixpoints).

ωAlg_Σ denotes the subcategory of Alg_Σ that consists of all ω-continuous Σ-algebras as objects and all ω-continuous Σ-homomorphisms between them.

CT_Σ is initial in ωAlg_Σ. ([23], Thm. 4.8)

Proof. A partial order on CT_Σ is defined as follows: For all $s \in S$ and $t, u \in CT_{\Sigma,s}$,

$$t \leq u \;\Leftrightarrow_{def}\; \forall\, w \in def(t) : t(w) \neq \bot \Rightarrow t(w) = u(w).$$

The Σ-tree Ω_s with

$$t(w) =_{def} \begin{cases} \bot_s & \text{if } w = \epsilon, \\ \text{undefined} & \text{otherwise,} \end{cases}$$

is the least element of $CT_{\Sigma,s}$ w.r.t. $\leq$.

Every ω-chain $\{t_i \mid i \in \mathbb{N}\}$ of Σ-trees has a supremum: For all $w \in \mathbb{N}^*$,

$$(\sqcup_{i\in\mathbb{N}} t_i)(w) \; =_{def} \; \begin{cases} t_i(w) & \text{if } \exists\, i \in \mathbb{N} : w \in def(t_i) \wedge t_i(w) \neq \bot, \\ \bot & \text{if } \exists\, i \in \mathbb{N} : w \in def(t_i) \wedge \forall\, k \geq i : t_k(w) = \bot, \\ \text{undefined} & \text{otherwise.} \end{cases}$$

Hence CT_Σ is an ω-CPO.

For all $f : e \to s \in F$, $(t_1, \dots, t_n) \in CT_{\Sigma,e}$ and $w \in \mathbb{N}^*$,

$$f^{CT_\Sigma}(t_1, \dots, t_n)(w) \; =_{def} \begin{cases} f & \text{if } w = \epsilon, \\ t_i(v) & \text{if } w = (i-1)v. \end{cases}$$

f^{CT_Σ} is ω-continuous: Let $e = e_1 \times \cdots \times e_n$. For all $1 \leq i \leq n$, let $\{t_{i,k} \mid k \in \mathbb{N}\}$ be an ω-chain of Σ-trees of type e_i. Then for all $w \in \mathbb{N}^*$,

$$f^{CT_\Sigma}(\sqcup_{k\in\mathbb{N}} t_{1,k}, \ldots, \sqcup_{k\in\mathbb{N}} t_{n,k})(w) = \left\{ \begin{array}{ll} f & \text{if } w = \epsilon \\ (\sqcup_{k\in\mathbb{N}} t_{i,k})(v) & \text{if } w = iv \end{array} \right\}$$
$$= \sqcup_{k\in\mathbb{N}} \left\{ \begin{array}{ll} f & \text{if } w = \epsilon \\ t_{i,k}(v) & \text{if } w = iv \end{array} \right\} = \sqcup_{k\in\mathbb{N}} f^{CT_\Sigma}(t_{1,k}, \ldots, t_{n,k})(w).$$

For the initiality of CT_Σ in ωAlg_Σ, consult [23], Thm. 4.15; [12], Thm. 3.2; or [4], Prop. IV.2. ❑

For all $t \in CT_\Sigma$ and $n \in \mathbb{N}$, $t|_n$ denotes the restriction of t to positions of length less than n: For all $w \in \mathbb{N}^*$,

$$(t|_n)(w) \;=_{def} \begin{cases} t(w) & \text{falls } |w| < n, \\ t(w) & \text{falls } |w| = n \wedge t(w) \in \cup BS, \\ \bot & \text{falls } |w| = n \wedge t(w) \in F \cup \{word, bag, set\}, \\ \text{undefined} & \text{otherwise.} \end{cases}$$

Hence $def(t|_n)$ is finite and $t = \sqcup_{n\in\mathbb{N}} t|_n$.

Completion Theorem

Let A be an ω-CPO and $f : T_\Sigma \rightarrow A$ be monotone. Then

$$\begin{array}{rcl} g : CT_\Sigma & \rightarrow & A \\ t & \mapsto & \begin{cases} f(t) & \text{falls } def(t) \text{ endlich ist} \\ \sqcup_{n\in\mathbb{N}} f(t|_n) & \text{sonst} \end{cases} \end{array}$$

is ω-continuous. g is Σ-homomorphic if A is an ω-continuous Σ-algebra and f is Σ-homomorphic.

Proof. See the proof of [23], Thm. 4.8. ❏

For all ω-continuous Σ-algebras A, $fold_\omega^A$ denotes the unique ω-continuous Σ-homomorphism from CT_Σ to A. For all $t \in CT_\Sigma$,

$$fold_\omega^A(t) = \sqcup_{n\in\mathbb{N}} fold^A(t|_n).$$

Hence by the Completion Theorem, $fold_\omega^A$ is ω-continuous.

CT_Σ is a $co\Sigma$-algebra: For all $s \in S$ and $t = f(t_1, \ldots, t_n) \in CT_{\Sigma,s}$,

$$d_s^{CT_\Sigma}(t) \;=_{def}\; ((t_1, \ldots, t_n), f).$$

CT_Σ is final in $Alg_{co\Sigma}$.

Proof.

Let A be a $co\Sigma$-algebra. An S-sorted function $unfold^A : A \to CT_\Sigma$ is defined as follows: For all $s \in S$, $a \in A_s$, $i \in \mathbb{N}$ and $w \in \mathbb{N}^*$, $d_s^A(a) = ((a_1, \ldots, a_n), f)$ implies

$$\begin{aligned} unfold^A(a)(\epsilon) &= f, \\ unfold^A(a)(iw) &= \begin{cases} unfold^A(a_{i+1})(w) & \text{if } 0 \leq i < n, \\ \text{undefined} & \text{otherwise.} \end{cases} \end{aligned}$$

$unfold^A(a)$ is represented by the tree whose root is labelled with f and whose subterms are given by $unfold^A(a_1), \ldots, unfold^A(a_n)$.

$unfold^A$ is $co\Sigma$-homomorphic: Let $s \in S$, $a \in A_s$ and $d_s^A(a) = ((a_1, \ldots, a_n), f)$. Then

$$d_s^{CT_\Sigma}(unfold^A(a)) = d_s^{CT_\Sigma}(f(unfold^A(a_1), \ldots, unfold^A(a_n)))$$
$$= ((unfold^A(a_1), \ldots, unfold^A(a_n)), f) = unfold^A((a_1, \ldots, a_n), f) = unfold^A(d_s^A(a)).$$

Let $h : A \rightarrow CT_\Sigma$ be a $co\Sigma$-homomorphism. Then

$$d_s^{CT_\Sigma}(h(a)) = h(d_s^A(a)) = h((a_1,\dots,a_n),f) = ((h(a_1),\dots,h(a_n)),f)$$
$$= d_s^{CT_\Sigma}(f(h(a_1),\dots,h(a_n)))$$

and thus $h(a) = f(h(a_1),\dots,h(a_n))$ because $d_s^{CT_\Sigma}$ is injective. We conclude that h agrees with $unfold^A$. ❑

Hence there is a $co\Sigma$-isomorphism

$$h : CT_\Sigma \xrightarrow{\sim} \nu co\Sigma$$

(see The final model of a flat destructive signature). h decomposes Σ-terms:

For all $s \in S$ and $t = f(t_1,\dots,t_n) \in CT_{\Sigma,s}$,

$$h(t)(d_s) = d_s^{\nu co\Sigma}(h(t)) = h(d_s^{CT_\Sigma}(t)) = h((t_1,\dots,t_n),f) = (h(t_1),\dots,h(t_n),f).$$

For instance, the $co\Sigma$-coterm shown in Figure 1 is the h-image of the Σ-term shown in Figure 2.

From destructors to constructors

Let $\Sigma = (S, BS, F, P)$ be a destructive signature and A be the final Σ-algebra.

By Lambek's Lemma, the final H_Σ-coalgebra

$$\alpha = \{\alpha_s : A_s \stackrel{\langle f^A \rangle_{f:e\to s\in F}}{\longrightarrow} H_\Sigma(A)_s \mid s \in S\}$$

(see Destructive-signature functors) is an isomorphism in Set^S. Hence there are the H_Σ-algebra

$$\{\alpha_s^{-1} : H_\Sigma(A)_s \stackrel{c_s^A}{\longrightarrow} A_s \mid s \in S\},$$

which corresponds to a $co\Sigma$-algebra where

$$co\Sigma \;=\; (S, \{c_s : \prod_{f:s\to e\in F} e \to s \mid s \in S\}, P)$$

is a constructive signature and for all $f : s \to e$, $f^A \circ c_s^A = \pi_f$.

Suppose that Σ is flat (see Destructive-signature functors). Then $co\Sigma$ is also flat – provided that we regard the (finite-coproduct) ranges of the function symbols of F as additional sorts and their injections as additional constructors, i.e.,

$$\begin{array}{rcl} co\Sigma & = & (\ S \cup S', \\ & & \quad \{c_s : \prod_{f:s\to e\in F} e \to s \mid s \in S\} \cup \\ & & \quad \{\iota_i : e_i \to e \to \mid e = e_1 + \cdots + e_n \in S',\ 1 \leq i \leq n\}, \\ & & \quad P\ ) \end{array}$$

where $S' = \{e \mid f : s \to e \in F\}$.

Hence the elements of the initial $co\Sigma$-algebra can be represented as finite ground $co\Sigma$-terms, i.e., (equivalence classes of) finitely branching trees of finite depth whose nodes are labelled with function symbols of $co\Sigma$.

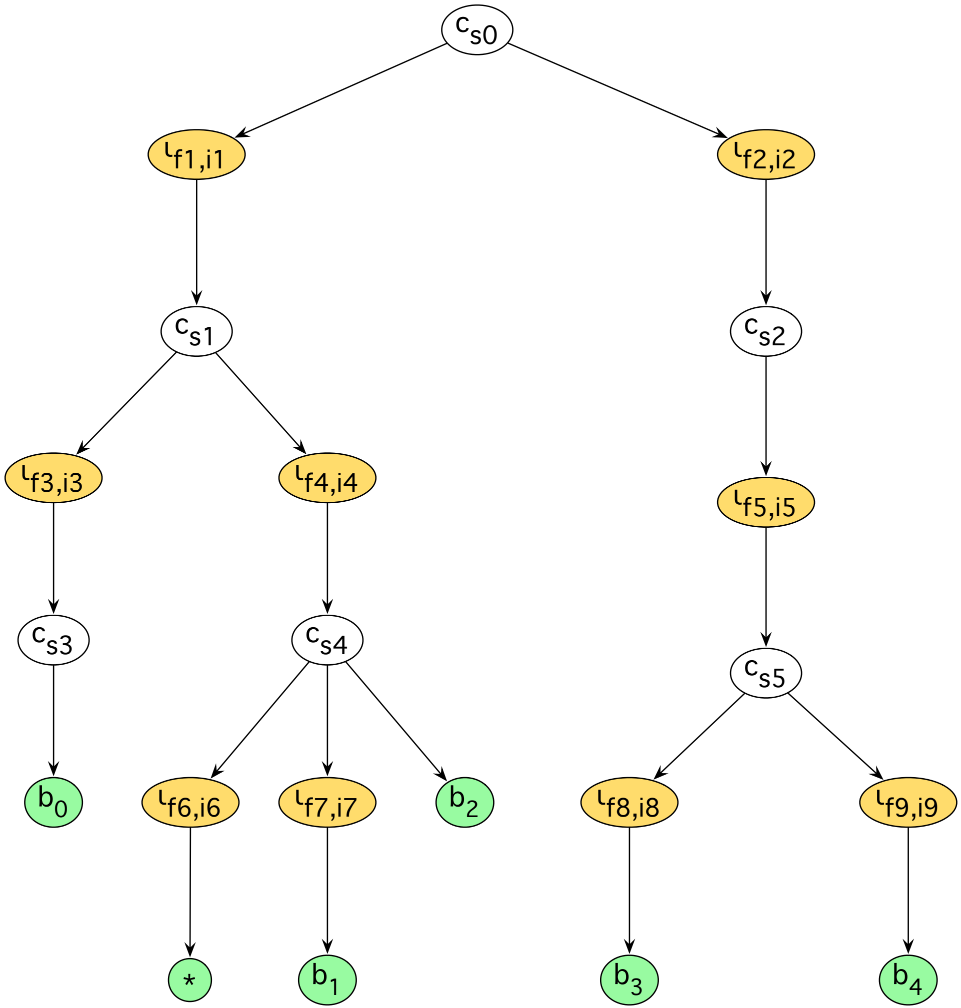


Figure 3. A finite ground coΣ-term t

These trees are in one-to-one correspondence with finite ground Σ-coterms, i.e., finite trees whose nodes are labelled with function symbols of Σ.

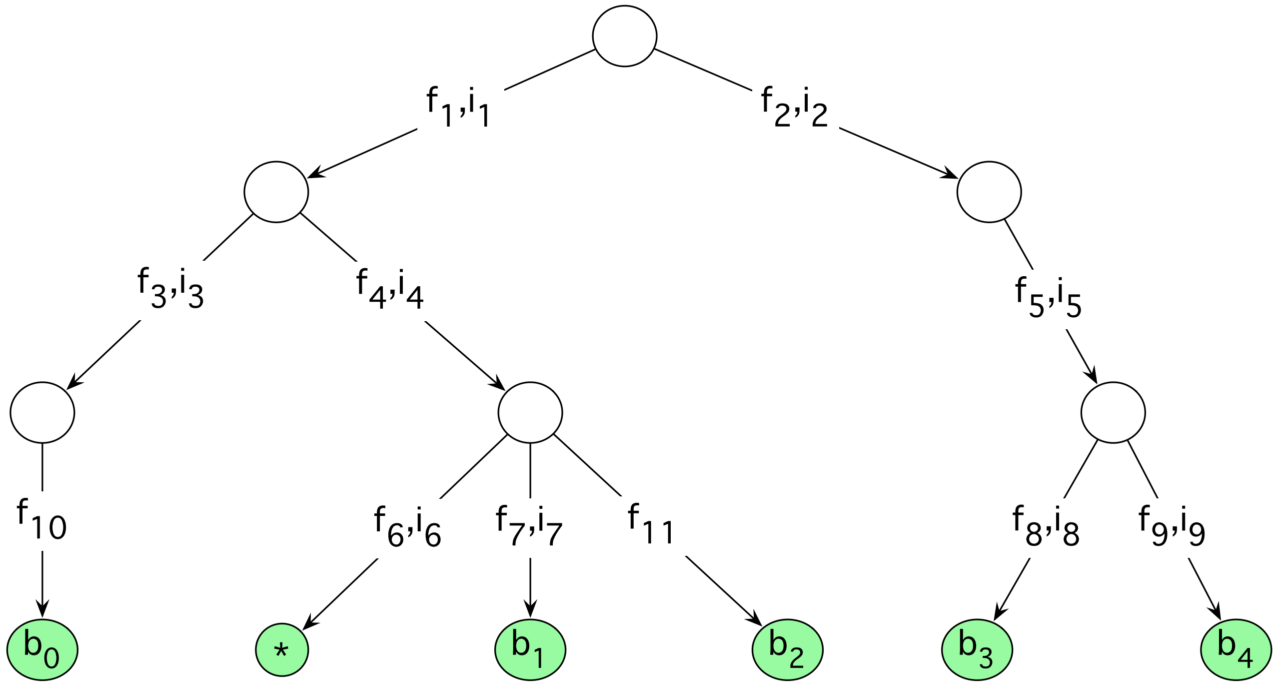


Figure 4. The unique Σ-coterm obtained from t

$\mu co\Sigma$ is a Σ-algebra: For all $s \in S$, $t = c_s(\iota_{f_1,i_1}(t_1),\dots,\iota_{f_n,i_n}(t_n)) \in \mu co\Sigma_s$ and $1 \leq k \leq n$,

$$f_k^{\mu co\Sigma}(t) =_{def} \iota_{f_k,i_k}(t_k).$$

Since $\nu\Sigma$ is final in Alg_Σ, there is a $co\Sigma$-homomorphism

$$h : \mu co\Sigma \rightarrow \nu\Sigma.$$

h decomposes Σ-coterms:

For all $s \in S$, $t = c_s(\iota_{f_1,i_1}(t_1),\dots,\iota_{f_n,i_n}(t_n)) \in \mu co\Sigma_s$ and $1 \leq k \leq n$,

$$h(t)(f_k) = f_k^{\nu\Sigma}(h(t)) = h(f_k^{\mu co\Sigma}(t)) = h(\iota_{f_k,i_k}(t_k)) = \iota_{f_k,i_k}(h(t_k)).$$

$img(h)$ consists of all *finite* ground Σ-coterms.

For instance, the Σ-coterm shown in Figure 4 is the h-image of the $co\Sigma$-term shown in Figure 3.

Recursive Σ-equations

Let $\Sigma = (S, BS, F, P)$ be a flat constructive signature and V be an S-sorted set of variables. An S-sorted function

$$E : V \to T_\Sigma(V)$$

is called a system of **recursive Σ-equations** (see Constructive-signature functors).

E is **ideal** if for all $x \in V$ $E(x) \notin V$.

Let A be a Σ-algebra. E induces the **step function**

$$\begin{aligned} E_A : A^V &\to A^V \\ g &\mapsto g^* \circ E \end{aligned}$$

(see Term adjunction). A **solution of E in A** is a fixpoint of E_A.

By Lemma EVAL, for all $g \in A^V$ and Σ-homomorphisms $h : A \to B$,

$$h \circ E_A(g) = E_B(h \circ g). \tag{1}$$

Let A be ω-continuous. Then the partial orders, least elements and suprema of A are lifted to A^V as usually, i.e., A^V is ω-CPO. By [23], Prop. 4.13, E_A is ω-continuous.

Hence by Kleene's Fixpoint Theorem (1),

$$lfp(E_A) \;=_{def}\; \sqcup_{n\in\mathbb{N}} E_A^n(\lambda x.\bot^A) \tag{2}$$

is the least solution of E in A.

For all ω-continuous Σ-homomorphisms $h : A \to B$,

$$h \circ lfp(E_A) \;=\; lfp(E_B). \tag{3}$$

Proof. By (1) and since $h(\bot^A) = \bot^B$, one obtains

$$h \circ E_A^n(\lambda x.\bot^A) \;=\; E_B^n(\lambda x.\bot^B)$$

for all $n \in \mathbb{N}$ by induction on n. Hence (3) holds true because h is ω-continuous. ❏

Solution Theorem

Every ideal system $E : V \to T_\Sigma(V)$ of recursive Σ-equations has a unique solution in CT_Σ.

Proof. Let $g : V \to CT_\Sigma$ be a solution of E in CT_Σ. Then

$$lfp(E_{CT_\Sigma}) \leq g. \tag{4}$$

Let $n \in \mathbb{N}$. By induction on n, it can be shown that for all $x \in V$ and $w \in \mathbb{N}^n$,

$$w \in def(g(x)) \;\; \text{implies} \;\; w \in def(E^{n+1}_{CT_\Sigma}(\Omega)(x))$$

(see [50], Satz 17). Hence by (2),

$$w \in def(g(x)) \;\; \text{implies} \;\; w \in def(\sqcup_{n\in\mathbb{N}} E^n_{CT_\Sigma}(\Omega)(x)) = def(lfp(E_{CT_\Sigma})(x))$$

and thus by (4), $def(g(x)) = def(lfp(E_{CT_\Sigma})(x))$. Consequently, (4) implies $g = lfp(E_{CT_\Sigma})$. ❏

$T_\Sigma(V)$ is a $co\Sigma$-algebra:

- For all $x \in V$, $E(x) = f(t_1, \ldots, t_n)$ implies $d_s^{T_\Sigma(V)}(x) = ((t_1, \ldots, t_n), f)$.
- For all $f : e \to s \in F$ and $(t_1, \ldots, t_n) \in T_\Sigma(V)_e$,
  $d_s^{T_\Sigma(X)}(f(t_1, \ldots, t_n)) = ((t_1, \ldots, t_n), f)$.

The Solution Theorem can also be concluded from the facts that CT_Σ and $T_\Sigma(V)$ are $co\Sigma$-algebras and CT_Σ is the final one:

Lemma COSOL

Let $h : V \to CT_\Sigma$ be an S-sorted function. $h^* : T_\Sigma(V) \to CT_\Sigma$ is $co\Sigma$-homomorphic iff h is a solution of E in CT_Σ.

Proof.

"$\Rightarrow$": Let h^* be $co\Sigma$-homomorphic, $s \in S$, $x \in V_s$ and $E(x) = f(t_1, \dots, t_n)$. Then

$$\begin{aligned} & d_s^{CT_\Sigma}(E_{CT_\Sigma}(h)(x)) = d_s^{CT_\Sigma}(h^*(E(x))) = d_s^{CT_\Sigma}(h^*(f(t_1, \dots, t_n))) \\ & = d_s^{CT_\Sigma}(f(h^*(t_1), \dots, h^*(t_n))) = ((h^*(t_1), \dots, h^*(t_n)), f) = h^*((t_1, \dots, t_n), f) \\ & = h^*(d_s^{T_\Sigma(V)}(x)) = d_s^{CT_\Sigma}(h(x)). \end{aligned} \tag{5}$$

Hence $\{(E_{CT_\Sigma}(h)(x), h(x))\} \cup \Delta_{CT_\Sigma}$ is a $co\Sigma$-congruence and thus we conclude $E_{CT_\Sigma}(h)(x) = h(x)$ by algebraic coinduction because CT_Σ is final in $Alg_{co\Sigma}$. Hence h is a solution of E in CT_Σ.

"$\Leftarrow$": Let h be a solution of E in CT_Σ. Then for all $x \in V$, $E_{CT_\Sigma}(h)(x) = h(x)$ and thus $d_s^{CT_\Sigma}(E_{CT_\Sigma}(h)(x)) = d_s^{CT_\Sigma}(h(x))$. Hence by re-arranging the equations of (5), one obtains

$$h^*(d_s^{T_\Sigma(V)}(x)) = d_s^{CT_\Sigma}(h(x)). \tag{6}$$

Moreover, for all $f : e \to s \in F$ and $(t_1, \ldots, t_n) \in T_{\Sigma,e}$,

$$\begin{aligned} h^*(d_s^{T_\Sigma(V)}(f(t_1,\ldots,t_n))) &= h^*((t_1,\ldots,t_n),f) = ((h^*(t_1),\ldots,h^*(t_n)),f) \\ &= d_s^{CT_\Sigma}(f(h^*(t_1),\ldots,h^*(t_n))) = d_s^{CT_\Sigma}(h^*(f(t_1,\ldots,t_n))) \end{aligned} \tag{7}$$

By (6) and (7), h^* is $co\Sigma$-homomorphic. ❏

Since there is exactly one $co\Sigma$-homomorphism from $T_\Sigma(V)$ to CT_Σ, Lemma COSOL implies that there is exactly one solution of E in CT_Σ: If there were two solutions $g, h : V \to CT_\Sigma$, then $g^* = h^*$ and thus $g = g^* \circ inc_V = h^* \circ inc_V = h$. We conclude that the Solution Theorem holds true.

Let $\Sigma = (S, BS, F, P)$ be a constructive signature, $\mu\Sigma$ be initial in Alg_Σ,

$\mathcal{K} = \prod_{s \in S} \mathcal{K}_s$ be a product category

and $(L : Set^S \to \mathcal{K}, R : \mathcal{K} \to Set^S, \eta, \epsilon)$ be an adjunction.

A $\mathcal{K}$-morphism $f : L(\mu\Sigma) \to A$ is Σ**-recursive** if the kernel of $f^\# : \mu\Sigma \to R(A)$ is compatible with F.

Lemma REC

$f : L(\mu\Sigma) \to A$ is Σ-recursive iff $R(A)$ is a Σ-algebra and $g^\# : \mu\Sigma \to R(A)$ coincides with $fold^{R(A)}$.

Proof. Lemma KER (1). ❑

Let $\Sigma = (S, BS, F, P)$ be a destructive signature, $\nu\Sigma$ be final in Alg_Σ,

$\mathcal{K} = \prod_{s\in S} \mathcal{K}_s$ be a product category

and $(L : \mathcal{K} \to Set^S, R : Set^S \to \mathcal{K}, \eta, \epsilon)$ be an adjunction.

A $\mathcal{K}$-morphism $f : A \to R(\nu\Sigma)$ is **Σ-corecursive** if the image of $f^* : L(A) \to \nu\Sigma$ is compatible with F.

Lemma COR

$f : A \to R(\nu\Sigma)$ is Σ-corecursive iff $L(A)$ is a Σ-algebra and $f^* : L(A) \to \nu\Sigma$ coincides with $unfold^{L(A)}$.

Proof. Lemma IMG (1). ❑

Conservative extensions

Let $\Sigma = (S, F, P)$ be a signature, $\Sigma' = (S', F', P')$ be a subsignature of Σ, AX be a set Σ-formulas, $AX' \subseteq AX$ be a set Σ'-formulas, A be a Σ-algebra and $B = A|_{\Sigma'}$.

$Alg^{=}_{\Sigma,AX}$ denotes the full subcategory $\mathcal{K}$ of $Alg_{\Sigma,AX}$ such that for all equality predicates $=: e \times e$ of P and $A \in \mathcal{K}$, $=^A = \Delta_A$. The objects of $Alg^{=}_{\Sigma,AX}$ are called **Σ, AX-algebras with equality**.

$Alg^{\in}_{\Sigma,AX}$ denotes the full subcategory $\mathcal{K}$ of $Alg_{\Sigma,AX}$ such that for all membership predicates $\in: e$ of P and $A \in \mathcal{K}$, $\in^A = A$. The objects of $Alg^{\in}_{\Sigma,AX}$ are called **Σ, AX-algebras with membership**.

Constructor extensions

Let Σ be constructive and $\mu\Sigma$ and $\mu\Sigma'$ be initial in $Alg^{=}_{\Sigma,AX}$ resp. $Alg^{=}_{\Sigma',AX'}$.

A is **F'-reachable** (or **F'-generated**) if $fold^B : \mu\Sigma' \to B$ is surjective.
A is **F'-consistent** if $fold^B$ is injective.

(Σ, AX) is a **conservative extension of** (Σ', AX') if $\mu\Sigma$ is F'-reachable and F'-consistent, i.e. if $\mu\Sigma|_{\Sigma'}$ and $\mu\Sigma'$ are isomorphic.

Intuitively,

A is F'-reachable if each element of A is obtained by folding an element of $\mu\Sigma'$;

A is F'-consistent if for each element a of A there is only one element of $\mu\Sigma'$ that folds into a.

A is F'-reachable iff $img(fold^B) = B$. (1)

A is F'-consistent iff $ker(fold^B) = \Delta_{\mu\Sigma'}$.

Given a category $\mathcal{K}$ of Σ-algebras, the full subcategory of F-reachable objects of $\mathcal{K}$ is denoted by $gen(\mathcal{K})$.

Lemma REACH

Let A be initial in $Alg_{\Sigma,AX}$.

A is F'-reachable iff $img(fold^B)$ is a Σ-invariant.

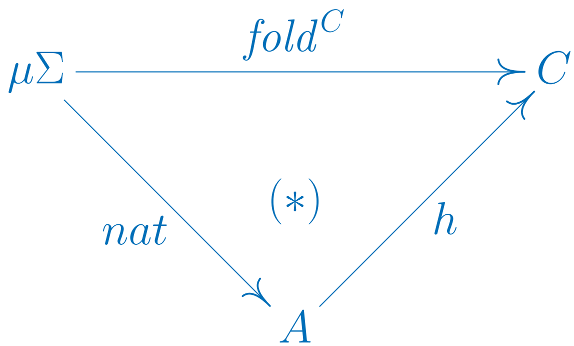
Proof. "$\Rightarrow$": Let A be F'-reachable. Then $img(fold^B) = B = A$ and thus $img(fold^B)$ is a Σ-invariant.

"$\Leftarrow$": Let $img(fold^B)$ be a Σ-invariant. By Lemma MAX (1), A is the least Σ-invariant of A. Hence $B = A \subseteq img(fold^B) \subseteq B$ and thus by (1), A is F'-reachable. ❏

Lemma CONEXT

Let $\mu\Sigma'$ be extendable to a (Σ, AX)-algebra C with equality. Then (Σ, AX) is a conservative extension of (Σ', AX').

Proof. Let $fold^C$ be the unique Σ-homomorphism from $\mu\Sigma$ to C, $A = \mu\Sigma/ker(fold^C)$ and $B = A|_{\Sigma'}$. By Lemma KER (2), there is a unique Σ-monomorphism $h : A \longrightarrow C$ such that $(*)$ commutes:



By Lemma NAT (4), A satisfies AX. Hence $A \in Alg^{=}_{\Sigma,AX}$ and thus $B \in Alg^{=}_{\Sigma',AX'}$. Let $fold^B$ be the unique Σ'-homomorphism from $\mu\Sigma'$ to B.

$$\mu\Sigma' \stackrel{fold^B}{\longrightarrow} B \stackrel{h|_{\Sigma'}}{\longrightarrow} C|_{\Sigma'} = \mu\Sigma'$$

agrees with the identity on $\mu\Sigma'$ because $\mu\Sigma'$ is initial. Since $id_{\mu\Sigma'}$ is epi, Lemma EPIMON implies that $h|_{\Sigma'}$ is also epi. We conclude that $\mu\Sigma'$ and B are Σ'-isomorphic and thus (Σ, AX) is a conservative extension of (Σ', AX'). ❑

Destructor extensions

Let Σ be destructive and $\nu\Sigma$ and $\nu\Sigma'$ be final in $Alg^{\in}_{\Sigma,AX}$ resp. $Alg^{\in}_{\Sigma',AX'}$.

A is F'**-observable** (or F'**-cogenerated**) if $unfold^B : B \to \nu\Sigma'$ is injective.
A is F'**-complete** if $unfold^B$ is surjective.

(Σ, AX) is a **conservative extension** of (Σ', AX') and $F \setminus F'$ is **derived from** F if $\nu\Sigma$ is F'-observable and F'-complete, i.e. $\nu\Sigma|_{\Sigma'}$ and $\nu\Sigma'$ are isomorphic.

Intuitively,

A is F'-observable if for each element a of A, all unfoldings of a in $\nu\Sigma'$ are the same;

A is F'-complete if each element of $\nu\Sigma'$ is the unfolding of an element of A.

A is F'-observable iff $ker(unfold^B) = \Delta_B$. (3)

A is F'-complete iff $img(unfold^B) = \nu\Sigma'$.

Given a category $\mathcal{K}$ of Σ-algebras, the full subcategory of F-observable objects of $\mathcal{K}$ is denoted by $obs(\mathcal{K})$.

Lemma OBS

Let A be final in $Alg_{\Sigma,AX}$.

A is F'-observable iff $ker(unfold^B)$ is a Σ-congruence.

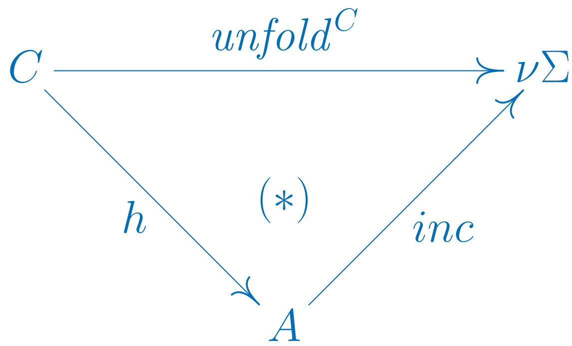
Proof. "$\Rightarrow$": Let A be F'-observable. Then $ker(unfold^B) = \Delta_B = \Delta_A$ and thus $ker(unfold^B)$ is a Σ-congruence.

"⇐": Let $ker(unfold^B)$ be a Σ-congruence. By Lemma MIN (1), Δ_A is the greatest Σ-congruence on A. Hence $\Delta_B \subseteq ker(unfold^B) \subseteq \Delta_A = \Delta_B$ and thus by (3), A is F'-observable. ❑

Lemma DESEXT

Let $\nu\Sigma'$ be extendable to a (Σ, AX)-algebra C with membership. Then (Σ, AX) is a conservative extension of (Σ', AX').

Proof. Let $unfold^C$ be the unique Σ-homomorphism from C to $\nu\Sigma$, $A = img(unfold^C)$ and $B = A|_{\Sigma'}$. By Lemma IMG (2), there is a unique Σ-epimorphism $h : C \to A$ such that $(*)$ commutes:



By Lemma INC (3), A satisfies AX. Hence $A \in Alg^{\in}_{\Sigma,AX}$ and thus $B \in Alg^{\in}_{\Sigma',AX'}$. Let $unfold^B$ be the unique Σ'-homomorphism from B to $\mu\Sigma'$.

$$\nu\Sigma' = C|_{\Sigma'} \stackrel{h|_{\Sigma'}}{\rightarrow} B \stackrel{unfold^B}{\longrightarrow} \nu\Sigma'$$

agrees with the identity on $\nu\Sigma'$ because $\nu\Sigma'$ is final. Since $id_{\nu\Sigma'}$ is mono, Lemma EPIMON implies that $h|_{\Sigma'}$ is also mono. We conclude that $\nu\Sigma'$ and B are Σ'-isomorphic and thus (Σ, AX) is a conservative extension of (Σ', AX'). ❑

Let $\Sigma = (S, F, P)$ be a constructive signature, $\Sigma' = (S, F, \emptyset)$ and $\mu\Sigma'$ be initial in $Alg_{\Sigma'}$.

Lemma REFL

Let $h : A \to B$ be a Σ-homomorphism that preserves all $p : e \in P$, i.e.,

$$p^A \;=\; \{a \in A_e \mid h(a) \in p^B\},$$

$e = \prod_{x \in V} e_x \in \mathbb{T}(S, BS)$ and $\varphi : e \in Fo_\Sigma$ be a negation-free Σ-formula.

If φ does not contain universal quantifiers, then

$$h(\varphi^A) \subseteq \varphi^B. \tag{1}$$

If h is epi, then

$$h^{-1}(\varphi^B) \subseteq \varphi^A. \tag{2}$$

Proof of (1) by induction on the size of φ.

Let $p : e \in P$, $x \in V_s$. W.l.o.g. we assume that r is unary.

$$f \in r(t)^A \;\Leftrightarrow\; t^A(f) \in r^A \;\Leftrightarrow\; t^B(h \circ f) \overset{\textit{Lemma EVAL}}{=} h(t^A(f)) \in r^B \;\Leftrightarrow\; h \circ f \in r(t)^B.$$

$$f \in (\varphi \wedge \psi)^A = \varphi^A \cap \psi^A \overset{i.h.}{\Rightarrow} h \circ f \in \varphi^B \cap \psi^B = (\varphi \wedge \psi)^B.$$

$$f \in (\varphi \vee \psi)^A = \varphi^A \cup \psi^A \overset{i.h.}{\Rightarrow} h \circ f \in \varphi^B \cup \psi^B = (\varphi \vee \psi)^B.$$

$$\begin{aligned} f \in (\exists x \varphi)^A \;&\Leftrightarrow\; \exists\, a \in A_s : \mathit{upd}(f, x, a) \in \varphi^A \\ &\overset{i.h.}{\Rightarrow}\; \exists\, a \in A_s : \mathit{upd}(h \circ f, x, h(a)) = h \circ \mathit{upd}(f, x, a) \in \varphi^B \\ &\Rightarrow\; \exists\, b \in B_s : \mathit{upd}(h \circ f, x, b) \in \varphi^B \;\Leftrightarrow\; h \circ f \in (\exists x \varphi)^B. \end{aligned}$$

Proof of (2) by induction on the size of φ.

Let $r \in R$, $s \in S$ and $x \in V_s$. W.l.o.g. we assume that r is unary.

$$h \circ f \in r(t)^B \;\Leftrightarrow\; h(t^A(f)) \overset{Lemma\ EVAL}{=} t^B(h \circ f) \in r^B \;\Leftrightarrow\; t^A(f) \in r^A \;\Leftrightarrow\; f \in r(t)^A.$$

$$h \circ f \in (\varphi \wedge \psi)^B = \varphi^B \cap \psi^B \overset{i.h.}{\Rightarrow} f \in \varphi^A \cap \psi^A = (\varphi \wedge \psi)^A.$$

$$h \circ f \in (\varphi \vee \psi)^B = \varphi^B \cup \psi^B \overset{i.h.}{\Rightarrow} f \in \varphi^A \cup \psi^A = (\varphi \vee \psi)^A.$$

$$\begin{aligned} h \circ f \in (\exists x \varphi)^B \;&\Leftrightarrow\; \exists\, b \in B_s : upd(h \circ f, x, b) \in \varphi^B \\ &\overset{h\ epi}{\Rightarrow} \exists\, a \in A_s : h \circ upd(f, x, a) = upd(h \circ f, x, h(a)) \in \varphi^B \\ &\overset{i.h.}{\Rightarrow} \exists\, a \in A_s : upd(f, x, a) \in \varphi^A \;\Leftrightarrow\; f \in (\exists x \varphi)^A. \end{aligned}$$

$$\begin{aligned} &h \circ f \in (\forall x \varphi)^B \;\Leftrightarrow\; \forall\, b \in B_s : upd(h \circ f, x, b) \in \varphi^B \\ &\Rightarrow \forall\, a \in A_s : h \circ upd(f, x, a) = upd(h \circ f, x, h(a)) \in \varphi^B \\ &\qquad\overset{i.h.}{\Rightarrow} \forall\, a \in A_s : upd(f, x, a) \in \varphi^A \;\Leftrightarrow\; f \in (\forall x \varphi)^A. \end{aligned}$$

❑

Abstraction with a least congruence

Let AX consist of $\forall$-free Horn clauses such that for all $A \in Alg_{\Sigma,AX}$, $=^A$ is a Σ-congruence, and $C = lfp(\mu\Sigma', \Sigma, AX)$.

Then $\sim \;\; = \;\; =^C$ is the least Σ-congruence on $\mu\Sigma'$.

Let $\mathcal{K} = Alg^{=}_{\Sigma,AX}$. By Lemma NAT, $C/{\sim} \in \mathcal{K}$.

Let $A \in \mathcal{K}$. We define $B \in Alg_\Sigma$ as the $fold^A$-pre-image of the interpretation of R in A, i.e., for all $r : w \in R$,

$$r^B \;\; =_{def} \;\; \{b \in \mu\Sigma'_w \mid fold^A(b) \in r^A\}.$$

Use induction on $\mathbb{N}$ and Kleene's Fixpoint Theorem (or transfinite induction and Zermelo's Fixpoint Theorem ????) to show that $fold^A$ extends to a Σ-homomorphism!

B satisfies AX and thus $B \in Alg_{\Sigma,AX}$.

Proof. Let $\varphi = (r(t_1,\dots,t_n) \Leftarrow \psi) \in AX$ and $g \in \psi^B$. By Lemma REFL (1), $fold^A \circ g \in \psi^A$. Since A satisfies φ, $fold^A \circ g \in r(t_1,\dots,t_n)^A$, i.e.,

$$(fold^A(t_1^B(g)),\dots,fold^A(t_n^B(g))) \stackrel{Lemma\ EVAL}{=} (t_1^A(fold^A \circ g),\dots,t_n^A(fold^A \circ g)) \in r^A.$$

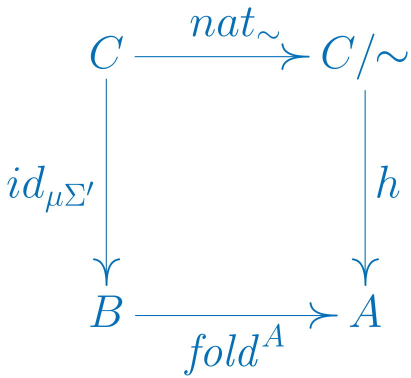
Hence $(t_1^B(g),\dots,t_n^B(g)) \in r^B$ and thus $g \in r(t_1,\dots,t_n)^B$. ❏

Theorem ABSINI $C/\!\sim$ is initial in $\mathcal{K}$.

Proof. Since C is the least $D \in Alg_{\Sigma,AX}$ with $D|_{\Sigma'} = \mu\Sigma'$, we obtain $C \leq B$. In particular,

$$\begin{aligned} \sim \;=\; =^C \;\subseteq\; =^B \;&=\; \{(t,u) \in (\mu\Sigma')^2 \mid fold^A(t) =^A fold^A(u)\} \\ &=\; ker(fold^A) \end{aligned}$$

because $=^A = \Delta_A$. Hence $h : C/\!\sim \longrightarrow A$ is well-defined by $h \circ nat_\sim = fold^A \circ id_{\mu\Sigma'}$.



Since $nat_\sim$ is epi and predicate preserving and $fold^A \circ id_{\mu\Sigma'}$ is Σ-homomorphic, Lemma EMH (1) implies that h is also Σ-homomorphic.

Let h' be any Σ-homomorphism from $C/\!\sim$ to A. Since $B|_{B\Sigma} = BA$ is initial in Alg_Σ, $h' \circ nat_\sim = h \circ nat_\sim$ and thus $h' = h$ because $nat_\sim$ is epi. ❏

Abstraction with a greatest congruence

Let AX consist of co-Horn clauses such that for all $A \in Alg_{\Sigma,AX}$, $=^A$ is a Σ-congruence, $C = gfp(\mu\Sigma', \Sigma, AX)$ and $\sim \;=\; =^C$ be a Σ-congruence on $\mu\Sigma'$. Hence $C \in gen(Alg_{\Sigma,AX})$.

Let $\mathcal{K} = gen(Alg^{=}_{\Sigma,AX})$. By Lemma NAT, $C/\!\sim\, \in \mathcal{K}$.

Let $A \in \mathcal{K}$. We define $B \in Alg_\Sigma$ as the $fold^A$-pre-image of the interpretation of R in A, i.e., for all $r : w \in R$,

$$r^B \;=_{def}\; \{b \in \mu\Sigma'_w \mid fold^A(b) \in r^A\}.$$

Use induction on $\mathbb{N}$ and Kleene's Fixpoint Theorem (or transfinite induction and Zermelo's Fixpoint Theorem ????) to show that $fold^A$ extends to a Σ-homomorphism!

B satisfies AX and thus $B \in gen(Alg_{\Sigma,AX})$.

Proof. Let $r \in R$, $\varphi = (r(t_1,\dots,t_n) \Rightarrow \psi) \in AX$ and $g \in r(t_1,\dots,t_n)^B$. Hence $(t_1^B(g),\dots,t_n^B(g)) \in r^B$ and thus

$$(t_1^A(fold^A \circ g),\dots,t_n^A(fold^A \circ g)) \stackrel{Lemma\ EVAL}{=} (fold^A(t_1^B(g)),\dots,fold^A(t_n^B(g))) \in r^A.$$

Hence $fold^A \circ g \in r(t_1,\dots,t_n)^A$. Since A satisfies φ, $fold^A \circ g \in \psi^A$. Since A is Σ-reachable, $fold^A$ is epi and thus Lemma REFL (2) implies $g \in \psi^B$. ❑

Theorem ABSFIN $C/\!\sim$ is final in $gen(Alg^{=}_{\Sigma,AX})$.

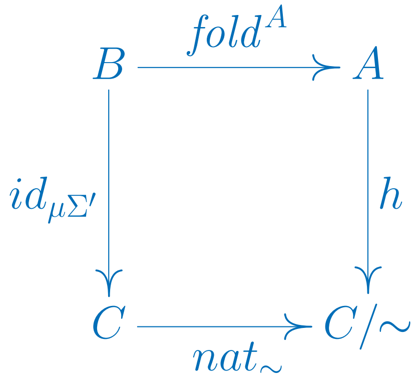
Proof. Since C is the greatest $D \in Alg_{\Sigma,AX}$ with $D|_{B\Sigma} = \mu\Sigma'$, we obtain $B \leq C$. In particular,

$$ker(fold^A) \;=\; \{(t,u) \in (\mu\Sigma')^2 \mid fold^A(t) =^A fold^A(u)\} \;=\; =^B \;\subseteq\; =^C \;=\; \sim$$

because $=^A \,=\, \Delta_A$.

Hence for all $t,u \in \mu\Sigma'$, $fold^A(t) = fold^A(u)$ implies $t \sim u$. Since A is Σ-reachable, $fold^A$ is epi and thus for all $a \in A$ there is $t \in \mu\Sigma'$ with $fold^A(t) = a$.

Hence $h : A \to C/{\sim}$ is well-defined by $h \circ fold^A = nat_\sim \circ id_{\mu\Sigma'}$.



Since $fold^A$ is epi and predicate preserving and $nat_\sim \circ id_{\mu\Sigma'}$ is Σ-homomorphic, Lemma EMH (1) implies that h is also $\Sigma_{BA'}$-homomorphic.

Let h' be any Σ-homomorphism from A to $C/{\sim}$. Since $B|_{B\Sigma} = \nu\Sigma'$ is initial in Alg_Σ, $h' \circ fold^A = h \circ fold^A$ and thus $h' = h$ because $fold^A$ is epi. ❑

Let $\Sigma = (S, F, P)$ be a destructive signature, $\Sigma' = (S, F, \emptyset)$ and $\nu\Sigma'$ be final in $Alg_{\Sigma'}$.

Lemma PRES

Let $h : A \to B$ be a Σ-homomorphism that preserves all $p \in P$, i.e.,

$$p^B \ = \ h(p^A)$$

and φ be a negation-free Σ-formula.

If φ does not contain universal quantifiers, then

$$f \in \varphi^A \quad \text{implies} \quad h \circ f \in \varphi^B. \tag{3}$$

If h is mono and for all atomic subformulas $r(t_1, \ldots, t_n)$ of φ, $t_1, \ldots, t_n$ are variables, then

$$g \in \varphi^B \quad \text{implies} \quad \exists\, f \in \varphi^A : h \circ f =_{free(\varphi)} g. \tag{4}$$

Proof of (3) by induction on the size of φ.

Let $r \in R$, $s \in S$ and $x \in V_s$. W.l.o.g. we assume that r is unary.

$$f \in r(t)^A \;\Leftrightarrow\; t^A(f) \in r^A \;\Leftrightarrow\; t^B(h \circ f) \overset{Lemma\ EVAL}{=} h(t^A(f)) \in r^B \;\Leftrightarrow\; h \circ f \in r(t)^B.$$

$$f \in (\varphi \wedge \psi)^A = \varphi^A \cap \psi^A \overset{i.h.}{\Rightarrow} h \circ f \in \varphi^B \cap \psi^B = (\varphi \wedge \psi)^B.$$

$$f \in (\varphi \vee \psi)^A = \varphi^A \cup \psi^A \overset{i.h.}{\Rightarrow} h \circ f \in \varphi^B \cup \psi^B = (\varphi \vee \psi)^B.$$

$$\begin{aligned} f \in (\exists x\varphi)^A \;&\Leftrightarrow\; \exists\, a \in A_s : upd(f, x, a) \in \varphi^A \\ &\overset{i.h.}{\Rightarrow}\; \exists\, a \in A_s : upd(h \circ f, x, h(a)) = h \circ upd(f, x, a) \in \varphi^B \\ &\Rightarrow\; \exists\, b \in B_s : upd(h \circ f, x, b) \in \varphi^B \;\Leftrightarrow\; h \circ f \in (\exists x\varphi)^B. \end{aligned}$$

Proof of (4) by induction on the size of φ.

Let $r \in R$, $s \in S$ and $x \in V_s$. W.l.o.g. we assume that r is unary.

$$g \in r(z)^B \;\Leftrightarrow\; g(z) \in r^B \;\Leftrightarrow\; \exists\, a \in r^A : h(a) = g(z)$$

$$\Leftrightarrow\; \exists\, f \in A^X : f(z) \in r^A \wedge h \circ f =_{\{z\}} g \;\Leftrightarrow\; \exists\, f \in r(z)^A : h \circ f =_{\mathit{free}(r(z))} g.$$

$$g \in (\varphi \wedge \psi)^B = \varphi^B \cap \psi^B \overset{i.h.}{\Rightarrow} \exists\, f \in \varphi^A : h \circ f =_{\mathit{free}(\varphi)} g \wedge \exists\, f' \in \psi^A : h \circ f' =_{\mathit{free}(\psi)} g$$

$$\overset{h \text{ mono}}{\Rightarrow} \exists\, f \in \varphi^A \cap \psi^A : h \circ f =_{\mathit{free}(\varphi) \cup \mathit{free}(\psi)} g$$

$$\Leftrightarrow\; \exists\, f \in (\varphi \wedge \psi)^A : h \circ f =_{\mathit{free}(\varphi \wedge \psi)} g.$$

$$g \in (\varphi \vee \psi)^B \overset{analogously}{\Rightarrow} \exists\, f \in (\varphi \vee \psi)^A : h \circ f = g.$$

$$g \in (\exists x \varphi)^B \;\Leftrightarrow\; \exists\, b \in B_s : \mathit{upd}(g, x, b) \in \varphi^B$$

$$\overset{i.h.}{\Rightarrow} \exists\, b \in B_s : \exists\, f \in \varphi^A : h \circ f =_{\mathit{free}(\varphi)} \mathit{upd}(g, x, b)$$

$$\Rightarrow\; \exists\, f \in A^X : \exists\, a \in A_s : \mathit{upd}(f, x, a) \in \varphi^A \wedge h \circ f =_{\mathit{free}(\varphi) \setminus \{x\}} g$$

$$\Rightarrow\; \exists\, f \in (\exists x \varphi)^A : h \circ f =_{\mathit{free}(\exists x \varphi)} g.$$

$$g \in (\forall x \varphi)^B \;\Leftrightarrow\; \forall\, b \in B_s : \mathit{upd}(g, x, b) \in \varphi^B$$

$$\overset{i.h.}{\Rightarrow} \forall\, b \in B_s : \exists\, f \in \varphi^A : h \circ f =_{\mathit{free}(\varphi)} \mathit{upd}(g, x, b)$$

$$\overset{h \text{ mono}}{\Rightarrow} \exists\, f \in A^X : \forall\, a \in A_s : \mathit{upd}(f, x, a) \in \varphi^A \wedge h \circ f =_{\mathit{free}(\varphi) \setminus \{x\}} g$$

$$\Rightarrow\; \exists\, f \in (\forall x \varphi)^A : h \circ f =_{\mathit{free}(\forall x \varphi)} g.$$

❑

Restriction with a greatest invariant

Let AX consist of co-Horn clauses $r(t_1,\dots,t_n) \Rightarrow \psi$ such that for all $A \in Alg_{\Sigma,AX}$, $\in^A$ is a Σ-invariant, $t_1,\dots,t_n$ are variables, $free(\psi) \subseteq \{t_1,\dots,t_n\}$ and ψ is $\forall$-free and membership compatible. Let $C = gfp(\nu\Sigma',\Sigma,AX)$. Then $inv = \in^C$ is the greatest Σ-invariant of $\nu\Sigma'$.

Let $\mathcal{K} = Alg^{\in}_{\Sigma,AX}$. By Lemma INC, $inv \in \mathcal{K}$.

Let $A \in \mathcal{K}$. We define $B \in Alg_\Sigma$ as the $unfold^A$-image of the interpretation of R in A, i.e., for all $r \in R$,

$$r^B =_{def} unfold^A(r^A).$$

Use induction on $\mathbb{N}$ and Kleene's Fixpoint Theorem (or transfinite induction and Zermelo's Fixpoint Theorem ????) to show that $unfold^A$ extends to a Σ-homomorphism!

B satisfies AX and thus $B \in Alg_{\Sigma,AX}$.

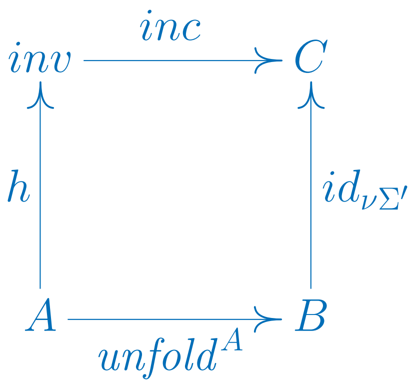
Proof. W.l.o.g. let $\varphi = (r(x_1,\dots,x_n) \Rightarrow \psi) \in AX$ and $g \in r(x_1,\dots,x_n)^B$. Hence $(g(x_1),\dots g(x_n)) \in r^B$ and thus $(f(x_1),\dots,f(x_n)) \in r^A$ and $unfold^A \circ f =_{\{x_1,\dots,x_n\}} g$ for some $f \in A^X$. Hence $f \in \psi^A$ because A satisfies φ, and thus by Lemma PRES (1), $unfold^A \circ f \in \psi^B$. Therefore, $free(\psi) \subseteq \{x_1,\dots,x_n\}$ implies $g \in \psi^B$. ❑

Theorem RESFIN inv is final in $\mathcal{K}$.

Proof. Since C is the greatest $D \in Alg_{\Sigma,AX}$ with $D|_{\Sigma'} = \nu\Sigma'$, we obtain $B \leq C$. In particular,

$$\begin{aligned} img(unfold^A) &= \{unfold^A(a) \mid a \in A\} = \{unfold^A(a) \mid a \in \in^A\} \\ &= \in^B \subseteq \in^C = inv \end{aligned}$$

because $\in^A = A$. Hence $h : A \to inv$ is well-defined by $inc \circ h = id_{\nu\Sigma'} \circ unfold^A$.



Since inc is mono and predicate preserving and $id_{\nu\Sigma'} \circ unfold^A$ is Σ-homomorphic, Lemma EMH (2) implies that h is also Σ-homomorphic.

Let h' be any Σ-homomorphism from A to inv. Since $B|_{\Sigma'} = \nu\Sigma'$ is final in Alg_Σ, $inc \circ h' = inc \circ h$ and thus $h' = h$ because inc is mono. ❑

Restriction with a least invariant

Let AX consist of Horn clauses $r(t_1,\dots,t_n) \Leftarrow \psi$ such that for all $A \in Alg_{\Sigma,AX}$, $\in^A$ is a Σ-invariant, $free(r(t_1,\dots,t_n)) \subseteq free(\psi)$, ψ is membership compatible and for all atomic subformulas $p(u_1,\dots,u_m)$ of ψ, $u_1,\dots,u_m$ are variables. Let $C = lfp(\nu\Sigma',\Sigma,AX)$ and $inv = \in^C$ be a Σ-invariant of $\nu\Sigma'$. Hence $C \in obs(Alg_{\Sigma,AX})$.

Let $\mathcal{K} = obs(Alg^{\in}_{\Sigma,AX})$. By Lemma INC, $inv \in \mathcal{K}$.

Let $A \in \mathcal{K}$. We define $B \in Alg_\Sigma$ as the $unfold^A$-image of the interpretation of R in A, i.e., for all $r \in R$,

$$r^B \;=_{def}\; unfold^A(r^A).$$

Use induction on $\mathbb{N}$ and Kleene's Fixpoint Theorem (or transfinite induction and Zermelo's Fixpoint Theorem ????) to show that $unfold^A$ extends to a Σ-homomorphism!

B satisfies AX and thus $B \in obs(Alg_{\Sigma,AX})$.

Proof. Let $\varphi = (r(t_1,\dots,t_n) \Leftarrow \psi) \in AX$ and $g \in \psi^B$. Since A is Σ-observable, $unfold^A$ is mono and thus Lemma PRES (2) implies $g =_{free(\psi)} unfold^A \circ f$ for some $f \in \psi^A$. Since A satisfies φ, $f \in r(t_1,\dots,t_n)^A$ and thus $(t_1^A(f),\dots,t_n^A(f)) \in r^A$. Hence

$$(t_1^B(unfold^A \circ f),\dots,t_n^B(unfold^A \circ f)) \stackrel{Lemma\ EVAL}{=} (unfold^A(t_1^A(f)),\dots,unfold^A(t_n^A(f))) \in r^B$$

and thus $unfold^A \circ f \in r(t_1,\dots,t_n)^B$. Therefore, $free(r(t_1,\dots,t_n)) \subseteq free(\psi)$ implies $g \in r(t_1,\dots,t_n)^B$. ❑

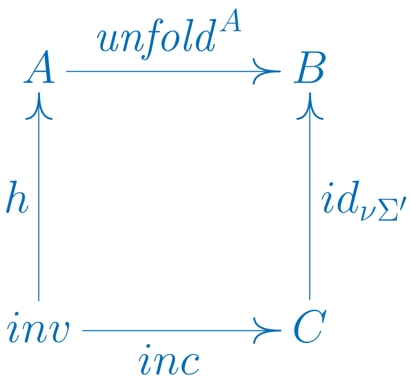
Theorem RESINI *inv* is initial in $obs(Alg^{\in}_{\Sigma,AX})$.

Proof. Since C is the least $D \in Alg_{\Sigma,AX}$ with $D|_{\Sigma'} = \nu\Sigma'$, we obtain $C \leq B$.

In particular,

$$inv \;=\; \in^C \;\subseteq\; \in^B \;=\; \{unfold^A(a) \mid a \in \in^A\} \;=\; \{unfold^A(a) \mid a \in A\}$$
$$=\; img(unfold^A) \qquad (*)$$

because $\in^A = A$. Since A is Σ-observable, $unfold^A$ is mono and thus for all $a, b \in A$, $unfold^A(a) = unfold^A(b)$ implies $a = b$. Hence by $(*)$, $h : inv \to A$ with $h(b) = (unfold^A)^{-1}(b)$ for all $b \in inv$ is well-defined. Therefore, $unfold^A \circ h = id_{\nu\Sigma'} \circ inc$.



Since $unfold^A$ is mono and predicate preserving and $id_{\nu\Sigma'} \circ inc$ is Σ-homomorphic, Lemma EMH (2) implies that h is also Σ-homomorphic.

Let h' be any Σ-homomorphism from inv to A. Since $B|_{B\Sigma} = BA$ is final in Alg_Σ, $unfold^A \circ h' = unfold^A \circ h$ and thus $h' = h$ because $unfold^A$ is mono. ❏

Notational conventions

Let $\Sigma = (S, BS, F, P)$ is a constructive resp. destructive signature.

$\mu\Sigma$ resp. $\nu\Sigma$ denotes the initial resp. final object of Alg_Σ.

We simply write f for the interpretation of $f \in F$ in $\mu\Sigma$ resp. $\nu\Sigma$.

The only argument of a function with domain 1 is omitted.
For instance, 0 stands for $0(*)$, nil stands for $nil(*)$.

Natural numbers

$$\begin{aligned} S &= \{nat\}, \\ F &= \{zero : 1 \to nat,\ succ : nat \to nat\}, \\ F' &= \{pred : nat \to 1 + nat\}, \\ Nat &= (S, F, \emptyset), \\ coNat &= (S, F', \emptyset). \end{aligned}$$

- For all $A \in Set^S$, $H_{Nat}(A)_{nat} = H_{coNat}(A)_{nat} = 1 + A_{nat}$.
- $\mu Nat_{nat} \cong \mathbb{N}$.
- $zero = 0$ and for all $n \in \mathbb{N}$, $succ(n) = n + 1$.
- $\nu coNat_{nat} \cong \mathbb{N}' =_{def} \mathbb{N} \cup \{\infty\}$.
- For all $n \in \mathbb{N}'$, $pred(n) = \begin{cases} * & \text{if } n = 0, \\ n - 1 & \text{if } n > 0, \\ \infty & \text{if } n = \infty. \end{cases}$

1.1 Recursion and currying: Addition on $\mathbb{N}$

The function $plus : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ satisfies the equations

$$plus(zero, n) = n \quad (1)$$
$$plus(succ(m), n) = succ(plus(m, n)) \quad (2)$$

Define $\mathcal{K} = Set$ and for all $A \in Set$, $L(A)_{nat} = A_{nat} \times \mathbb{N}$ and $R(A)_{nat} = A_{nat}^{\mathbb{N}}$.

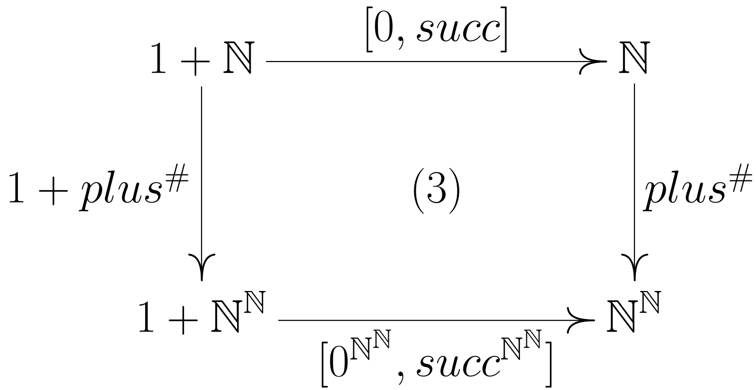
By (2), the kernel of $plus^{\#} : \mathbb{N} \to \mathbb{N}^{\mathbb{N}}$ is compatible with $succ$:

$$plus^{\#}(m) = plus^{\#}(n)$$
$$\Rightarrow plus^{\#}(succ(m)) = \lambda i.plus(succ(m), i) = \lambda i.succ(plus(m, i)) = \lambda i.succ(plus^{\#}(m)(i))$$
$$= \lambda i.succ(plus^{\#}(n)(i)) = \lambda i.succ(plus(n, i)) = \lambda i.plus(succ(n), i) = plus^{\#}(succ(n)).$$

Hence *plus* is *Nat*-recursive and thus by Lemma REC, $plus^{\#}$ agrees with $fold^{\mathbb{N}^{\mathbb{N}}}$ where

$$0^{\mathbb{N}^{\mathbb{N}}} = \lambda n.n,$$
$$succ^{\mathbb{N}^{\mathbb{N}}} = \lambda f.\lambda n.(f(n) + 1).$$

The validity of (1) and (2) is equivalent to the commutativity of (3):



1.2 Corecursion and coproduct: Addition on $\mathbb{N} \cup \{\infty\}$ (see [33])

The function $plus : \mathbb{N}' \times \mathbb{N}' \to \mathbb{N}'$ satisfies the equations

$$pred(plus(0,0)) = * \quad (1)$$

$$n \neq 0 \Rightarrow pred(plus(0,n)) = id(pred(n)) \quad (2)$$

$$m \neq 0 \Rightarrow pred(plus(m,n)) = plus(pred(m),n) \quad (3)$$

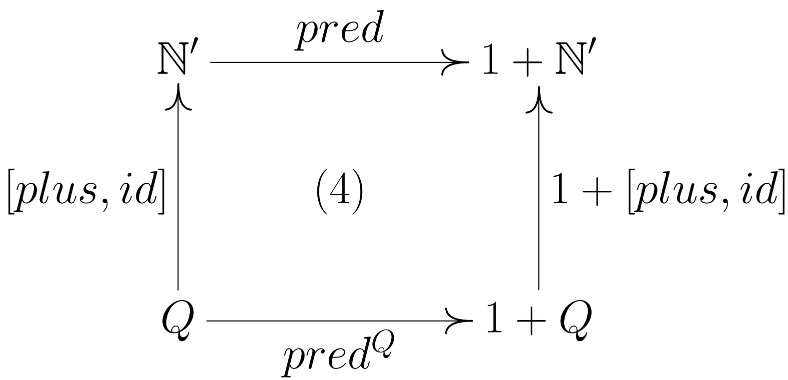
Define $\mathcal{K} = Set^2$ and for all $A, B \in Set$, $L(A,B)_{nat} = A_{nat} + B_{nat}$ and $R(A)_{nat} = (A_{nat}, A_{nat})$.

Let $Q = \mathbb{N}' \times \mathbb{N}' + \mathbb{N}'$. By (1)-(3), the image of $(plus, id)^* = [plus, id] : Q \to \mathbb{N}'$ is compatible with $pred$.

Hence $(plus, id) : (\mathbb{N}' \times \mathbb{N}', \mathbb{N}') \to (\mathbb{N}', \mathbb{N}')$ is $coNat$-corecursive and thus by Lemma COR, $[plus, id]$ agrees with $unfold^Q$ where for all $m, n \in \mathbb{N}'$,

$$\begin{aligned} pred^Q(m,n) &= \begin{cases} * & \text{if } m = n = 0, \\ (0, n-1) & \text{if } m = 0 \wedge n \in \mathbb{N}' \setminus \{0\}, \\ (m-1, n) & \text{if } m \in \mathbb{N}' \setminus \{0\}, \end{cases} \\ pred^Q(n) &= pred(n). \end{aligned}$$

The validity of (1)-(3) is equivalent to the commutativity of (4):



1.3 Recursion and product: Factorial numbers (see [28])

Let $n \in \mathbb{N}$. The function $fact : \mathbb{N} \to \mathbb{N}$ satisfies the equations

$$\begin{aligned} \langle fact, id\rangle(zero) &= (1,0) && (1)\\ \langle fact, id\rangle(succ(n)) &= (fact(n) * (id(n)+1), id(n)+1) && (2) \end{aligned}$$

Define $\mathcal{K} = Set^2$ and for all $A, B \in Set$, $L(A)_{nat} = (A_{nat}, A_{nat})$ and $R(A,B)_{nat} = A_{nat} \times B_{nat}$.

By (1) and (2), the kernel of $(fact, id)^{\#} = \langle fact, id\rangle : \mathbb{N} \to \mathbb{N} \times \mathbb{N}$ is compatible with $succ$:

$$\begin{aligned} &(fact(m), id(m)) = \langle fact, id\rangle(m) = \langle fact, id\rangle(n) = (fact(n), id(n))\\ &\quad \Rightarrow \langle fact, id\rangle(m+1) = (fact(m+1), id(m+1))\\ &\qquad = (fact(m) * (id(m)+1), id(m)+1) = (fact(n) * (id(n)+1), id(n)+1)\\ &\qquad = (fact(n+1), id(n+1)) = \langle fact, id\rangle(n+1). \end{aligned}$$

Hence $(fact, id) : (\mathbb{N}, \mathbb{N}) \rightarrow (\mathbb{N}, \mathbb{N})$ is Nat-recursive and thus by Lemma REC, $\langle fact, id\rangle$ agrees with $fold^{\mathbb{N}\times\mathbb{N}}$ where

$$\begin{aligned} 0^{\mathbb{N}\times\mathbb{N}} &= (1, 0), \\ succ^{\mathbb{N}\times\mathbb{N}} &= \lambda(m, n).(m * (n + 1), n + 1). \end{aligned}$$

The validity of (1) and (2) is equivalent to the commutativity of (3):

$$\begin{array}{ccc} 1 + \mathbb{N} & \xrightarrow{[0,\, succ]} & \mathbb{N} \\ {\scriptstyle 1 + \langle fact, id\rangle}\downarrow & (3) & \downarrow{\scriptstyle \langle fact, id\rangle} \\ 1 + \mathbb{N} \times \mathbb{N} & \xrightarrow[{[0^{\mathbb{N}\times\mathbb{N}},\, succ^{\mathbb{N}\times\mathbb{N}}]}]{} & \mathbb{N} \times \mathbb{N} \end{array}$$

1.4 Recursion and product: Fibonacci numbers (see [28])

The function $\mathit{fib} : \mathbb{N} \to \mathbb{N}$ satisfies the equations

$$\begin{aligned} \mathit{fib}(zero) &= 0 \\ \mathit{fib}(succ(zero)) &= 1 \\ \mathit{fib}(succ(succ(n))) &= \mathit{fib}(n) + \mathit{fib}(succ(n)) \end{aligned}$$

Again, these equations do not imply that the kernel of fib is a Σ-congruence.

We regard the composition $\mathit{fib} \circ succ$ as a further function $\mathit{fib}' : \mathbb{N} \to \mathbb{N}$ and transform the above equations into a mutually recursive definition of fib and fib':

$$\langle \mathit{fib}, fib' \rangle(zero) = (0, 1) \tag{1}$$

$$\langle \mathit{fib}, fib' \rangle(succ(n)) = (\mathit{fib}'(n), \mathit{fib}(n) + \mathit{fib}'(n)) \tag{2}$$

Define $\mathcal{K} = Set^2$ and for all $A, B \in Set$, $L(A)_{nat} = (A_{nat}, A_{nat})$ and $R(A, B)_{nat} = A_{nat} \times B_{nat}$.

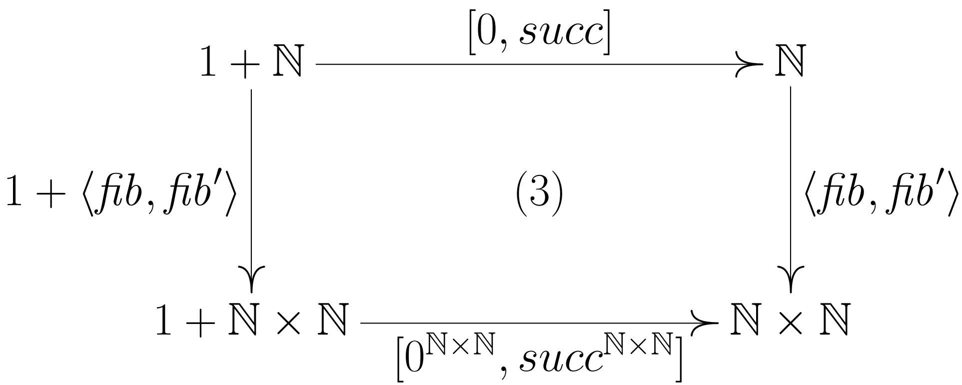
By (1) and (2), the kernel of $(fib, fib')^{\#} = \langle fib, fib' \rangle : \mathbb{N} \to \mathbb{N} \times \mathbb{N}$ is compatible with $succ$:

$$(fib(m), fib'(m)) = \langle fib, fib' \rangle(m) = \langle fib, fib' \rangle(n) = (fib(n), fib'(n))$$
$$\Rightarrow \langle fib, fib' \rangle(succ(m)) = (fib(succ(m)), fib'(succ(m))) = (fib'(m), fib(m) + fib'(m))$$
$$= (fib'(n), fib(n) + fib'(n)) = (fib(succ(n)), fib'(succ(n))) = \langle fib, fib' \rangle(succ(n)).$$

Hence $(fib, fib') : (\mathbb{N}, \mathbb{N}) \to (\mathbb{N}, \mathbb{N})$ is Nat-recursive and thus by Lemma REC, $\langle fib, fib' \rangle$ agrees with $fold^{\mathbb{N}\times\mathbb{N}}$ where

$$0^{\mathbb{N}\times\mathbb{N}} = (0, 1),$$
$$succ^{\mathbb{N}\times\mathbb{N}} = \lambda(m, n).(n, m + n).$$

The validity of (1) and (2) is equivalent to the commutativity of (3):



1.5 Recursion and currying: Replication

Let X be a set. The function $repl : \mathbb{N} \times X \to X^*$ satisfies the equations

$$repl(zero, e) = nil \quad (1)$$

$$repl(succ(n), e) = cons(e, repl(n, e)) \quad (2)$$

where $nil = nil^{\mu List(X)}$ and $cons = cons^{\mu List(X)}$ (see Lists and Streams).

Define $\mathcal{K} = Set$ and for all $A \in Set$, $L(A)_{nat} = A \times X$ and $R(A)_{nat} = A^X$.

Let $Z = (X^*)^X$. By (2), the kernel of $repl^\# : \mathbb{N} \to Z$ is compatible with $succ$:

$$\begin{aligned} & repl^\#(m) = repl^\#(n) \\ & \Rightarrow repl^\#(succ(m)) = \lambda e.cons(e, repl^\#(m)(e)) = \lambda e.cons(e, repl(m, e)) \\ & \quad = \lambda e.cons(e, repl(n, e)) = \lambda e.cons(e, repl^\#(n)(e)) = repl^\#(succ(n)). \end{aligned}$$

Hence $repl$ is Nat-recursive and thus by Lemma REC, $repl^\#$ agrees with $fold^Z$ where

$$0^Z = \lambda e.\epsilon,$$

$$succ^Z = \lambda f.\lambda e.(e : f(e)).$$

The validity of (1) and (2) is equivalent to the commutativity of (3):

$$\begin{array}{ccc}
1+\mathbb{N} & \xrightarrow{[0,\,succ]} & \mathbb{N} \\
{\scriptstyle 1+repl^{\#}}\downarrow & (3) & \downarrow{\scriptstyle repl^{\#}} \\
1+Z & \xrightarrow[{[0^Z,\,succ^Z]}]{} & Z
\end{array}$$

1.6 Corecursion and identity: Length of a colist

Let X be a set. The function $length : X^\infty \to \mathbb{N}'$ satisfies the equations

$$pred(length(\epsilon)) = * \quad (1)$$

$$s \in X^* \;\Rightarrow\; pred(length(x\!:\!s)) = length(s) \quad (2)$$

$$s \in X^{\mathbb{N}} \;\Rightarrow\; pred(length(s)) = length(s) \quad (3)$$

Define $\mathcal{K} = Set$ and $L = R = Id_{Set}$.

By (1)-(3), the image of $length$ is compatible with $pred$. To see this, complete $length$ to an S-sorted function h with $h_{nat} = length$ and $h_1 = id_1$. Then (1)-(3) imply

$$pred(h(\epsilon)) = * = h(*),$$
$$s \in X^* \;\Rightarrow\; pred(h(x:s)) = h(s),$$
$$s \in X^{\mathbb{N}} \;\Rightarrow\; pred(h(s)) = h(s),$$

i.e., the image of h is compatible with $pred$.

Hence $length$ is $coNat$-corecursive and thus by Lemma COR, $length$ agrees with $unfold^{X^\infty}$ where for all $s \in X^\infty$,

$$pred^{X^\infty}(s) = \begin{cases} * & \text{if } s = \epsilon, \\ s' & \text{if } s = x:s' \text{ for some } x \in X \text{ and } s' \in X^*, \\ \lambda n.s(n+1) & \text{if } s \in X^{\mathbb{N}}. \end{cases}$$

The validity of (1)-(3) is equivalent to the commutativity of (4):

$$
\begin{array}{ccc}
\mathbb{N}' & \xrightarrow{\quad pred \quad} & 1 + \mathbb{N}' \\
\uparrow{\scriptstyle length} & (4) & \uparrow{\scriptstyle 1 + length} \\
X^{\infty} & \xrightarrow[\quad pred^{X^{\infty}} \quad]{} & 1 + X^{\infty}
\end{array}
$$

Lists and streams

Let X be a set.

$$\begin{aligned}
S &= \{list\}, \\
F &= \{nil : 1 \to list,\ cons : X \times list \to list\}, \\
F' &= \{split : list \to 1 + (X \times list)\}, \\
F'' &= \{head : list \to X,\ tail : list \to list\}, \\
List(X) &= (S, F, \emptyset), \\
coList(X) &= (S, F', \emptyset), \\
Stream(X) &= (S, F'', \emptyset).
\end{aligned}$$

- For all $A \in Set^S$, $H_{List(X)}(A)_{list} = H_{coList(X)}(A)_{list} = 1 + X \times A_{list}$ and $H_{Stream(X)}(A)_{list} = X \times A_{list}$.
- $\mu List(X)_{list} \cong X^*$.
- $nil = \epsilon$ and for all $x \in X$ and $s \in X^*$, $cons(x, s) = x : s$.
- $\nu coList(X)_{list} \cong X^\infty$.

- For all $s \in X^\infty$,

$$split(s) = \begin{cases} * & \text{if } s = \epsilon, \\ (x, s') & \text{if } \exists\, x \in X,\ s' \in X^\infty : s = x{:}s', \\ (s(0), \lambda n.s(n+1)) & \text{if } s \in X^{\mathbb{N}}. \end{cases}$$

- $\nu Stream_{list} \cong X^{\mathbb{N}}$.
- For all $s \in X^{\mathbb{N}}$, $head(s) = s(0)$ and $tail(s) = \lambda n.s(n+1)$.

2.1 Constructor extension: Replication

In 1.5 we have shown that there is a unique interpretation in $\mu List(X)$ of an additional constructor $repl : \mathbb{N} \times X \to list$ such that the corresponding extension of $\mu List(X)$ satisfies the equations (1) and (2) of 1.5.

Let $\Sigma = (S, F \cup \{repl\}, \{=: list \times list\})$, $\Sigma' = (S, F \cup \{repl\}, \emptyset)$ and AX be a set of Σ-Horn clauses such that for all $A \in Alg_{\Sigma,AX}$, $=^A$ is a Σ-congruence, and AX includes (1) and (2) of 1.5.

Let $A = lfp(\Sigma, \mu\Sigma', AX)$. By Theorem ABSINI, $A/{=^A}$ is initial in $Alg^{=}_{\Sigma,AX}$. Since the initial $List(X)$-algebra with equality can be extended to a (Σ, AX)-algebra with equality, we conclude from Lemma CONEXT that (Σ, AX) is a conservative extension of $(List(X), \emptyset)$.

2.2 Recursion and identity: Length of a finite list

The function $length : X^* \to \mathbb{N}$ satisfies the equations

$$length(nil) = 0 \quad (1)$$

$$length(cons(x,s)) = length(s) + 1 \quad (2)$$

Define $\mathcal{K} = Set$ and $L = R = Id_{Set}$.

By (2), the kernel of *length* is compatible with *cons*:

$$length(s) = length(s')$$

$$\Rightarrow length(cons(x,s)) = length(s) + 1 = length(s') + 1 = length(cons(x,s')).$$

Hence *length* is $List(X)$-recursive and thus by Lemma REC, *length* agrees with $fold^{\mathbb{N}}$ where $nil^{\mathbb{N}} = 0$ and $cons^{\mathbb{N}} = \lambda(x,n).n+1$.

The validity of (1) and (2) is equivalent to the commutativity of (3):

$$
\begin{array}{ccc}
1 + X \times X^* & \xrightarrow{[nil, cons]} & X^* \\
{\scriptstyle 1 + X \times length} \downarrow & (3) & \downarrow {\scriptstyle length} \\
1 + X \times \mathbb{N} & \xrightarrow[{[nil^{\mathbb{N}}, cons^{\mathbb{N}}]}]{} & \mathbb{N}
\end{array}
$$

2.3 Destructor extension: Length of a colist

In 1.6 we have shown that there is a unique interpretation in $\nu coList(X)$ of an additional destructor $length : list \to nat + 1$ such that the corresponding extension of $\nu coList(X)$ satisfies the equations (1)-(3) of 1.6.

Let $\Sigma = (S, F' \cup \{length\}, \{\in: list\})$, $\Sigma' = (S, F' \cup \{length\}, \emptyset)$ and AX be a set of Σ-co-Horn clauses such that for all $A \in Alg_{\Sigma,AX}$, $\in^A$ is a Σ-invariant, and AX includes the following co-Horn clauses:

$$\begin{aligned}
\in_{list}(s) &\Rightarrow (length(s) = 0 \Rightarrow split(s)) \\
\in_{list}(s) &\Rightarrow (length(s) = n+1 \Rightarrow \exists\, x, s' : (split(s) = (x, s') \wedge length(s') = n)) \\
\in_{list}(s) &\Rightarrow (length(s) = * \Rightarrow \exists\, x, s' : (split^B(s) = (x, s') \wedge length(s') = *).
\end{aligned}$$

Let $A = gfp(\Sigma, \nu\Sigma', AX)$. By Theorem RESFIN, $\in^A$ is final in $Alg^{\in}_{\Sigma,AX}$. Since the final $coList(X)$-algebra with membership can be extended to a (Σ, AX)-algebra with membership, we conclude from Lemma DESEXT that (Σ, AX) is a conservative extension of $(coList(X), \emptyset)$.

2.4 Recursion and currying: Concatenation of finite lists

The function $conc : X^* \times X^* \to X^*$ satisfies the equations

$$conc(nil, s) = s \tag{1}$$

$$conc(cons(x, s), s') = cons(x, conc(s, s')) \tag{2}$$

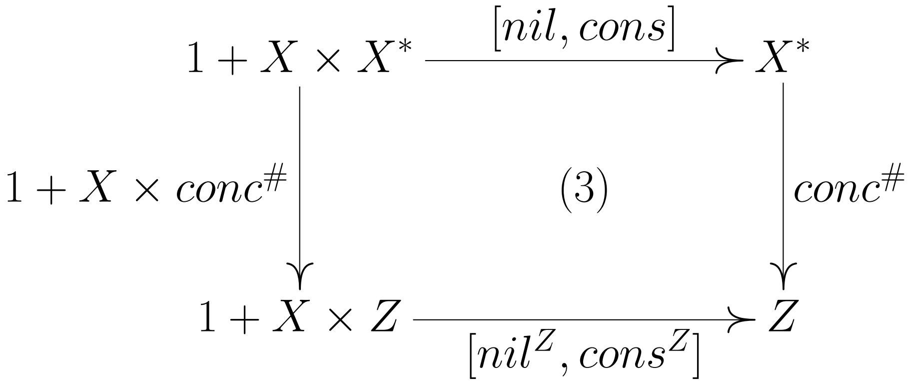
Define $\mathcal{K} = Set$ and for all $A \in Set$, $L(A)_{list} = A_{list} \times X^*$ and $R(A)_{list} = A_{list}^{X^*}$.

Let $Z = (X^*)^{X^*}$. By (2), the kernel of $conc^\# : X^* \to Z$ is compatible with $cons$:

$$conc^\#(s) = conc^\#(s')$$
$$\Rightarrow conc^\#(cons(x,s)) = \lambda s''.conc(cons(x,s),s'') = \lambda s''.cons(x, conc(s,s''))$$
$$= \lambda s''.cons(x, conc^\#(s)(s'')) = \lambda s''.cons(x, conc^\#(s')(s''))$$
$$= \lambda s''.cons(x, conc(s',s'')) = \lambda s''.conc(cons(x,s'),s'') = conc^\#(cons(x,s')).$$

Hence $conc$ is $List(X)$-recursive and thus by Lemma REC, $conc^\#$ agrees with $fold^Z$ where $nil^Z = \lambda s.s$ and $cons^Z = \lambda(x,f).\lambda s.cons(x, f(s))$.

The validity of (1) and (2) is equivalent to the commutativity of (3):



2.5 Corecursion and coproduct: Concatenation of colists (see [33])

The function $conc : X^\infty \times X^\infty \to X^\infty$ satisfies the equations

$$split(s) = * \wedge split(s') = * \;\Rightarrow\; split(conc(s,s')) = * \tag{1}$$
$$split(s) = * \wedge split(s') = (x,s'') \;\Rightarrow\; split(conc(s,s')) = (x, id(s'')) \tag{2}$$
$$split(s) = (x,s'') \;\Rightarrow\; split(conc(s,s')) = (x, conc(s'',s')) \tag{3}$$

Define $\mathcal{K} = Set^2$ and for all $A, B \in Set$, $R(A)_{list} = (A_{list}, A_{list})$ and $L(A,B)_{list} = A_{list} + B_{list}$.

Let $Q = X^\infty \times X^\infty + X^\infty$. By (1)-(3), the image of $(conc, id)^* = [conc, id] : Q \to X^\infty$ is compatible with $split$: Let $h = [conc, id]$.

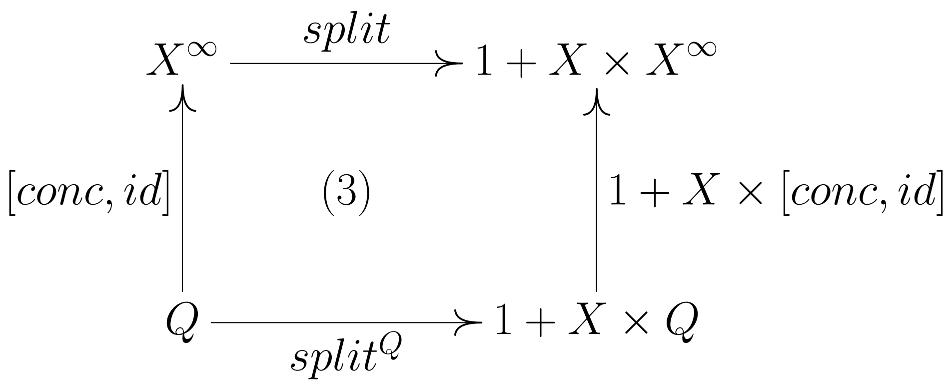
$$\begin{aligned}
split(s) = * \wedge split(s') = * \;&\Rightarrow\; split(h(s,s')) = * = h(*),\\
split(s) = * \wedge split(s') = (x,s'')&\\
&\Rightarrow\; split(h(s,s')) = (x, h(s'')) = (h(x), h(s'')) = h(x,s''),\\
split(s) = (x,s'') \;&\Rightarrow\; split(h(s,s')) = (x, h(s'',s')) = (h(x), h(s'',s')) = h(x,(s'',s')),
\end{aligned}$$

i.e., the image of h is compatible with $split$.

Hence $(conc, id)$ is $coList(X)$-corecursive and thus by Lemma COR, $(conc, id)$ agrees with $unfold^Q$ where for all $s, s' \in X^\infty$,

$$split^Q(s,s') = \begin{cases} * & \text{if } split(s) = split(s') = *, \\ (x,(s,s'')) & \text{if } split(s) = * \wedge split(s') = (x,s''), \\ (x,(s'',s')) & \text{if } split(s) = (x,s''), \end{cases}$$
$$split^Q(s) = split(s).$$

The validity of (1)-(3) is equivalent to the commutativity of (4):



2.6 Recursion and identity: Folding a finite list from the right

Let A be a set and $Z = (X \times A \to A) \to A \to A$.
The function $foldr : X^* \to (X \times A \to A) \to A \to A$ satisfies the equations

$$foldr(nil)(f)(a) \;=\; a \qquad (1)$$
$$foldr(cons(e,s))(f)(a) \;=\; f(e, foldr(s)(f)(a)) \qquad (2)$$

Define $\mathcal{K} = Set$ and $L = R = Id_{Set}$.

By (2), the kernel of $foldr$ is compatible with $cons$:

$$\begin{aligned} & foldr(s) = foldr(s') \\ & \quad \Rightarrow \; foldr(cons(x,s)) = \lambda f.\lambda a.f(e, foldr(s)(f)(a)) = \lambda f.\lambda a.f(x, foldr(s')(f)(a)) \\ & \quad\quad = foldr(cons(x,s')). \end{aligned}$$

Hence $foldr$ is $List(X)$-recursive and thus by Lemma REC, $foldr$ agrees with $fold^Z$ where for all $f : X \times A \to A$, $a \in A$, $x \in X$ and $g \in Z$,

$$nil^Z(f)(a) \;=\; a,$$
$$cons^Z(x,g)(f)(a) \;=\; \lambda s.g(f)(a)(x:s).$$

The validity of (1) and (2) is equivalent to the commutativity of (3):

$$\begin{array}{ccc}
1 + X \times X^* & \xrightarrow{[nil, cons]} & X^* \\
{\scriptstyle 1 + X \times foldr} \downarrow & (3) & \downarrow {\scriptstyle foldr} \\
1 + X \times Z & \xrightarrow[{[nil^Z, cons^Z]}]{} & Z
\end{array}$$

2.7 Recursion and identity: Filter a finite list

Let $Z = (X \to 2) \to X^*$. The function $filter : X^* \to Z$ satisfies the equations

$$\begin{array}{rcll}
filter(nil)(f) & = & nil & (1) \\
filter(cons(x, s))(f) & = & if\ f(x)\ then\ filter(s)(f)\ else\ x : filter(s)(f) & (2)
\end{array}$$

Define $\mathcal{K} = Set$ and $L = R = Id_{Set}$.

By (2), the kernel of *filter* is compatible with *cons*:

$$
\begin{aligned}
&\mathit{filter}(s) = \mathit{filter}(s')\\
&\Rightarrow \mathit{filter}(\mathit{cons}(x,s)) = \lambda f.\mathit{if}\ \ f(x)\ \mathit{then}\ \mathit{filter}(s)(f)\ \mathit{else}\ x\!:\!\mathit{filter}(s)(f)\\
&\quad = \lambda f.\mathit{if}\ \ f(x)\ \mathit{then}\ \mathit{filter}(s')(f)\ \mathit{else}\ x\!:\!\mathit{filter}(s')(f) = \mathit{filter}(\mathit{cons}(x,s')).
\end{aligned}
$$

Hence *filter* is *List*(X)-recursive and thus by Lemma REC, *filter* agrees with fold^Z where for all $f : X \to 2$, $x \in X$ and $g \in Z$, $\mathit{nil}^Z(f) = \mathit{nil}$ and $\mathit{cons}^Z = \lambda(x,g).\lambda f.\lambda s.g(f)(x:s)$.

The validity of (1) and (2) is equivalent to the commutativity of (3):

$$
\begin{array}{ccc}
1+X\times X^* & \xrightarrow{[\mathit{nil},\,\mathit{cons}]} & X^* \\
{\scriptstyle 1+X\times \mathit{filter}}\downarrow & (3) & \downarrow{\scriptstyle \mathit{filter}} \\
1+X\times Z & \xrightarrow[{[\mathit{nil}^Z,\,\mathit{cons}^Z]}]{} & Z
\end{array}
$$

2.8 Corecursion and coproduct: A blinker

Suppose that $on, off \in X$. The functions $blink : 1 \to X^{\mathbb{N}}$ and $blink' : 1 \to X^{\mathbb{N}}$ satisfy the equations

$$\langle head, tail\rangle(blink) = (on, blink') \quad (1)$$
$$\langle head, tail\rangle(blink') = (off, blink) \quad (2)$$

Define $\mathcal{K} = Set^2$ and for all $A, B \in Set$, $R(A)_{list} = (A_{list}, A_{list})$ and $L(A, B)_{list} = A_{list} + B_{list}$.

Let $Q = 1 + 1$. By (1) and (2), the image of $(blink, blink')^* = [blink, blink'] : Q \to X^{\mathbb{N}}$ is compatible with *head* and *tail*.

Hence $(blink, blink') : Q \to (X^{\mathbb{N}}, X^{\mathbb{N}})$ is $Stream(X)$-corecursive and thus by Lemma COR, $[blink, blink']$ agrees with $unfold^Q$ where $\langle head^Q, tail^Q\rangle(*, 1) = (on, (*, 2))$ and $\langle head^Q, tail^Q\rangle(*, 2) = (off, (*, 1))$.

The validity of (1) and (2) is equivalent to the commutativity of (3):

$$\begin{CD} X^{\mathbb{N}} @>{\langle head, tail\rangle}>> X \times X^{\mathbb{N}} \\ @A{[blink, blink']}AA (3) @AA{X \times [blink, blink']}A \\ Q @>>{\langle head^Q, tail^Q\rangle}> X \times Q \end{CD}$$

$f : \{x, y\} \to X^{\mathbb{N}}$ with $f(x) = blink$ and $f(y) = blink'$ solves the set $\{x = cons(1, y),\ y = cons(0, x)\}$ of $Stream$-equations (see Recursive Σ-equations).

2.9 Corecursion and coproduct: Alternation of successors and squares (see [28])

The functions $nats : \mathbb{N} \to X^{\mathbb{N}}$ and $squares : \mathbb{N} \to X^{\mathbb{N}}$ satisfy the equations

$$\langle head, tail\rangle(nats(n)) = (n, squares(n)) \quad (1)$$

$$\langle head, tail\rangle(squares(n)) = (n * n, nats(n + 1)) \quad (2)$$

Define $\mathcal{K} = Set^2$ and for all $A, B \in Set$, $R(A)_{list} = (A_{list}, A_{list})$ and

$$L(A, B)_{list} = A_{list} + B_{list}.$$

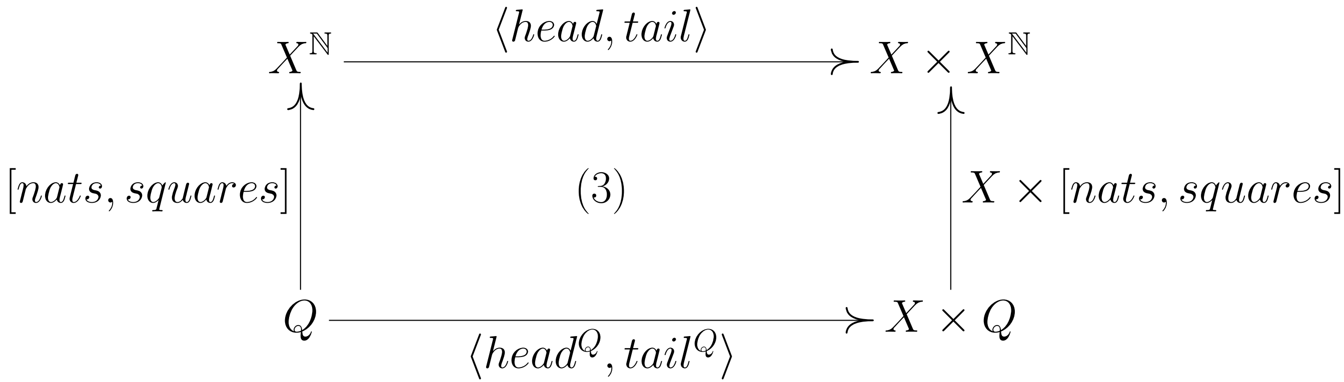
Let $Q = \mathbb{N} + \mathbb{N}$. By (1) and (2), the image of

$$(nats, squares)^* = [nats, squares] : Q \to X^{\mathbb{N}}$$

is compatible with *head* and *tail*.

Hence $(nats, squares) : (\mathbb{N}, \mathbb{N}) \to (X^{\mathbb{N}}, X^{\mathbb{N}})$ is *Stream*-recursive and thus by Lemma COR, $[nats, squares]$ agrees with $unfold^Q$ where for all $n \in \mathbb{N}$,
$\langle head^Q, tail^Q \rangle(n, 1) = (n, (n, 2))$ and $\langle head^Q, tail^Q \rangle(n, 2) = (n * n, (n + 1, 1))$.

The validity of (1) and (2) is equivalent to the commutativity of (3):



2.10 Corecursion and coproduct: Insertion into a stream (see [65])

The function $insert : X \times X^{\mathbb{N}} \to X^{\mathbb{N}}$ satisfies the equation

$$\langle head, tail\rangle(insert(x,s)) = if\ \ x \leq head(s)\ then\ (x,s)\ else(head(s), insert(x, tail(s)))$$

Analogously to 1.3, this equation does not imply that the image of *insert* is compatible with *head* and *tail*. Hence we transform them into equations for *insert and* the identity on $X^{\mathbb{N}}$:

$$\begin{aligned} \langle head, tail\rangle(insert(x,s)) &= if\ \ x \leq head(s) \\ &\quad\ then\ (x, id(s))\ else\ (head(s), insert(x, tail(s))) & (1) \\ \langle head, tail\rangle(id(s)) &= (head(s), id(tail(s))) & (2) \end{aligned}$$

Define $\mathcal{K} = Set^2$ and for all $A, B \in Set$, $R(A)_{list} = (A_{list}, A_{list})$ and $L(A,B)_{list} = A_{list} + B_{list}$.

Let $Q = (X \times X^{\mathbb{N}}) + X^{\mathbb{N}}$. By (1)-(3), the image of

$$(insert, id)^* = [insert, id] : Q \to X^{\mathbb{N}}$$

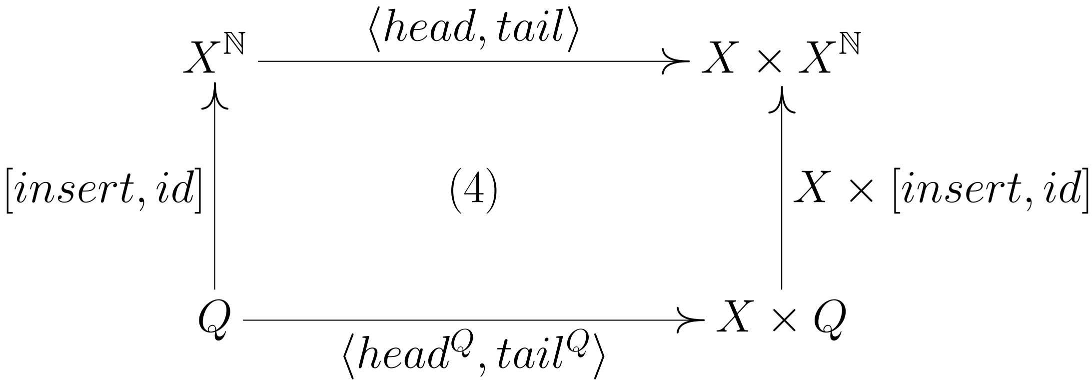
is compatible with *head* and *tail*.

Hence $(insert, id) : (X \times X^{\mathbb{N}}, X^{\mathbb{N}}) \to (X^{\mathbb{N}}, X^{\mathbb{N}})$ is *Stream*-corecursive and thus by Lemma COR, $[insert, id]$ agrees with $unfold^Q$ where for all $e \in X$ and $s \in X^{\mathbb{N}}$,

$$\langle head^Q, tail^Q \rangle(x, s) = \begin{cases} (x, s) & \text{if } e \leq head(s), \\ (head(s), (x, tail(s))) & \text{otherwise}, \end{cases}$$

$$\langle head^Q, tail^Q \rangle(s) = (head(s), tail(s)).$$

The validity of (1)-(3) is equivalent to the commutativity of (4):



2.11 Corecursion and coproduct: Exchange stream elements (see [65])

The function $exch : X^{\mathbb{N}} \to X^{\mathbb{N}}$, which exchanges each two consecutive elements of a strem, satisfies the equations

$$\begin{aligned} head(exch(s)) &= head(tail(s)) \\ \langle head, tail\rangle(tail(exch(s))) &= (head(s), exch(tail(tail(s)))) \end{aligned}$$

Analogously to 1.4, we regard the composition $tail \circ exch$ as a further function

$$exch' : X^{\mathbb{N}} \to X^{\mathbb{N}}$$

and transform the above equations into a mutually recursive definition of $exch$ and $exch'$:

$$\langle head, tail\rangle(exch(s)) = (head(tail(s)), exch'(s)) \tag{1}$$

$$\langle head, tail\rangle(exch'(s)) = (head(s), exch(tail(tail(s)))) \tag{2}$$

Define $\mathcal{K} = Set^2$ and for all $A, B \in Set$, $R(A)_{list} = (A_{list}, A_{list})$ and $L(A, B)_{list} = A_{list} + B_{list}$.

Let $Q = X^{\mathbb{N}} + X^{\mathbb{N}}$. By (1) and (2), the image of $(exch, exch')^* = [exch, exch'] : Q \to X^{\mathbb{N}}$ is compatible with $head$ and $tail$.

Hence $(exch, exch') : (X^{\mathbb{N}}, X^{\mathbb{N}}) \to (X^{\mathbb{N}}, X^{\mathbb{N}})$ is *Stream*-recursive and thus by Lemma COR, $[exch, exch']$ agrees with $unfold^Q$ where for all $s \in X^{\mathbb{N}}$,
$\langle head^Q, tail^Q\rangle(s, 1) = (head(tail(s)), (s, 2))$ and
$\langle head^Q, tail^Q\rangle(s, 2) = (head(s), (tail(tail(s)), 1))$.

The validity of (1) and (2) is equivalent to the commutativity of (3):

$$\begin{array}{ccc}
X^{\mathbb{N}} & \xrightarrow{\langle head, tail\rangle} & X \times X^{\mathbb{N}} \\
{\scriptstyle [exch, exch']}\uparrow & (3) & \uparrow{\scriptstyle X \times [exch, exch']} \\
Q & \xrightarrow[\langle head^Q, tail^Q\rangle]{} & X \times Q
\end{array}$$

2.12 Corecursion and coproduct: Flatten a cotree

Let $T = \nu coTree(X)$ (see Labelled trees). The functions $flatten : T \to X^\infty$ and $flattenL : T^\infty \to X^\infty$ satisfy the equations

$$split(flatten(t)) = (root(t), flattenL(subtrees(t))) \quad (1)$$

$$split(ts) = * \;\Rightarrow\; split(flattenL(ts)) = * \quad (2)$$

$$split(ts) = (u, us)$$
$$\Rightarrow\; split(flattenL(ts)) = (root(u), flattenL(conc(subtrees(u), us)) \quad (3)$$

where $conc : T^\infty \times T^\infty \to T^\infty$ is defined as in 2.5.

Define $\mathcal{K} = Set^2$ and for all $A, B \in \mathcal{L}$, $R(A)_{list} = (A_{list}, A_{list})$ and $L(A, B)_{list} = A_{list} + B_{list}$.

By (1)-(3), the image of

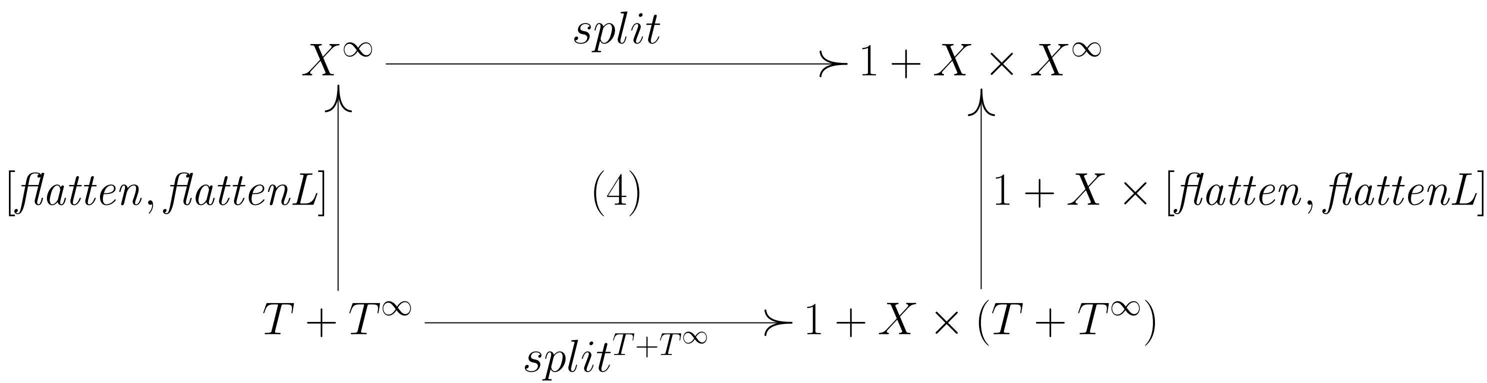
$$(flatten, flattenL)^* = [flatten, flattenL] : T + T^\infty \to X^\infty$$

is compatible with $split$.

Hence $(flatten, flattenL) : (T, T^\infty) \to (X^\infty, X^\infty)$ is $coList(X)$-corecursive and thus by Lemma COR, $[flatten, flattenL]$ agrees with $unfold^{T+T^\infty}$ where for all $t \in T$ and $ts \in T^\infty$,

$$\begin{array}{rcl} split^{T+T^\infty}(t) & = & (root(t), subtrees(t)), \\ split^{T+T^\infty}(ts) & = & \begin{cases} * & \text{if } split(ts) = *, \\ (u, us) & \text{if } split(ts) = (root(u), conc(subtrees(u), us)). \end{cases} \end{array}$$

The validity of (1)-(3) is equivalent to the commutativity of (4):



2.13 Recursion and identity: Subtrees

Let $Z = (\nu coBintree(X) \to \nu coBintree(X))$ (see Destructive signatures). The function

$$subtree : 2^* \to Z$$

satisfies the equations

$$subtree(nil)(t) = t \quad (1)$$
$$fork(t) = (u, e, u') \;\Rightarrow\; subtree(cons(0, s))(t) = subtree(s)(u) \quad (2)$$
$$fork(t) = (u, e, u') \;\Rightarrow\; subtree(cons(1, s))(t) = subtree(s)(u') \quad (3)$$

Define $\mathcal{K} = Set$ and $L = R = Id_{Set}$.

By (1)-(3), the kernel of $subtree$ is compatible with $fork$.

Hence $subtree$ is $List(2)$-recursive and thus by Lemma REC, $subtree$ agrees with $fold^Z$ where for all $s \in 2^*$, $f \in Z$ and $t \in \nu coBintree(X)$,

$$nil^Z = id,$$
$$cons^Z(b, f)(t) = \begin{cases} f(u) & \text{if } b = 0 \text{ and } fork(t) = (u, e, u'), \\ f(u') & \text{if } b = 1 \text{ and } fork(t) = (u, e, u'). \end{cases}$$

The validity of (1)-(3) is equivalent to the commutativity of (4):

$$
\begin{array}{ccc}
1 + 2 \times 2^* & \xrightarrow{[nil, cons]} & 2^* \\
{\scriptstyle 1 + 2 \times subtree} \downarrow & (4) & \downarrow {\scriptstyle subtree} \\
1 + 2 \times Z & \xrightarrow[{[nil^Z, cons^Z]}]{} & Z
\end{array}
$$

Labelled binary trees

Let X be a set.

$$\begin{aligned}
S &= \{btree\},\\
F &= \{empty : 1 \to btree,\ join : btree \times X \times btree \to btree\},\\
F' &= \{split : btree \to 1 + (btree \times X \times btree)\},\\
F'' &= \{root : btree \to X,\ left, right : btree \to btree\},\\
Bintree(X) &= (S, F, \emptyset),\\
coBintree(X) &= (S, F', \emptyset),\\
Infbintree(X) &= (S, F'', \emptyset).
\end{aligned}$$

- For all $A \in Set^S$, $H_{Bintree(X)}(A)_{btree} = H_{coBintree(X)}(A)_{btree} = 1 + A_{btree} \times X \times A_{btree}$ and $H_{Infbintree(X)}(A)_{btree} = A_{btree} \times X \times A_{btree}$.
- $\mu Bintree(X)_{btree} \cong T$ where T is the least set of expressions such that $\bot \in T$ and for all $x \in X$ and $t, u \in T$, $x(t, u) \in T$.
- $empty = \bot$ and for all $x \in X$ and $t, u \in T$, $join(t, x, u) = x(t, u)$.

- $\nu coBintree(X)_{btree} \cong T'$ where T' is the set of partial functions $t : 2^* \longrightarrow X$ such that for all $w \in 2^*$,
  - if $t(w0)$ is defined, then $t(w)$ is defined,
  - if $t(w1)$ is defined, then $t(w0)$ is defined.
- For all $t \in T'$,

$$split(t) = \begin{cases} * & \text{if } t = \Omega, \\ (\lambda w.t(0w), t(\epsilon), \lambda w.t(1w)) & \text{otherwise.} \end{cases}$$

- $\nu Infbintree(X)_{btree} \cong X^{2^*}$.
- For all $t \in X^{2^*}$, $root(t) = t(\epsilon)$, $left(t) = \lambda w.t(0w)$ and $right(t) = \lambda w.t(1w)$.

3.1 Recursion and product: Check balancing (see [21])

Let $T = \mu Bintree(X)_{btree}$. The functions $depth : T \longrightarrow \mathbb{N}$ and $bal : T \longrightarrow 2$ satisfy the equations

$$\langle height, bal\rangle(empty) = (0, True) \tag{1}$$

$$\begin{aligned}\langle height, bal\rangle(join(t, x, u)) = (&max(height(t), height(u)) + 1,\\ &bal(t) \wedge bal(u) \wedge height(t) = height(u))\end{aligned} \tag{2}$$

Define $\mathcal{K} = Set^2$ and for all $A, B \in Set$, $L(A)_{btree} = (\mathbb{N}, 2)$ and $R(A, B)_{btree} = A_{btree} \times B_{btree}$.

By (1) and (2), the kernel of

$$(height, bal)^{\#} = \langle height, bal \rangle : T \to \mathbb{N} \times 2$$

is compatible with *join*.

Hence $(height, bal) : (T, T) \to (\mathbb{N}, 2)$ is $Bintree(X)$-recursive and thus by Lemma REC, $\langle height, bal \rangle$ agrees with $fold^{\mathbb{N}\times 2}$ where

$$\begin{aligned} empty^{\mathbb{N}\times 2} &= (0, True), \\ join^{\mathbb{N}\times 2} &= \lambda((m, b), x, (n, c)).(max(m, n) + 1, b \wedge c \wedge m = n). \end{aligned}$$

The validity of (1) and (2) is equivalent to the commutativity of (3):

$$
\begin{array}{ccc}
1 + T \times X \times T & \xrightarrow{[empty,\, join]} & T \\
\Big\downarrow {\scriptstyle 1 + \langle height,\, bal \rangle} & (3) & \Big\downarrow {\scriptstyle \langle height,\, bal \rangle} \\
1 + (\mathbb{N} \times 2) \times X \times (\mathbb{N} \times 2) & \xrightarrow[{[empty^{\mathbb{N}\times 2},\, join^{\mathbb{N}\times 2}]}]{} & \mathbb{N} \times 2
\end{array}
$$

3.2 Corecursion and identity: Mirror a tree (see [31, 46])

Let $T = \nu coBintree(X)_{btree}$. The function $mirror : T \to T$ satisfies the equations

$$split(t) = * \;\Rightarrow\; split(mirror(t)) = * \quad (1)$$
$$split(t) = (u, x, u') \;\Rightarrow\; split(mirror(t)) = (mirror(u'), x, mirror(u)) \quad (2)$$

Define $\mathcal{K} = Set$ and $R = L = Id_{Set}$.

Extend *mirror* to the constant types X and 1. Then (1) and (2) read as follows:

$$split(t) = * \;\Rightarrow\; split(mirror(t)) = * = mirror(*),$$
$$split(t) = (u, x, u')$$
$$\Rightarrow\; split(mirror(t)) = (mirror(u'), mirror(x), mirror(u)) = mirror(u', x, u),$$

Hence the image of *mirror* is compatible with *split*.

Hence *mirror* is *coBintree*(X)-corecursive and thus by Lemma COR, *mirror* agrees with $unfold^T$ where for all $t \in T$,

$$split^T(t) = \begin{cases} * & \text{if } t = \Omega, \\ (\lambda w.t(1w), t(\epsilon), \lambda w.t(0w)) & \text{otherwise.} \end{cases}$$

The validity of (1) and (2) is equivalent to the commutativity of (3):

$$
\begin{array}{ccc}
T & \xrightarrow{\;split\;} & 1 + T \times X \times T \\
{\scriptstyle mirror}\uparrow & (3) & \uparrow{\scriptstyle 1 + mirror \times X \times mirror} \\
T & \xrightarrow[\;split^T\;]{} & 1 + T \times X \times T
\end{array}
$$

Since T is a final algebra, properties of $mirror^T$ like $mirror^T \circ mirror^T = id_T$ are shown by algebraic coinduction (see, e.g., [46]).

3.3 Destructor extension: Subtrees

In 2.13 have shown that there is a unique interpretation in $\nu coBintree(X)$ of an additional destructor $subtree : 2^* \rightarrow (btree \rightarrow btree)$ such that the corresponding extension of $\nu coBintree(X)$ satisfies the equations (1)-(3) of 2.13.

Let $\Sigma = (S, F' \cup \{subtree' : btree \rightarrow (2^* \rightarrow btree)\}, \{\in: btree\})$,
$\Sigma' = (S, F' \cup \{subtree'\}, \emptyset)$ and AX be a set of Σ-co-Horn clauses such that for all $A \in Alg_{\Sigma,AX}$, $\in^A$ is a Σ-invariant, and AX includes the following co-Horn clauses:

$$\begin{aligned}
\in_{btree}(t) &\Rightarrow subtree'(t)(\epsilon) = t, \\
\in_{btree}(t) &\Rightarrow (split(t) = (u, x, u') \Rightarrow subtree'(t)(0:w) = subtree'(u)(w)), \\
\in_{btree}(t) &\Rightarrow (split(t) = (u, x, u') \Rightarrow subtree'(t)(1:w) = subtree'(u')(w)).
\end{aligned}$$

Let $A = gfp(\Sigma, \nu\Sigma', AX)$. By Theorem RESFIN, $\in^A$ is final in $Alg^{\in}_{\Sigma,AX}$. Since the final $coBintree(X)$-algebra with membership can be extended to a (Σ, AX)-algebra with membership, we conclude from Lemma DESEXT that (Σ, AX) is a conservative extension of $(coBintree(X), \emptyset)$.

3.4 Least Restriction: Finite trees, EF and AF (see [46])

Let $\Sigma = (S, F', \{\mathit{finite}, EF, AF\})$ and AX be a set of Σ-Horn clauses such that for all $A \in Alg_{\Sigma,AX}$, $\in^A$ is a Σ-invariant. Moreover, let AX include the following axioms:

$$\begin{aligned}
\mathit{finite}(t) &\Leftarrow \mathit{split}(t) = * \vee (\mathit{split}(t) = (u, x, u') \wedge \mathit{finite}(u) \wedge \mathit{finite}(u')) \\
EF(P)(t) &\Leftarrow \mathit{split}(t) = (u, x, u') \wedge (P(x) \vee EF(P)(u) \vee EF(u')) \\
AF(P)(t) &\Leftarrow \mathit{split}(t) = (u, x, u') \wedge (P(x) \vee (AF(P)(u) \wedge AF(u')))
\end{aligned}$$

where P is a predicate variable.

Let $A = \mathit{lfp}(\Sigma, \nu\mathit{coBintree}, AX)$. By Theorem RESINI, $\in^A$ is initial in $\mathit{obs}(Alg^{\in}_{\Sigma,AX})$, the category of F'-observable Σ-coalgebras B such that B satisfies AX and $\in^B = B$.

3.5 Greatest Restriction: Infinite trees, AG and EG (see [46])

Let $\Sigma = (S, F', \{\mathit{infinite}, AG, EG\})$ and AX be set of Σ-co-Horn clauses such that for all $A \in Alg_{\Sigma,AX}$, $\in^A$ is a Σ-invariant. Moreover, let AX include the following axioms:

$$\begin{aligned}
\mathit{infinite}(t) &\Rightarrow \exists\, u, x, u' : \mathit{split}(t) = (u, x, u') \wedge (\mathit{infinite}(u) \vee \mathit{infinite}(u')) \\
AG(P)(t) &\Rightarrow \exists\, u, x, u' : (\mathit{split}(t) = (u, x, u') \Rightarrow (P(x) \wedge AG(P)(u) \wedge AG(P)(u'))) \\
EG(P)(t) &\Rightarrow \exists\, u, x, u' : (\mathit{split}(t) = (u, x, u') \Rightarrow (P(x) \wedge AG(P)(u) \wedge AG(P)(u')))
\end{aligned}$$

where P is a predicate variable.

Let $A = \mathit{lfp}(\nu\, \mathit{coBintree}, \Sigma, AX)$. By Theorem RESFIN, $\in^A$ is final in $Alg^{\in}_{\Sigma,AX}$, the category of Σ-algebras B such that B satisfies AX and $\in^B = B$.

Labelled trees (from 4.2 under construction!)

Let X be a set.

$$\begin{aligned}
S &= \{tree, trees\}, \\
F &= \{join : X \times trees \to tree,\ nil : 1 \to trees, \\
&\quad cons : tree\ trees \to trees\}, \\
F' &= \{root : tree \to X,\ subtrees : tree \to trees, \\
&\quad split : trees \to 1 + (tree \times trees)\}, \\
Tree(X) &= (S, F, \emptyset), \\
coTree(X) &= (S, F', \emptyset).
\end{aligned}$$

- For all $A \in Set^S$, $H_{Tree(X)}(A)_{tree} = H_{coTree(X)}(A)_{tree} = X \times A_{trees}$ and $H_{Tree(X)}(A)_{trees} = H_{coTree(X)}(A)_{trees} = 1 + (A_{tree} \times A_{trees})$.
- $\mu Tree(X)_{tree} \cong T$ and $\mu Tree(X)_{trees} \cong T^*$ where T is the least set of expressions such that for all $x \in X$ and $ts \in T^*$, $x \in T$ and $x(ts) \in T$.
- $nil = \epsilon$
  and for all $x \in X$, $t \in T$ and $ts \in T^*$, $join(x, ts) = x(ts)$ and $cons(t, ts) = t : ts$.

- $\nu\, coTree(X)_{tree} \cong T'$ and $\nu\, coTree(X)_{trees} \cong (T')^{\infty}$ where T' is the set of partial functions $t : (\mathbb{N} \cup \{\omega\})^* \to X$ such that for all $w \in (\mathbb{N} \cup \{\omega\})^*$ and $i \in \mathbb{N}$,
  - $t(\epsilon)$ is defined,
  - if $t(w0)$ is defined, then $t(w)$ is defined,
  - if $t(w(i+1))$ is defined, then $t(wi)$ is defined,
  - if $t(w\omega)$ is defined, then for all $i \in \mathbb{N}$, $t(wi)$ is defined.
- For all $t \in T'$, $root(t) = t(\epsilon)$ and

$$subtrees(t) = \begin{cases} * & \text{if } t = \Omega, \\ \lambda i.\lambda w.t(iw) & \text{otherwise.} \end{cases}$$

- For all $ts \in (T')^{\infty}$,

$$split(ts) = \begin{cases} * & \text{if } ts = \epsilon, \\ (ts(0), \lambda i.ts(i+1)) & \text{otherwise.} \end{cases}$$

4.1 Recursion and identity: Flatten a finite tree (see [28])

The functions $\mathit{flatten} : \mu \mathit{Tree}(X)_{tree} \to X^*$ and $\mathit{flattenL} : \mu \mathit{Tree}(X)_{trees} \to X^*$ satisfy the equations

$$\begin{aligned} \mathit{flatten}(\mathit{join}(x, ts)) &= x : \mathit{flattenL}(ts) & (1) \\ \mathit{flattenL}(\mathit{nil}) &= \mathit{nil} & (2) \\ \mathit{flattenL}(\mathit{cons}(t, ts)) &= \mathit{flatten}(t) +\!\!+ \mathit{flattenL}(ts) & (3) \end{aligned}$$

Define $\mathcal{K} = \mathit{Set}$ and $L = R = \mathit{Id}_{\mathit{Set}}$.

Since $S = \{\mathit{tree}, \mathit{trees}\}$, *flatten* and *flattenL* provide the *tree*- resp. *trees*-component of an S-sorted function $\mathit{flatten}' : \mu \mathit{Tree}(X) \to (X^*, X^*)$.

By (1)-(3), the kernel of *flatten* is compatible with *join* and *cons*.

Hence $\mathit{flatten}'$ is $\mathit{Tree}(X)$-recursive and thus by Lemma REC, $\mathit{flatten}'$ agrees with fold^{X^*} where $\mathit{join}^{X^*} = \lambda(x, s).(x : s)$, $\mathit{nil}^{X^*} = \epsilon$ and $\mathit{cons}^{X^*} = \lambda(s, s').(s +\!\!+ s'))$.

The validity of (1)-(3) is equivalent to the commutativity of (4) and (5):

$$
\begin{array}{ccc}
X \times \mu\,Tree(X)_{trees} & \xrightarrow{\;join\;} & \mu\,Tree(X)_{tree} \\
{\scriptstyle X \times \mathit{flattenL}}\downarrow & (4) & \downarrow{\scriptstyle \mathit{flatten}} \\
X \times X^* & \xrightarrow[\;join^{X^*}\;]{} & X^*
\end{array}
$$

$$
\begin{array}{ccc}
1 + (Ltree(X) \times \mu\,Tree(X)_{trees}) & \xrightarrow{\;[nil, cons]\;} & \mu\,Tree(X)_{trees} \\
{\scriptstyle 1 + (\mathit{flatten} \times \mathit{flattenL})}\downarrow & (5) & \downarrow{\scriptstyle \mathit{flattenL}} \\
1 + (X^* \times X^*) & \xrightarrow[\;[nil^{X^*}, cons^{X^*}]\;]{} & X^*
\end{array}
$$

4.2 Least restriction: Cotrees with finite outdegree

Let AX be given by the following Horn clauses over $coTree$:

$$\begin{array}{rcl} \in_{tree}(t) & \Leftarrow & \in_{trees}(subtrees\langle t\rangle) \\ \in_{trees}(ts) & \Leftarrow & [[x,y]split]ts = [x]p \vee \\ & & ([[x,y]split]ts = [y]p \wedge \in_{tree}(\pi_1\langle p\rangle) \wedge \in_{trees}(\pi_2\langle p\rangle)) \end{array}$$

AX satisfies the assumptions of Restriction with a least invariant. Hence $inv = \in^{lfp(\overline{AX})}$ is initial in $obs(Alg_{coTree,AX})$, the category of $coTree$-observable $coTree$-coalgebras A such that A satisfies AX and $\in^A = A$.

4.3 Destructor extension: Flatten a cotree

We have shown that there is a unique interpretation in $\nu coList(X)$ of additional destructors $flatten : tree \rightarrow list$ and $flattenL : trees \rightarrow list$ such that the corresponding extension of $\nu coTree$ satisfies the equations (1)-(3) of 2.12.

Let $coTree' = coTree \cup \{flatten, flattenL\}$. By Lemma DESEXT (1), $coTree'$ is a conservative extension of $coTree$.

Let $C = \{flatten, flattenL\}$. $\nu coTree'$ is isomorphic to the $coTree'$-coalgebra $B =_{def} Tree_{coTree,C}(BA)$ of C-colored $coTree$-trees over BA (see Colored Σ-trees).

B_{tree} can be represented as the set of partial functions

$$t : \mathbb{N}^* \to X \times B_{list}$$

(see 2.3) such that $t(\epsilon)$ is defined and for all $w \in \mathbb{N}^*$ and $i \in \mathbb{N}$,

- if $t(wi)$ is defined, then $t(w)$ is defined,
- if $t(w(i+1))$ is defined, then $t(wi)$ is defined.

B_{trees} can be represented as the union of B_{list} and the set of partial functions

$$ts : \mathbb{N} \to B_{tree} \times B_{list}$$

such that $ts(0)$ is defined and for all $i \in \mathbb{N}$, if $ts(i+1)$ is defined, then $ts(i)$ is defined. With respect to this interpretation, the destructors of $coTree'$ are interpreted as follows:

For all $t \in B_{tree}$ and $ts \in B_{trees}$,

$$
\begin{array}{rcl}
root^B(t) & = & \pi_1(t(\epsilon)), \\
subtrees^B(t) & = & \lambda i.\lambda w.t(iw), \\
flatten^B(t) & = & \pi_2(t(\epsilon)), \\
split^B(ts) & = & \begin{cases} * & \text{if } ts \in B_{list}, \\ (\pi_1(ts(0)), \lambda i.ts(i+1)) & \text{otherwise}, \end{cases} \\
flattenL^B(ts) & = & \begin{cases} ts & \text{if } ts \in B_{list}, \\ \pi_2(ts(0)) & \text{otherwise.} \end{cases}
\end{array}
$$

Let AX be given by the $coTree'$-formulas

$$\in_{tree}(t) \Rightarrow \in_{trees}(subtrees\langle t\rangle) \tag{1}$$

$$\in_{trees}(ts) \Rightarrow \in_{1+tree\times trees}([[y,z]split]ts) \tag{2}$$

$$\in_{tree\times trees}(p) \Rightarrow \in_{tree}(\pi_1\langle p\rangle) \wedge \in_{trees}(\pi_2\langle p\rangle) \tag{3}$$

$$\begin{aligned}\in_{tree}(t) \Rightarrow\ & \exists\, p : ([[y,z]split]flatten\langle t\rangle = [z]p \wedge \pi_1\langle p\rangle = root\langle t\rangle \wedge \\ & \quad \pi_2\langle p\rangle = flattenL\langle subtrees\langle t\rangle\rangle)\end{aligned} \tag{4}$$

$$\begin{aligned}\in_{trees}(ts) \Rightarrow\ & \exists\, p,q : ([[y,z]split]ts = [y]p \wedge [[y,z]split]flattenL\langle ts\rangle = [y]q) \vee \\ & \exists\, p,q : ([[y,z]split]ts = [z]p \wedge [[y,z]split]flattenL\langle ts\rangle = [z]q \wedge \\ & \quad \pi_1\langle q\rangle = root\langle \pi_1\langle p\rangle\rangle \wedge \\ & \quad \pi_2\langle q\rangle = flattenL\langle conc\langle subtrees\langle \pi_1\langle p\rangle\rangle, \pi_2\langle p\rangle\rangle\rangle)\end{aligned} \tag{5}$$

AX consists of inverse Horn clauses over $coTree'$ that satisfy the assumptions of Restriction with a greatest invariant. Hence $gfp(\overline{AX}) = B$. Let $inv = \in^B$.

For all $t, t' \in inv_{tree}$,
$flatten^B(t) \neq flatten^B(t')$ implies $u^B(t) \neq u^B(t')$ for some $u \in Obs_{coTree,tree}$. (6)
For all $ts, ts' \in inv_{trees}$,
$flattenL^B(ts) \neq flattenL^B(ts')$ implies $u^B(ts) \neq u^B(ts')$ for some $u \in Obs_{coTree,trees}$. (7)

Proof.
Since B satisfies (4) and (5), inv satisfies the conclusions of (4) and (5) or, equivalently,

the equations (1)-(3) of 2.12. Hence $t \in inv_{tree}$ iff

$$flatten^B(t) = (root^B(t), flattenL^B(subtrees^B(t))), \qquad (8)$$

and $ts \in inv_{trees}$ iff for all $u \in B_{tree}$ and $us \in B_{trees}$,

$$split^B(ts) = * \ \text{ implies } \ split^B(flattenL^B(ts)) = *, \qquad (9)$$

$$split^B(ts) = (u, us)$$
$$\text{implies } \ flattenL^B(ts) = (root^B(u), flattenL^B(conc^B(subtrees^B(u), us))). \qquad (10)$$

It is easy to see that

- $Obs_{coTree,tree} = \{obs_w \mid w \in \mathbb{N}^*\}$ where $obs_\epsilon = \{[0]root\}$ and for all $w \in \mathbb{N}^+$, $obs_w = [0 \cdot obsL_w]subtrees$,
- $Obs_{coTree,trees} = \{obsL_w \mid w \in \mathbb{N}^+\}$ where for all $i > 0$ and $w \in \mathbb{N}^*$, $obsL_{0w} = [0, [10 \cdot obs^B_w]\pi_1]split$ and $obsL_{iw} = [0, [10 \cdot obsL_{(i-1)w}]\pi_2]split$,
- for all $t \in B_{tree}$ and $w \in \mathbb{N}^*$, $obs^B_w(t) = t(w)$ if $t(w)$ is defined, and $obs^B_w(t) = *$ otherwise, (11)
- for all $ts \in B_{trees}$, $i \in \mathbb{N}$ and $w \in \mathbb{N}^+$, $obsL_{iw}(ts) = ts(i)(w)$ if $ts(i)(w)$ is defined, and $obsL_{iw}(ts) = *$ otherwise. (12)

By (8)-(10) and the definition of B, for all $t \in inv_{tree}$, $ts \in inv_{trees}$ and $s \in B_{list}$,

$$flatten^B(t) = s \;\Leftrightarrow\; \forall\, n \in domain(s) : t(leafPos(t)(n)) = s(n),$$
$$flattenL^B(ts) = s \;\Leftrightarrow\; \forall\, n \in domain(s) : ts(i)(w) = s(n) \text{ where } leafPosL(ts)(n) = iw,$$

and thus by (11) and (12),

$$flatten^B(t) = s \;\Leftrightarrow\; \forall\, n \in domain(s) : obs^B_{leafPos(t)(n)}(t) = s(n), \quad (13)$$
$$flattenL^B(ts) = s \;\Leftrightarrow\; \forall\, n \in domain(s) : obsL^B_{leafPosL(ts)(n)}(ts) = s(n), \quad (14)$$

where $leafPos(t)(n)$ and $leafPosL(ts)(n)$ are the positions of the n-th leaf of t and ts, respectively.

Haskell code for $leafPos : B_{tree} \to \mathbb{N} \to \mathbb{N}^*$ and $leafPosL : B_{trees} \to \mathbb{N} \to \mathbb{N}^+$:

```
leafPos  = (!!) . leafPoss
leafPosL = (!!) . leafPossL

leafPoss :: B_tree -> [[Int]]
leafPoss t = if null ts then [[]] else leafPossL ts
             where ts = subtrees t

leafPossL :: B_trees -> [[Int]]
leafPossL ts = if null ts then [] else concatMap g [0..length ts-1]
               where g i = map (i:) $ leafPoss $ ts!!i
```

Let $t, t' \in B_{tree}$ and $s, s' \in B_{list}$ such that $\mathit{flatten}^B(t) = s \neq s' = \mathit{flatten}^B(t')$. Let $domain(t) \neq domain(t')$. Then there is $w \in \mathbb{N}^*$ such that $t(w)$ is defined and $t'(w)$ is undefined. Hence by (11), $obs_w^B(t) = t(w)$ and $obs_w^B(t') = *$, and thus (6) is valid for $u = obs_w$. Let $domain(t) = domain(t')$. Then $domain(s) = domain(s')$ and there is $n \in domain(s)$ such that $s(n) \neq s'(n)$ and for all $i < n$, $s(i) = s'(i)$. By (13),

$$obs^B_{leafPos(t)(n)}(t) = s(n) \neq s'(n) = obs^B_{leafPos(t')(n)}(t') = obs^B_{leafPos(t)(n)}(t').$$

Hence (6) is valid for $u = obs_{leafPos(t)(n)}$.

Let $ts, ts' \in B_{trees}$ and $s, s' \in B_{list}$ such that $\mathit{flattenL}^B(ts) = s \neq s' = \mathit{flattenL}^B(ts')$. Let $domain(ts) \neq domain(ts')$ or $domain(ts(i)) \neq domain(ts'(i))$ for some $i \in domain(ts) = domain(ts')$. Then there are $i \in \mathbb{N}$ and $w \in \mathbb{N}^*$ such that $ts(i)(w)$ is defined and $ts'(i)(w)$ is undefined. Hence by (12), $obsL^B_{iw}(ts) = ts(i)(w)$ and $obsL^B_{iw}(ts') = *$, and thus (7) is valid for $t = obs_{iw}$. Let $domain(ts) = domain(ts')$ and for all $i \in domain(ts)$, $domain(ts(i)) = domain(ts'(i))$. Then $domain(s) = domain(s')$ and there is $n \in domain(s)$ such that $s(n) \neq s'(n)$. By (14),

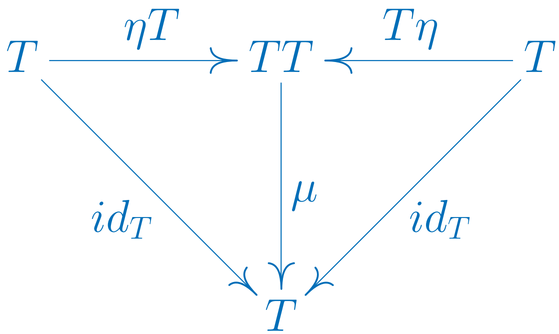
$$obs^B_{leafPosL(ts)(n)}(ts) = s(n) \neq s'(n) = obs^B_{leafPosL(ts')(n)}(ts') = obs^B_{leafPos(ts)(n)}(ts').$$

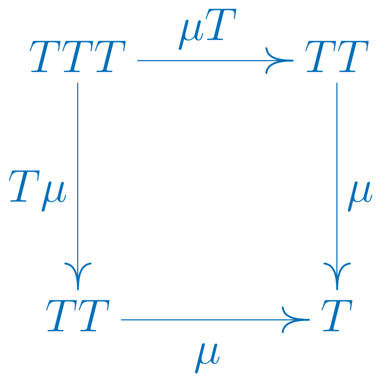
Hence (7) is valid for $u = obs_{leafPosL(ts)(n)}$. ❑

Let $\in^A = \nu\, coTree$. Then A satisfies AX. Hence $A \in Alg^{\in}_{coTree',AX}$ and thus by Lemma DESEXT (2), (6) and (7) imply $\in^B|_{coTree} \cong \nu\, coTree$.

Monads and comonads

A **monad** (or **algebraic theory in monoid form**) in $\mathcal{K}$ is a triple $M = (T, \eta, \mu)$ consisting of a functor $T : \mathcal{K} \to \mathcal{K}$ and natural transformations $\eta : Id_{\mathcal{K}} \to T$ (**unit**) and $\mu : TT \to T$ (**multiplication**) such that the following diagrams commute:





Let $A, B \in \mathcal{K}$. For all $f : A \to B$, the extension $f^* : T(A) \to B$ is defined as $\mu_B \circ T(f)$. Conversely, $\mu = id^*_{T(A)}$.

A monad in $\mathcal{K}$ is a monoid in the category $\mathcal{K}^{\mathcal{K}}$ with functors as objects and natural transformations as morphisms.

In Haskell, M is defined in terms of *return* $= \eta$ and *bind* $: T(A) \to (A \to T(B)) \to T(B)$. (also denoted by `>>=`): For all $t \in T(A)$ and $f : A \to T(B)$,

$$bind(t)(f) = \mu_B(T(f)(t)) = f^*(t).$$

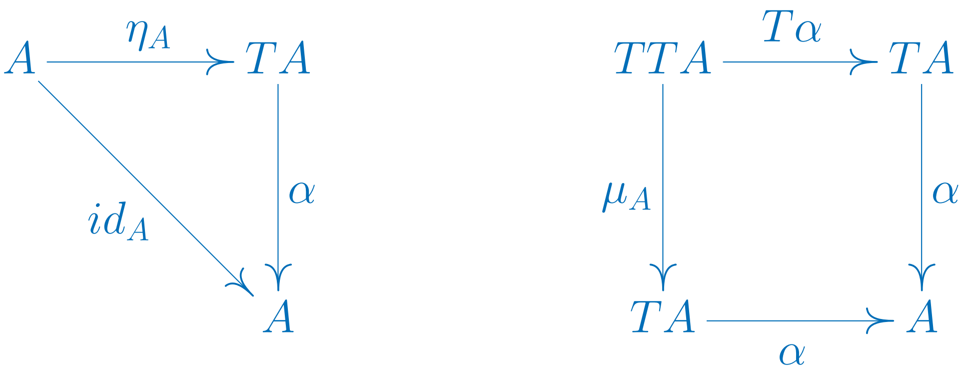
Conversely, $\mu(t) = id^*_{T(A)}(t) = bind(t)(id_{T(A)})$. μ is called *join* in Haskell.

Example

The **list monad** is given by $\mathcal{LM} = (T, \eta, \mu)$ is defined as follows: For all $A \in Set$,

$$T(A) = A^*, \quad \eta_A = \lambda a.[a] : A \to T(A) \quad \mu_A = concat : T(T(A)) \to T(A).$$ ❏

An M**-algebra** or **Eilenberg-Moore algebra** is a T-algebra $\alpha : TA \to A$ such that the following diagrams commute:



The category of M-algebras is denoted by Alg_M. Alg_M is a full subcategory of Alg_T.

Let $\mathcal{A} = (L : \mathcal{K} \to \mathcal{L}, R : \mathcal{L} \to \mathcal{K}, \eta : Id_{\mathcal{K}} \to RL, \epsilon : LR \to Id_{\mathcal{K}})$ be an adjunction.

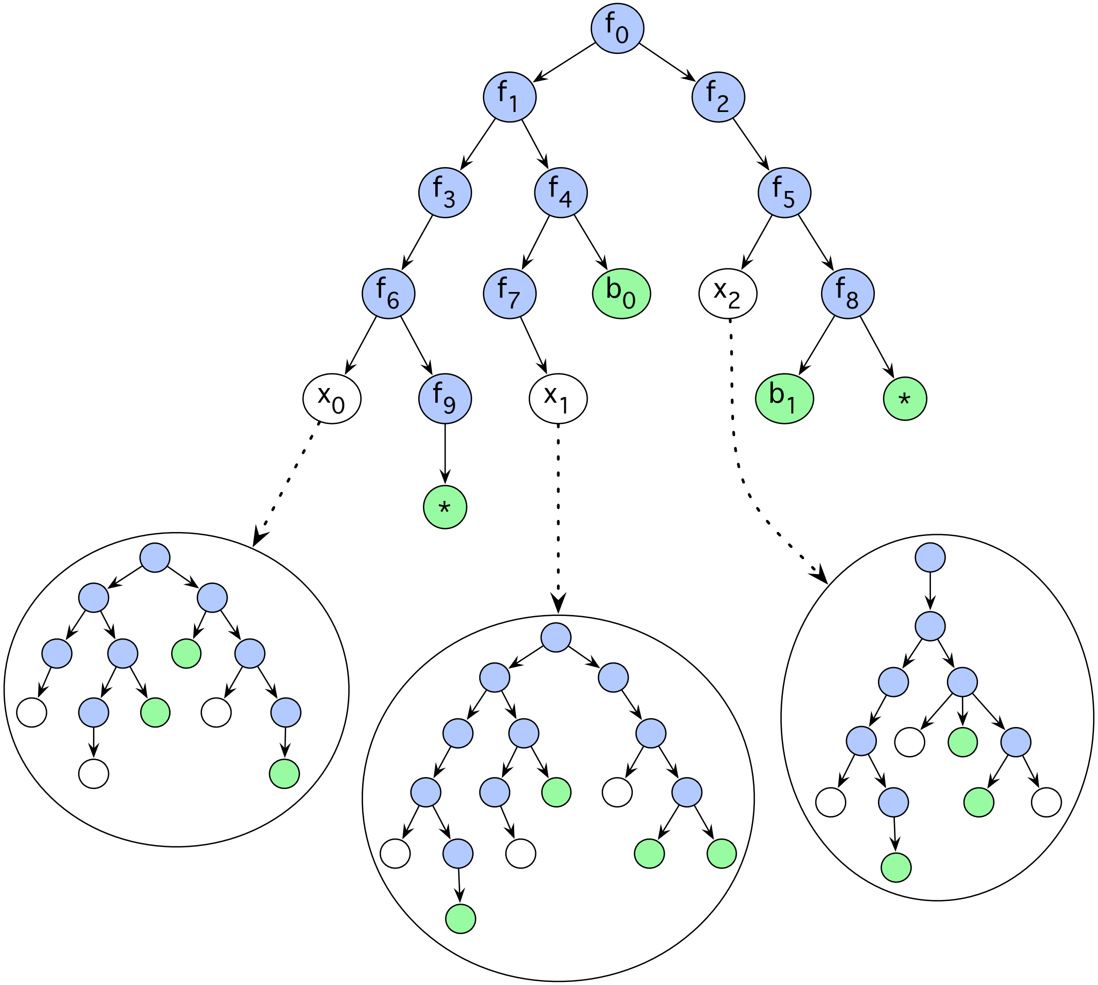
$M(\mathcal{A}) = (RL, \eta, R\epsilon L : RLRL \to RL)$ is a monad, called **the monad induced by** $\mathcal{A}$.

Let $\Sigma = (S, F, P)$ be a (flat) constructive signature.

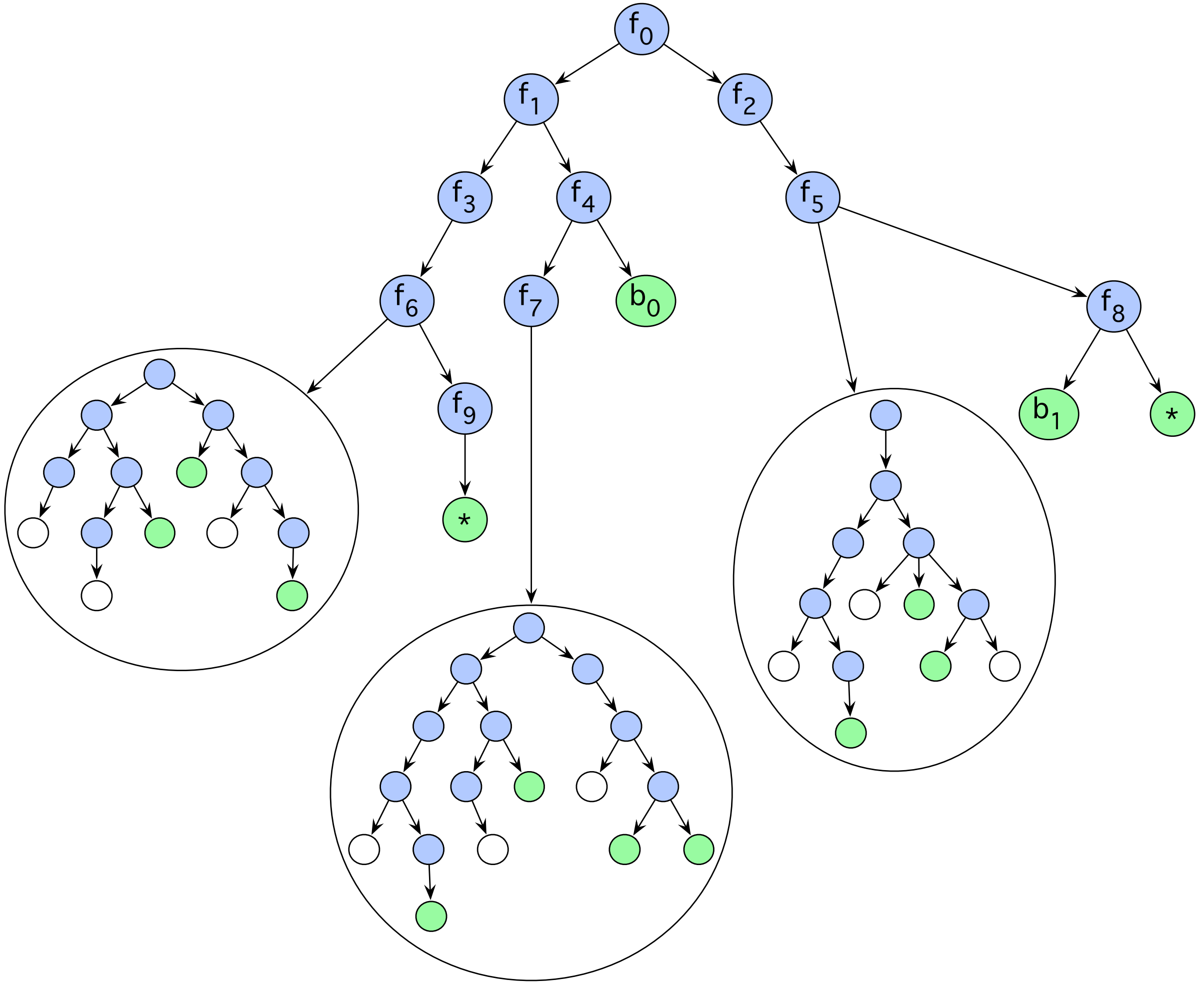
The monad induced by the adjunction $\mathcal{A}_\Sigma = (T_\Sigma, U_S, \eta, \epsilon)$ is called the **monad freely generated by** Σ (see Term adjunction).

The multiplication $\mu : U_S T_\Sigma U_S T_\Sigma \to U_S T_\Sigma$ of the monad freely generated by Σ is defined as follows: For all sets X and trees $t \in T_\Sigma(T_\Sigma(X))$, $\mu_X(t)$ is the tree in $T_\Sigma(X)$ that is obtained from t by substituting each leaf n of t with the label of n (which is in $T_\Sigma(X)$).

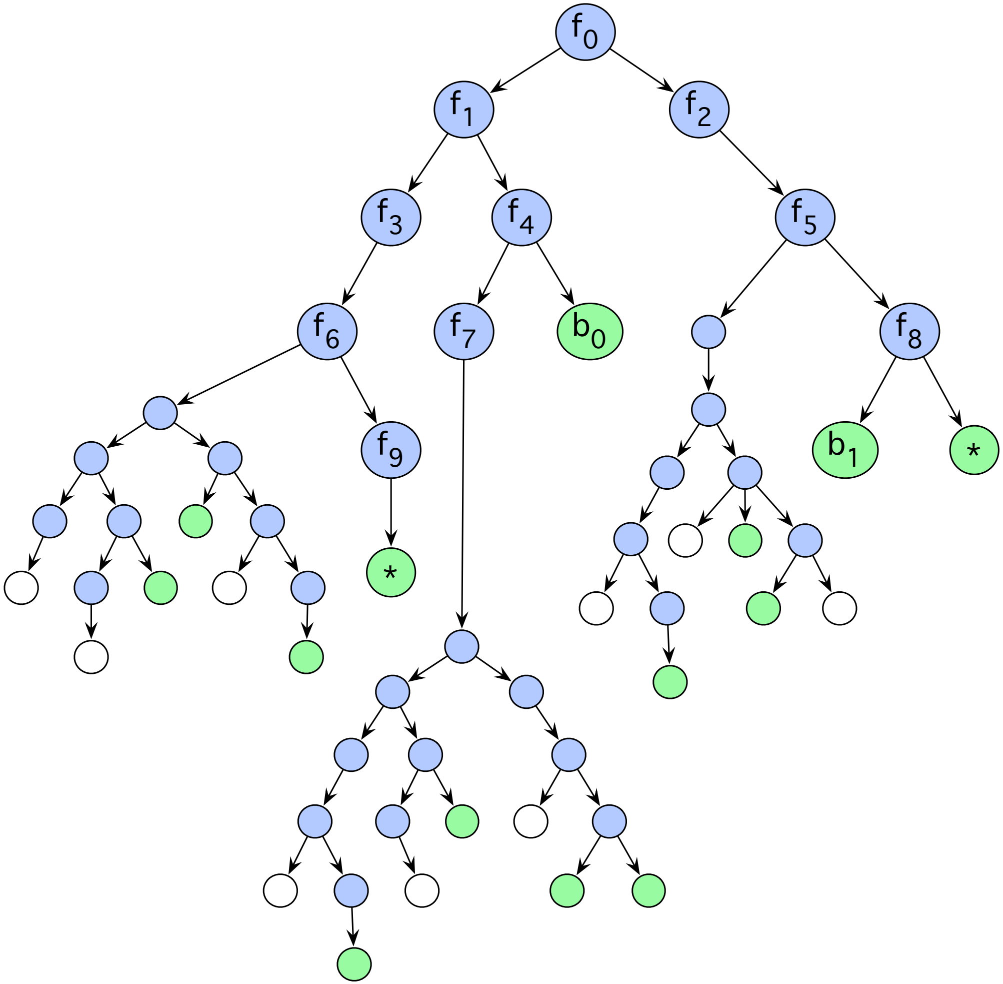
The categories $Alg_{M(\mathcal{A}_\Sigma)}$ and Alg_Σ are isomorphic.



A Σ-term t over X together with a valuation $g : X \to T_\Sigma(Y)$



The term u over $T_\Sigma(Y)$ that results from applying $T_\Sigma(g) : T_\Sigma(X) \to T_\Sigma(T_\Sigma(Y))$ to t



The term over X that results from applying $\mu_Y : T_\Sigma(T_\Sigma(Y)) \rightarrow T_\Sigma(Y)$ *to* u

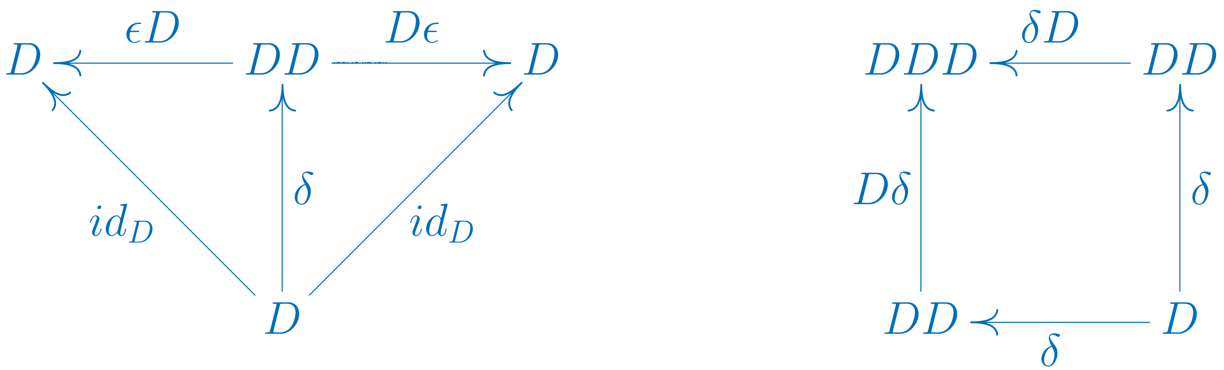
Let $M = (T : \mathcal{K} \to \mathcal{K}, \eta, \mu)$ be a monad.

The forgetful functor $U_M : Alg_M \to \mathcal{K}$ has a left adjoint $F_M : \mathcal{K} \to Alg_M$.

Let $\mathcal{A}_M = (U_M, F_M, \eta, \epsilon)$ be the corresponding adjunction.

The monad induced by $\mathcal{A}_M$ coincides with M: $M(\mathcal{A}_M) = M$.

A **comonad** in $\mathcal{K}$ is a triple $CM = (D, \epsilon, \delta)$ consisting of a functor $D : \mathcal{K} \to \mathcal{K}$ and natural transformations $\epsilon : D \to Id_{\mathcal{K}}$ (**counit**) and $\delta : D \to DD$ (**comultiplication**) such that the following diagrams commute:



Let $A, B \in \mathcal{K}$. For all $g : A \to B$, the extension $g^{\#} : A \to D(B)$ is defined as $D(g) \circ \delta_A$. Conversely, $\delta = id^{\#}_{D(A)}$.

In Haskell, CM is defined in terms of $retract = \epsilon$ and

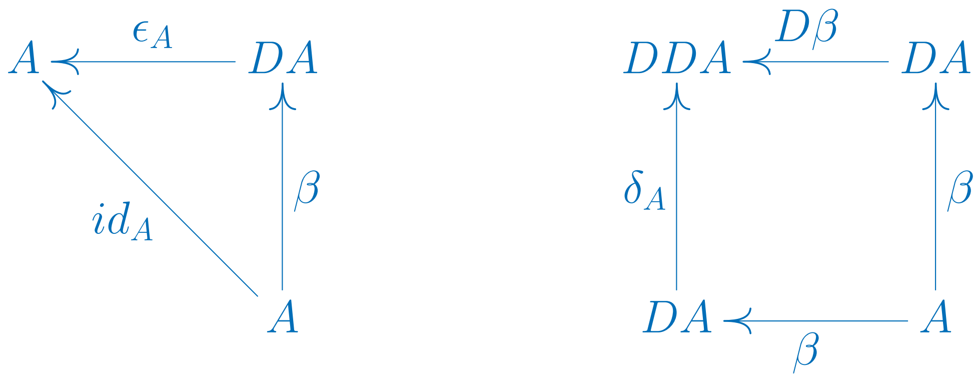
$$cobind : D(A) \to (D(A) \to B) \to D(B)$$

(also denoted by `=>>`): For all $d \in D(A)$ and $g : D(A) \to B$,

$$cobind(d)(g) = D(g)(\delta_A(d)) = g^{\#}(d).$$

Conversely, $\delta(d) = id^{\#}_{D(A)}(d) = cobind(d)(id_{D(A)})$.

A CM-**coalgebra** is a D-coalgebra $\beta : A \to DA$ such that the following diagrams commute:



The category of CM-coalgebras is denoted by $coAlg_{CM}$. $coAlg_{CM}$ is a full subcategory of $coAlg_D$.

Let $\mathcal{A} = (L : \mathcal{K} \to \mathcal{L}, R : \mathcal{L} \to \mathcal{K}, \eta : Id_{\mathcal{K}} \to RL, \epsilon : LR \to Id_{\mathcal{K}})$ be an adjunction.

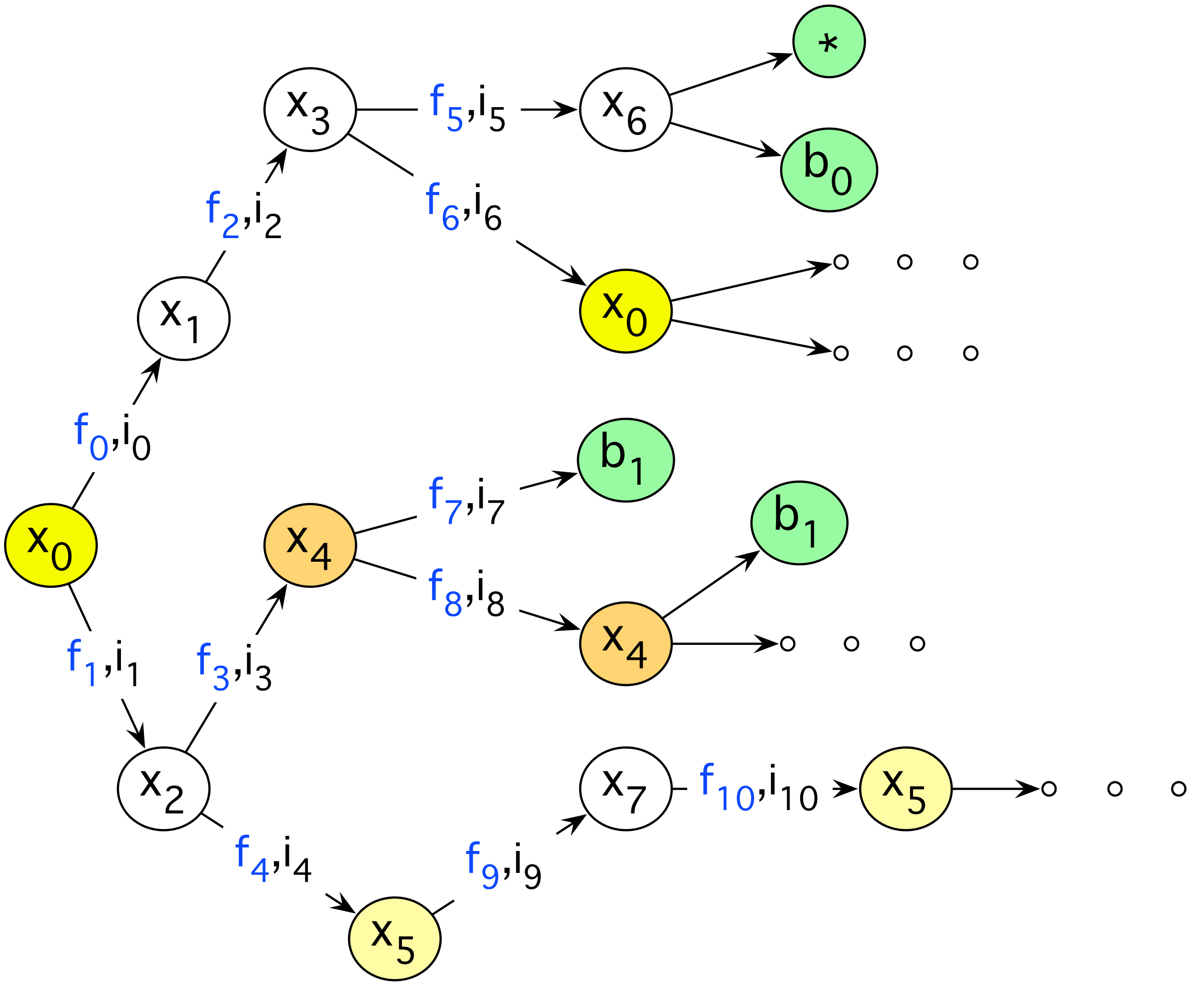
$CM(\mathcal{A}) = (LR, \epsilon, L\eta R : LR \to LRLR)$ is a comonad, called **the comonad induced by** $\mathcal{A}$.

Let $\Sigma = (S, F, P)$ be a (flat) destructive signature.

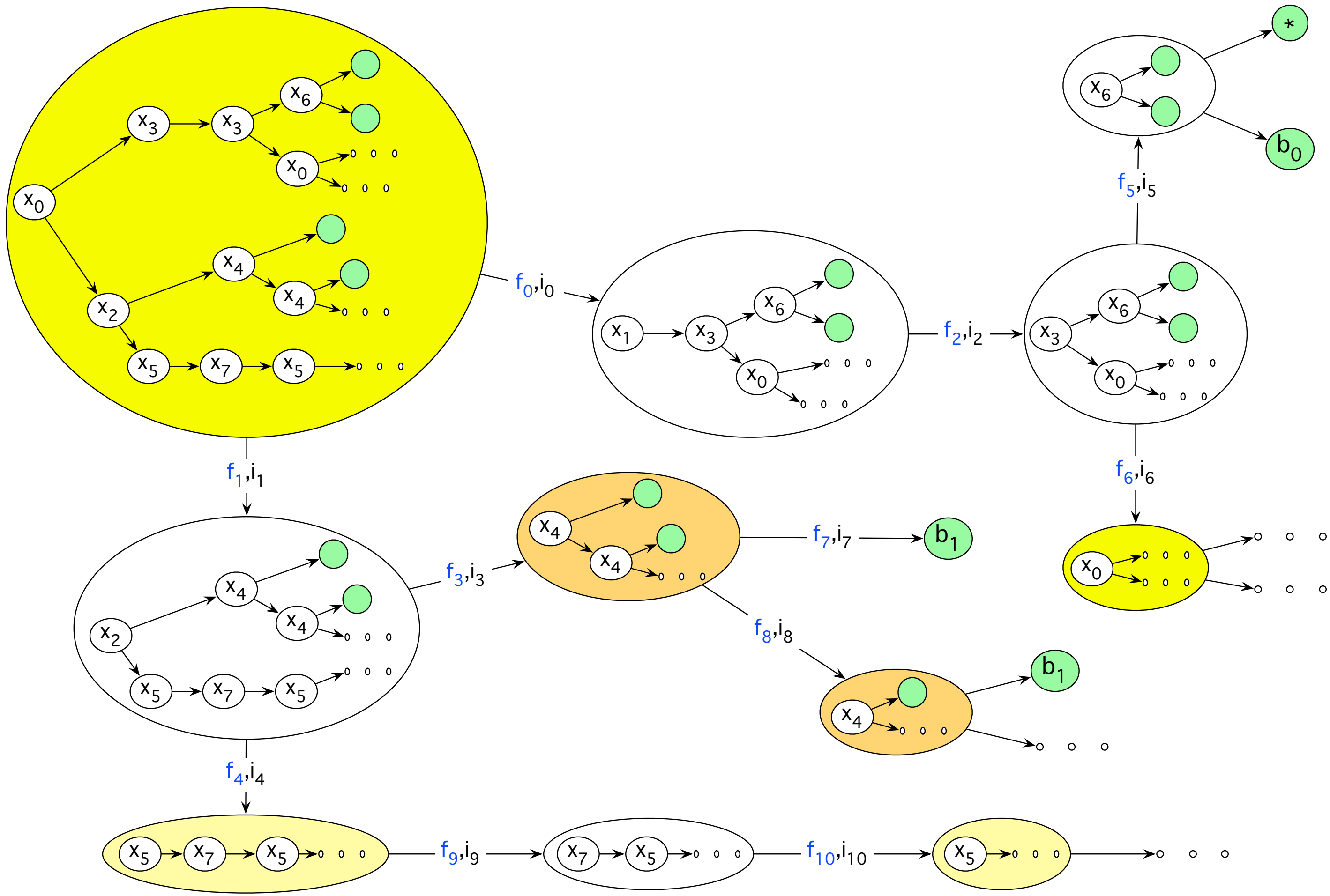
The comonad induced by the adjunction $\mathcal{A}_\Sigma = (U_S, coT_\Sigma, \eta, \epsilon)$ is called the **comonad cofreely generated by** Σ (see Coterm adjunction).

The comultiplication $\delta : U_S coT_\Sigma \to U_S coT_\Sigma U_S coT_\Sigma$ of the comonad cofreely generated by Σ is defined as follows: For all sets X and trees $t \in coT_\Sigma(X)$, $\delta_X(t)$ is the tree in $coT_\Sigma(coT_\Sigma(X))$ that is obtained from t by replacing the label of each node n of t with the subtree of t whose root is n.

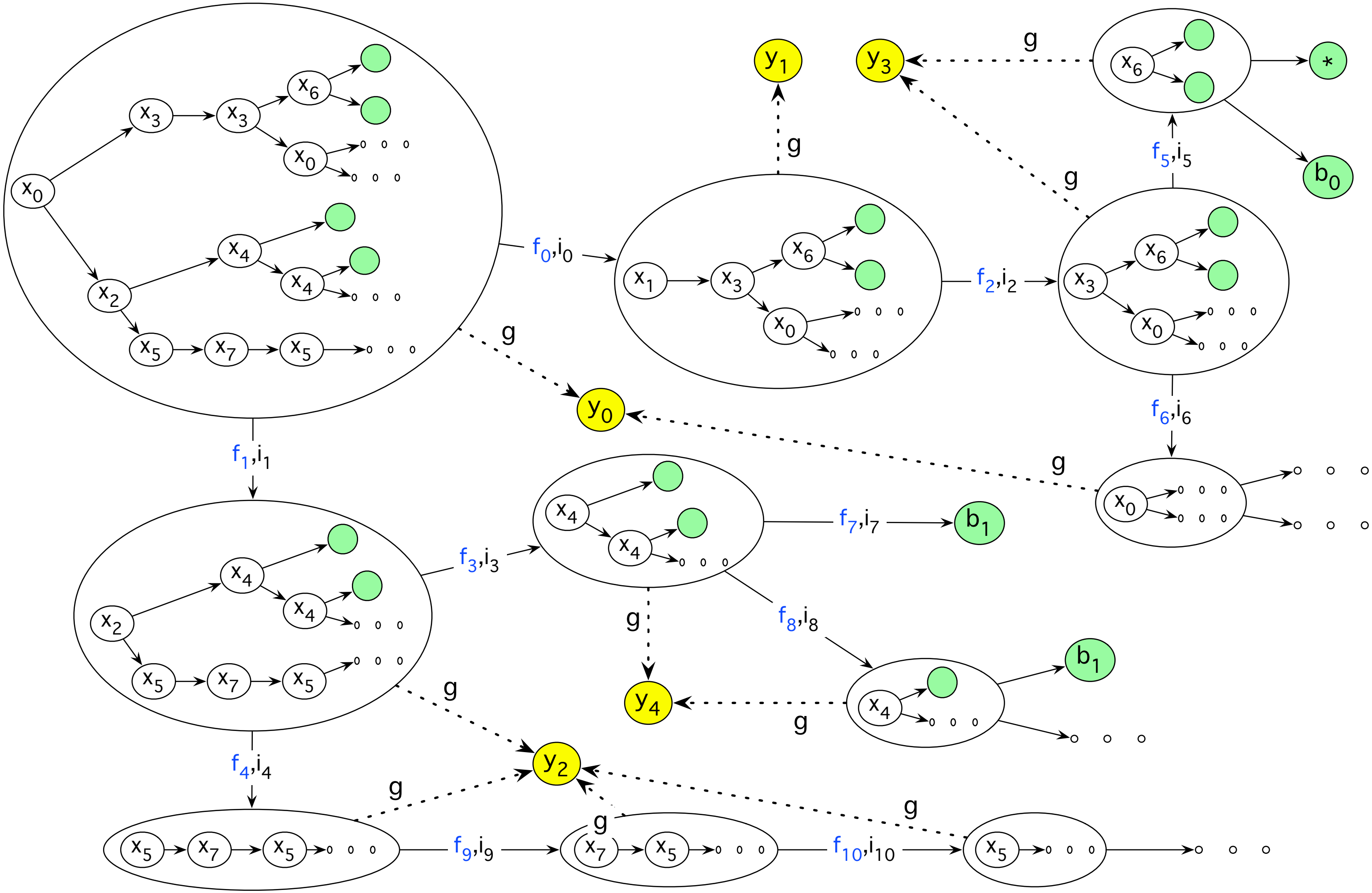
The categories $coAlg_{CM(\mathcal{A}_\Sigma)}$ and $coAlg_\Sigma$ are isomorphic.



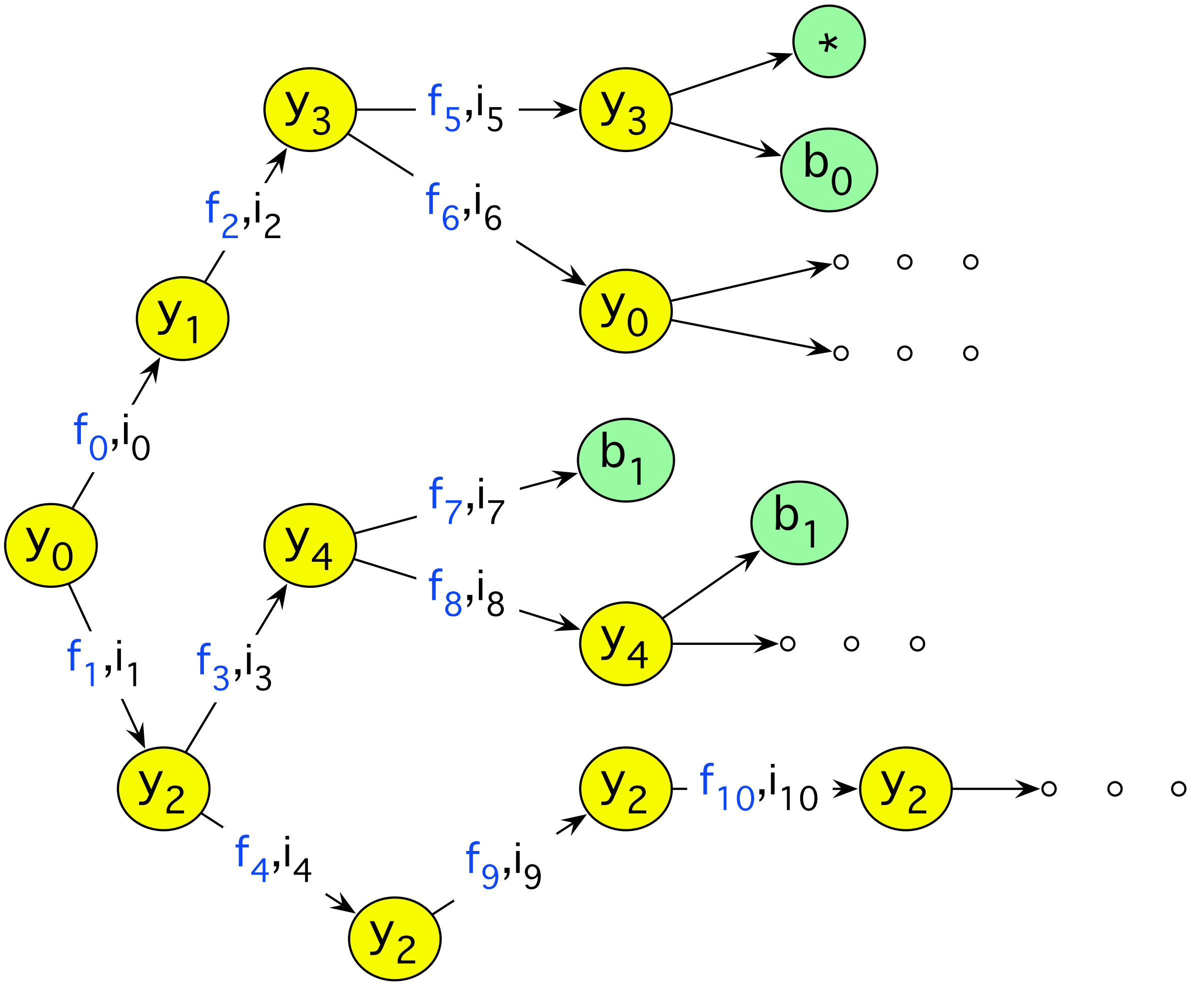
A Σ*-coterm* t *over* X


x0
x3
x3
x6
x0
x2
x4
x4
x5
x7
x5
f0,i0
x1
x3
x6
x0
f2,i2
x3
x6
x0
f5,i5
x6
*
b0
f1,i1
f6,i6
x0
x2
x4
x4
x5
x7
x5
f3,i3
x4
x4
f7,i7
b1
f8,i8
x4
b1
f4,i4
x5
x7
x5
f9,i9
x7
x5
f10,i10
x5

$$\textit{The coterm } u \textit{ over } coT_\Sigma(X)$$
$$\textit{that results from applying } \delta_X : coT_\Sigma(X) \to coT_\Sigma(coT_\Sigma(X)) \textit{ to } t$$


x0
x3
x3
x6
x0
x2
x4
x4
x5
x7
x5
f0,i0
x1
x3
x6
x0
f2,i2
x3
x6
x0
f5,i5
x6
*
b0
y1
y3
g
g
g
g
y0
f6,i6
x0
f1,i1
x2
x4
x4
x5
x7
x5
f3,i3
x4
x4
f7,i7
b1
f8,i8
x4
b1
y4
g
g
y2
f4,i4
x5
x7
x5
f9,i9
x7
x5
f10,i10
x5
g
g
g
g

The coterm u over $coT_\Sigma(X)$ together with a coloring $g : coT_\Sigma(X) \longrightarrow Y$



The coterm over Y *that results from applying* $coT_\Sigma(g) : coT_\Sigma(coT_\Sigma(X)) \to coT_\Sigma(Y)$ *to u*

Let $CM = (D : \mathcal{K} \to \mathcal{K}, \epsilon, \delta)$ be a comonad.

The forgetful functor $U_{CM} : coAlg_{CM} \to \mathcal{K}$ has a right adjoint $C_{CM} : \mathcal{K} \to coAlg_{CM}$.

Let $\mathcal{A}_{CM} = (U_{CM}, F_{CM}, \eta, \epsilon)$ be the corresponding adjunction.

The comonad induced by $\mathcal{A}_{CM}$ coincides with CM: $CM(\mathcal{A}_{CM}) = CM$.

Distributive laws and bialgebras

Given two functors $T, D : \mathcal{K} \to \mathcal{K}$, a **distributive law** is a natural transformation $\lambda : TD \to DT$.

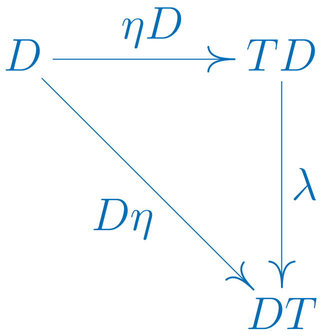
Given a distributive law $\lambda : TD \to DT$, a $\mathcal{K}$-morphism $TA \xrightarrow{\alpha} A \xrightarrow{\beta} DA$ is a λ-**bialgebra** if the following diagram commutes:

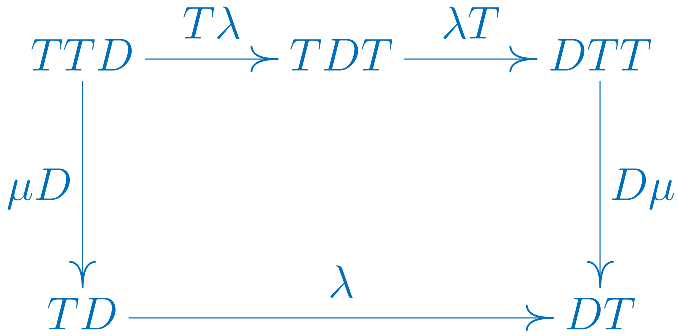
$$\begin{array}{ccccc} TA & \xrightarrow{\quad\alpha\quad} & A & \xrightarrow{\quad\beta\quad} & DA \\ {\scriptstyle T\beta}\big\downarrow & & & & \big\uparrow{\scriptstyle D\alpha} \\ TDA & & \xrightarrow{\qquad\lambda_A\qquad} & & DTA \end{array} \qquad \Longrightarrow \qquad \begin{cases} \alpha \in Mor(coAlg_D) \text{ maps } \lambda_A \circ T\beta \text{ to } \beta, \\ \beta \in Mor(Alg_T) \text{ maps } \alpha \text{ to } D\alpha \circ \lambda_A. \end{cases}$$

Conversely,

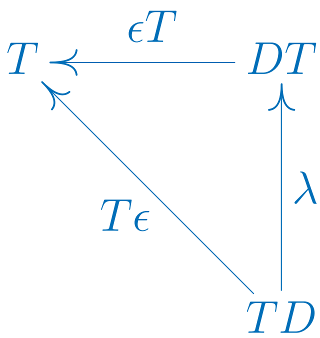
- if $TA \xrightarrow{\alpha} A$ is the initial T-algebra, then there is a unique Alg_T-morphism β from α to $D\alpha \circ \lambda_A$ and thus $TA \xrightarrow{\alpha} A \xrightarrow{\beta} DA$ is a(n initial) λ-bialgebra,
- if $A \xrightarrow{\beta} DA$ is the final D-coalgebra, then there is a unique $coAlg_D$-morphism α from $\lambda_A \circ T\beta$ to β and thus $TA \xrightarrow{\alpha} A \xrightarrow{\beta} DA$ is a (final) λ-bialgebra.

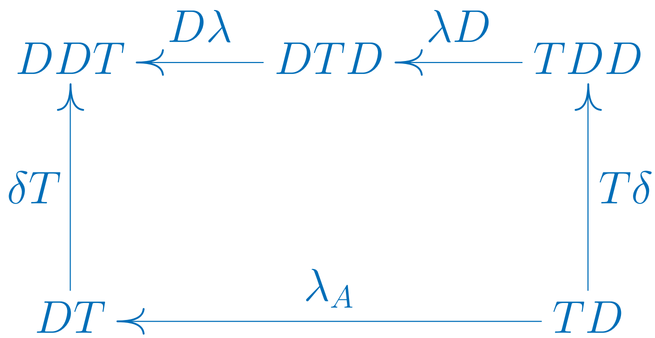
Given a monad $M = (T, \eta, \mu)$, a distributive law $\lambda : TD \to DT$ is M-**compatible** if the following diagrams commute:





Given a comonad $CM = (D, \epsilon, \delta)$, a distributive law $\lambda : TD \to DT$ is CM-**compatible** if the following diagrams commute:





Examples

Given a monad $M = (T, \eta, \mu)$ in Set, the strength $st^{T,A}$ of T and A is M-compatible.

Given a monoid A with multiplication $\cdot$ and unit e,

$$CM = ((-)^A, \epsilon, \delta)$$

with $\epsilon_B(f) = f(e)$ and $\delta_B(f) = \lambda a.\lambda b.f(a \cdot b)$ for all sets B and $f \in B^A$ is a comonad and $st^{T,A}$ is CM-compatible.

Given a T-algebra $\alpha : TB \to B$, let $D = (-)^A \times B$.

$$\lambda : TD \to DT$$

with

$$\lambda_X : TDX = T(X^A \times B) \xrightarrow{\langle T(\pi_1), T(\pi_2)\rangle} T(X^A) \times TB \xrightarrow{st_X^{T,A} \times \alpha} (TX)^A \times B = DTX$$

is an M-compatible distributive law. ❑

A previous notion of coterms

Let $w \in \mathbb{N}^*$.

- For all $x \in X_s$,

$$x(w) =_{def} \begin{cases} x & \text{if } w = \epsilon, \\ \mathit{undefined} & \text{otherwise.} \end{cases}$$

- For all $f : s_1 \dots s_n \to s \in F$ and $t_i \in T_\Sigma(X)_{s_i}$, $1 \le i \le n$,

$$f\langle t_1, \dots, t_n \rangle(w) =_{def} \begin{cases} f & \text{if } w = \epsilon, \\ t_{i+1}(v) & \text{if } w = iv \text{ for some } i \in \mathbb{N},\ v \in \mathbb{N}^*, \\ \mathit{undefined} & \text{otherwise.} \end{cases}$$

- For all $f : s \to s_1 \dots s_n \in F$ and $t_i \in coT_\Sigma(X)_{s_i}$, $1 \le i \le n$,

$$[t_1, \dots, t_n]f(w) =_{def} \begin{cases} f & \text{if } w = \epsilon, \\ t_{i+1}(v) & \text{if } w = iv \text{ for some } i \in \mathbb{N},\ v \in \mathbb{N}^*, \\ \mathit{undefined} & \text{otherwise.} \end{cases}$$

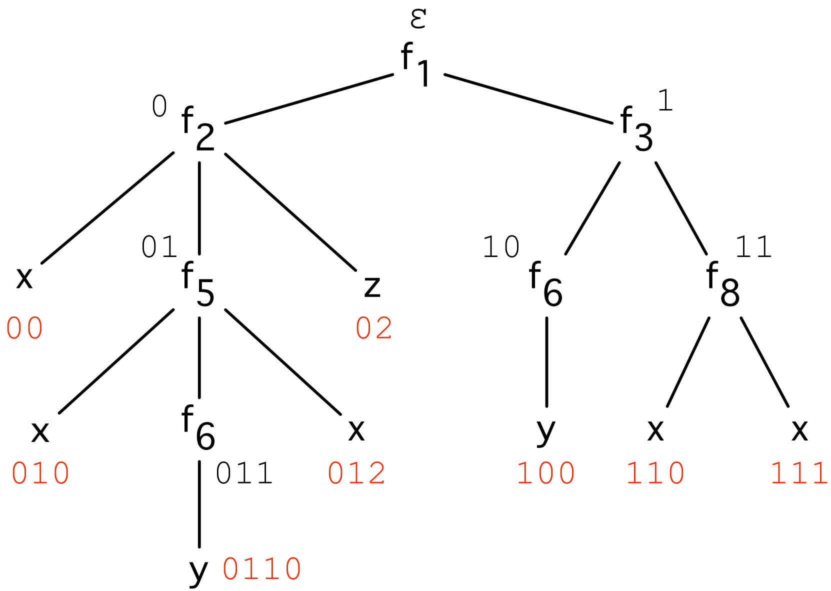
Given a coterm t and $w \in \mathbb{N}^*$, $path(t, w)$ returns the sequence of symbols on the path from the root to node w of t: For all $x \in X$, $[t_1,\dots,t_n]f \in coT_\Sigma(X)$, $i \in \mathbb{N}$ and $w \in \mathbb{N}^*$,

$$path(x,w) =_{def} \begin{cases} x & \text{if } w = \epsilon, \\ undefined & \text{otherwise,} \end{cases}$$

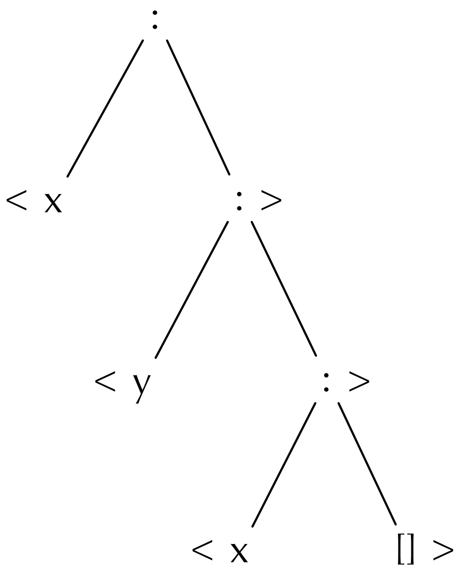
$$path([t_1,\dots,t_n]f, iw) =_{def} \begin{cases} f\ path(t_{i+1}, w) & \text{if } 0 \leq i < n, \\ undefined & \text{otherwise.} \end{cases}$$

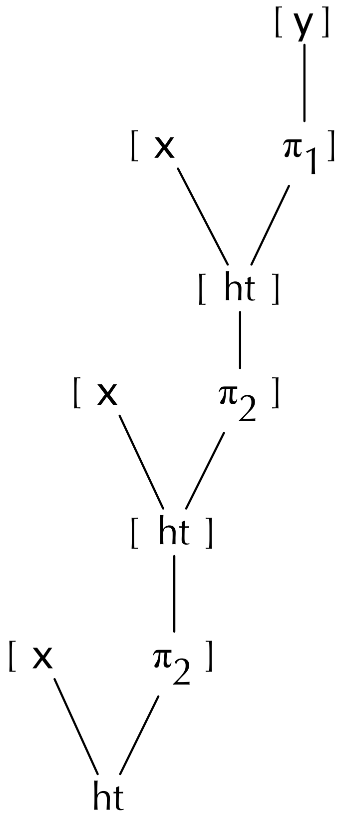
A term resp. coterm t over $\mathbb{N}^*$ such that all function symbols of t belong to $F \setminus BF$ and for all $x \in var(t) \cup cov(t)$, $sort(x) \in BS$ and $t(x) = x$, is called a **Σ-generator** resp. **Σ-observer**.

Given $w \in \mathbb{N}^*$ and a co/term t, $w \cdot t$ denotes the co/term obtained from t by replacing each co/variable v of t with wv.



The tree representing the term $f_1\langle f_2\langle x, f_5\langle x, f_6\langle y\rangle, x\rangle, z\rangle, f_3\langle f_6\langle y\rangle, f_8\langle x, x\rangle\rangle\rangle$
or the coterm $[[[x,[x,[y]f_6,x]f_5,z]f_2,[[y]f_6,[x,x]f_8]f_3]f_1$





The term $: \langle x : \langle y : \langle x, [] \rangle \rangle \rangle$ generates lists of length 3 from two elements.

If applied to a list with at least three elements, the coterm $[x, [[x, [[x, [y]\pi_1]ht]\pi_2]ht]\pi_2]ht$ returns the third element at exit y. If the list has fewer elements, the coterm returns this fact by taking exit x. The underlying signatures are given later.

The S-sorted set $coT_\Sigma(Y)$ of **Σ-coterms over X** is inductively defined as follows:

- For all $s \in S$, $Y_s \subseteq coT_\Sigma(Y)_s$.
- For all $f : s \to s_1 \dots s_n \in F$ and $t_i \in coT_\Sigma(Y)_{s_i}$, $1 \leq i \leq n$, $[t_1, \dots, t_n]f \in coT_\Sigma(Y)_s$.

$[t_1, \dots, t_n]f$ is also written as $[t_i]_{i=1}^n f$.

A Σ-term t is a **ground term** if $var(t)$ is empty.

Given $t \in T_\Sigma(V)$, $var(t)$ denotes the set of variables occurring in t.

Given $t \in coT_\Sigma(Y)$, $cov(t)$ denotes the set of covariables occurring in t.

Let $\Sigma = (S, F, P)$ be a signature, V be a $\mathbb{T}(S, BS)$-sorted set of variables and A be a Σ-algebra.

The $\mathbb{T}(S, BS)$-sorted function

$$\_^A \quad = \quad \{\_^A : T_\Sigma(V)_e \to (A^V \to A_e) \mid e \in \mathbb{T}(S, BS)\}$$

is inductively defined as follows: Let $g \in A^V$.

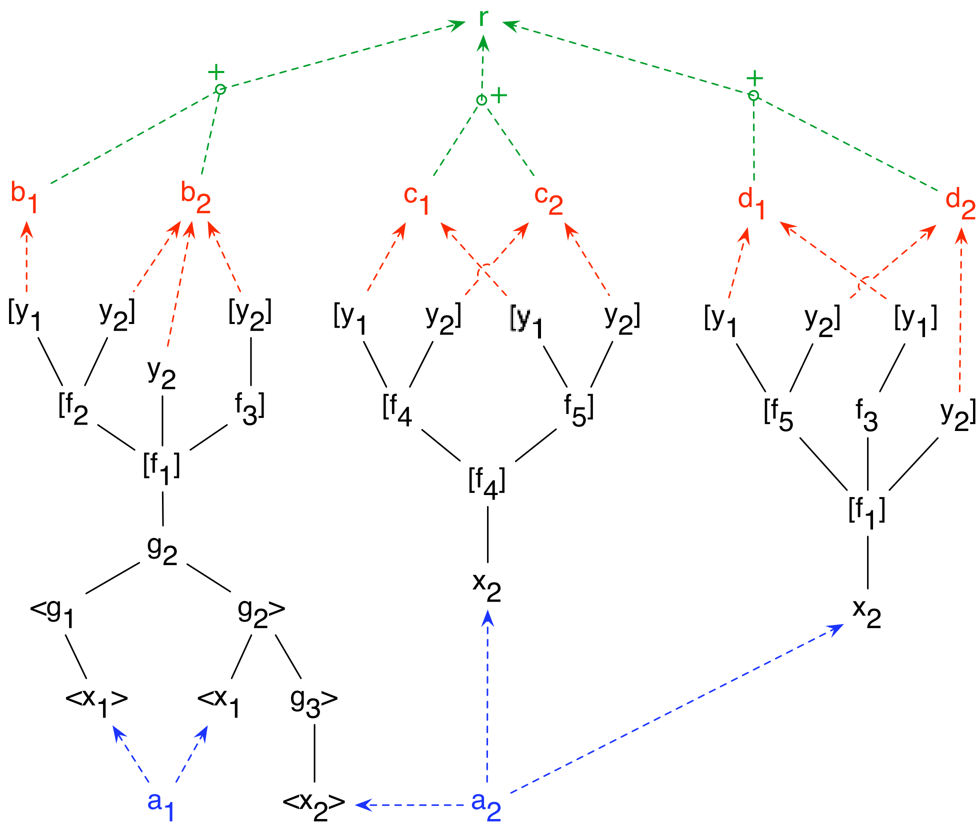
- For all $x \in V$, $x^A(g) = g(x)$.
- For all base sets B of Σ and $b \in B$, $b^A(g) = x$.
- For all $f : e \to e' \in F$ and $t \in T_\Sigma(V)_e$, $f(t)^A(g) = f^A(t^A(g))$.
- For all $n > 1$ and $t_1, \dots, t_n \in T_\Sigma(V)$, $(t_1, \dots, t_n)^A(g) = (t_1^A(g), \dots, t_n^A(g))$.

The **coterm evaluation** $\_^A : coT_\Sigma(Y) \to (A \to A \cdot Y)$ is inductively defined as follows:

- For all $s \in S$, $x \in Y_s$ and $a \in A_s$, $x^A(a) = (a, x)$.
- For all $f : s \to s_1 \dots s_n \in F \setminus BF$, $t_i \in coT_\Sigma(X)_{s_i}$, $1 \le i \le n$, and $a \in A_s$,

$$f^A(a) = (b, i) \;\Rightarrow\; ([t_1, \dots, t_n]f)^A(a) = t_i^A(b).$$

According to their respective intuitive meaning, ground Σ-terms are called **generators** if Σ is constructive, and Σ-terms with a single variable are called **observers** if Σ is destructive.



The data flow induced by the formula $r(t_1, t_2, t_3)$ where
$t_1 = [[[y_1, y_2]f_2, y_2, [y_2]f_3]f_1]g_2\langle g_1\langle x_1\rangle, g_2\langle x_1, g_3\langle x_2\rangle\rangle\rangle$,
$t_2 = [[[y_1, y_2]f_4, [y_1, y_2]f_5]f_4]x_2$ and $t_3 = [[[y_1, y_2]f_5, [y_1]f_3, y_2]f_1]x_2$.

$$r(t_1, t_2, t_3)^A = \{h \in A^X \mid (t_1^A(h), t_2^A(h), t_3^A(h)) \in r^A\}$$

Alternative representation of coT_Σ

Let BA be the union of all base sets of Σ. For all $s \in S$,

$$Beh_{0,s} =_{def} \prod_{t \in Obs_{\Sigma,s}} (BA \times cov(t)).$$

Intuitively, an element of $Beh_{0,s}$ is a tuple of possible results of applying s-observers to any s-element of a Σ-algebra. The result of applying observer t is a pair (a, x) that consists of an "output" value $a \in BA$ and a covariable x of t representing the "exit" where a is returned.

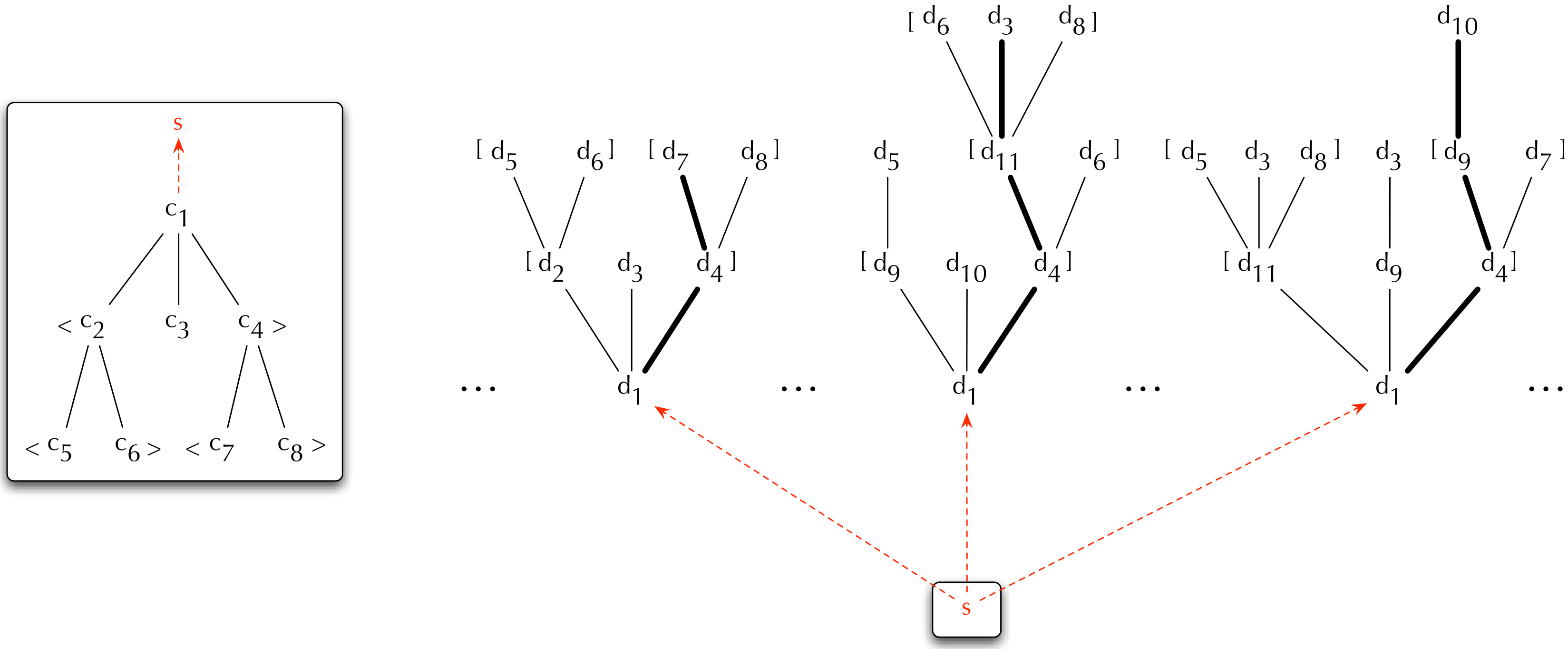
$b \in Beh_{0,s}$ is called a **Σ-behavior** if for all $t, u \in Obs_{\Sigma,s}$, $n \in \mathbb{N}$ and $w \in \mathbb{N}^n$,

$$path(t, w) = path(u, w) \text{ implies } take(n+1)(\pi_2(b_t)) = take(n+1)(\pi_2(b_u)). \qquad (1)$$

By (1), the "runs" of two observers t and u on b "take the same direction" as long as both observers apply the same destructors. In particular, if they start with the same destructor f, they take the same exit of f, formally: for all $b \in Beh_\Sigma(BA)_s$ and $t, u \in Obs_{\Sigma,s}$, $t(\epsilon) = u(\epsilon)$ implies $head(\pi_2(b_t)) = head(\pi_2(b_u))$. Hence

for all $f : s \to s_1 \dots s_n \in F$ and $b \in Beh_{\Sigma,s}$ there is $1 \leq i_{f,b} \leq n$ such that
for all $t \in Obs_{\Sigma,s}$, $t(\epsilon) = f$ implies $head(\pi_2(b_t)) = i_{f,b}$. (2)

An element of $\mu\Sigma \cong T_\Sigma$ (left) resp. $\nu\Sigma_{BA} \cong Beh_\Sigma$ (right):



- For all $s \in S$, $\nu\Sigma_s = Beh_{\Sigma,s}$.
- For all $f : s \to s_1 \dots s_n \in F \setminus BF$ and $(b_t)_{t \in Obs_{\Sigma,s}} \in Beh_{\Sigma,s}$,

$$f^{\nu\Sigma}(b) = (((\langle \pi_1, tail \circ \pi_2 \rangle (b_{[t_1,\dots,t_n]f}))_{t_i \in Obs_{\Sigma,s_i}}, i)$$

where $i = i_{f,b}$ and for all $k \neq i$, $t_k \in Obs_{\Sigma,s_k}$. Note that $head(\pi_1(b_{[t_1,\dots,t_n]f})) = i$.

For all Σ-algebras A, the unique Σ-morphism $unfold^A : A \to \nu\Sigma$ is defined as follows:

For all $s \in S$ and $a \in A_s$,

$$unfold_s^A(a) \ = (t^A(a))_{t \in Obs_{\Sigma,s}}.$$

Labelled Σ-trees

For all $s \in S \setminus BS$, let lab_s be an additional destructor with $dom(lab_s) = s$ and $ran(lab_s) \in BS$, $Lab = \{lab_s \mid s \in S \setminus BS\}$ and $co\Sigma_{Lab} = (S, coF \cup Lab \cup BF, P, B\Sigma)$.

Given an S-sorted set X, the S-sorted set $CT_{\Sigma,Lab}(X)$ of (Σ, Lab)**-trees over** X consists of all partial functions $t : \mathbb{N}^* \to (X \times (F \setminus BF)) \cup X$ such that for all $s \in S$, $t \in CT_{\Sigma,Lab}(X)_s$ iff for all $w \in \mathbb{N}^*$,

- $(\pi_1(t(\epsilon)) \in X_{ran(lab_s)} \wedge \pi_2(t(\epsilon)) \in F \wedge ran(\pi_2(t(\epsilon))) = s) \ \vee \ t(\epsilon) \in X_s.$
- If $\pi_2(t(w)) \in F$, then for all $0 \leq i < |w|$, $s' = dom(\pi_2(t(w)))_i$ and $s'' = ran(\pi_2(t(wi)))$:

$$(s' = s'' \wedge \pi_1(t(wi)) \in X_{ran(lab_{s'})} \wedge \pi_1(t(wi)) \in F) \ \vee \ t(wi) \in X_{s'}.$$

$CT =_{def} CT_{\Sigma,Lab}(BA)$ is final in $Alg_{co\Sigma_{Lab}\downarrow BA}$.

Proof. The following definitions turn CT into a $co\Sigma{\downarrow}BA$-coalgebra:

- For all $s \in S \setminus BS$ and $t \in CT_s$, $|dom(t(\epsilon))| = k$ implies

$$\begin{aligned} d_s^{CT}(t) &=_{def} ((\lambda w.t(0w), \ldots, \lambda w.t((k-1)w)), \pi_1(t(\epsilon))), \\ lab_s^{CT}(t) &=_{def} \pi_2(t(\epsilon)). \end{aligned}$$

- $CT|_{B\Sigma} =_{def} BA$.

Let (A, g) be a $co\Sigma_{Lab}\!\downarrow\! BA$-algebra. An S-sorted function $unfold^A : A \to CT$ is defined as follows:

- For all $s \in S \setminus BS$, $a \in A_s$, $i \in \mathbb{N}$ and $w \in \mathbb{N}^*$, $d_s^A(a) = ((a_1, \ldots, a_n), f)$ implies

$$\begin{aligned} \pi_1(unfold^A(a)(\epsilon)) &=_{def} f, \\ \pi_2(unfold^A(a)(\epsilon)) &=_{def} lab_s^A(a), \\ unfold^A(a)(iw) &=_{def} \begin{cases} unfold^A(a_i)(w) & \text{if } 0 \le i < |dom(f)|, \\ \text{undefined} & \text{otherwise,} \end{cases} \end{aligned}$$

  in short: $unfold^A(a) =_{def} lab_s^A(a) : f(unfold^A(a_1), \ldots, unfold^A(a_n))$.
- $unfold^A|_{B\Sigma} = g$.

$unfold^A$ is a $co\Sigma$-homomorphism: Let $s \in S \setminus BS$, $a \in A_s$ and $d_s^A(a) = ((a_1, \ldots, a_n), f)$.

Then

$$d_s^{CT}(unfold^A(a)) = d_s^{CT}(lab_s^A(a) : f(unfold^A(a_1), \dots, unfold^A(a_n)))$$
$$= ((unfold^A(a_1), \dots, unfold^A(a_n)), f) = unfold^A((a_1, \dots, a_n), f) = unfold^A(d_s^A(a)),$$
$$lab_s^{CT}(unfold^A(a)) = lab_s^{CT}(lab_s^A(a) : f(unfold^A(a_1), \dots, unfold^A(a_n)))$$
$$= lab_s^A(a).$$

Let $h : A \to CT$ be a $co\Sigma$-homomorphism. Then

$$d_s^{CT}(h(a)) = h(d_s^A(a)) = h((a_1, \dots, a_n), f) = ((h(a_1), \dots, h(a_n)), f)$$
$$= d_s^{CT}(lab_s^A(a) : f(h(a_1), \dots, h(a_n))),$$
$$lab_s^{CT}(h(a)) = lab_s^A(a) = lab_s^{CT}(lab_s^A(a) : f(h(a_1), \dots, h(a_n)))$$

and thus $h(a) = f(h(a_1), \dots, h(a_n))$ because $\langle d_s^{CT}, lab_s^{CT} \rangle$ is injective. We conclude that h agrees with $unfold^A$. ❏

Let $C = \{length\}$. $\nu coList'$ is isomorphic to the $coList'$-coalgebra $B =_{def} Tree_{coList,C}(BA)$ of C-colored $coList$-trees over BA (see Colored Σ-trees).

B_{list} can be represented as the union of $\mathbb{N}'$ and the set of partial functions $s : \mathbb{N} \to X \times \mathbb{N}'$ such that $s(0)$ is defined and for all $i \in \mathbb{N}$, if $s(i+1)$ is defined, then $s(i)$ is defined. With respect to this interpretation, the destructors of $coList'$ are interpreted as follows:

$B_1 = \{\infty\}$ and for all $s \in B_{list}$,

$$split^B(s) = \begin{cases} * & \text{if } s \in \mathbb{N}', \\ (\pi_1(s(0)), \lambda i.s(i+1)) & \text{otherwise}, \end{cases}$$

$$length^B(s) = \begin{cases} s & \text{if } s \in \mathbb{N}', \\ \pi_2(s(0)) & \text{otherwise.} \end{cases}$$

Let AX be given by the $coList'$-formulas

$$\in_{list}(s) \;\Rightarrow\; \in_{1+entry\times list}([[x,y]split]s) \quad (1)$$

$$\in_{entry\times list}(p) \;\Rightarrow\; \in_{list}(\pi_2\langle p\rangle) \quad (2)$$

$$\in_{list}(s) \;\Rightarrow\; [[x,y]length]s = [[[x]0, [[[x]succ, y]length]\pi_2]split]s \quad (3)$$

AX consists of inverse Horn clauses over $coList'$ that satisfy the assumptions of Restriction with a greatest invariant. Hence $gfp(\overline{AX}) = B$. Let $inv = \in^B$.

For all $s, s' \in inv_{list}$,

$$length^B(s) \neq length^B(s') \text{ implies } t^B(s) \neq t^B(s') \text{ for some } t \in Obs_{coList,list}. \quad (4)$$

Proof.

Since B satisfies (3), inv satisfies the conclusion of (3) or, equivalently, the equations

(1)-(3) of 1.6. Hence $s \in inv_{list}$ iff for all $n \in \mathbb{N}$,

$$length^B(s) = 0 \text{ implies } split^B(s) = *, \tag{5}$$

$$length^B(s) = n+1 \text{ implies } \exists\, e, s' : (split^B(s) = (e, s') \wedge length^B(s') = n), \tag{6}$$

$$length^B(s) = \infty \text{ implies } \exists\, e, s' : (split^B(s) = (e, s') \wedge length^B(s') = \infty). \tag{7}$$

It is easy to see that

- $Obs_{coList,list} = \{obs_n \mid n \in \mathbb{N}\}$ where $obs_0 = [0, [10]\pi_1]split$ and for all $n > 0$, $obs_n = [0, [10 \cdot obs_{n-1}]\pi_2]split$,
- for all $s \in B_{list}$ and $n \in \mathbb{N}$, $obs_n(s) \neq *$ iff $s(n)$ is defined. (8)

By (5)-(7) and the definition of B, for all $s \in inv_{list}$ and $n \in \mathbb{N}$,

$$length^B(s) = n \;\Leftrightarrow\; s(n) \text{ is undefined } \wedge \forall\, i < n : s(i) \text{ is defined},$$

$$length^B(s) = \infty \;\Leftrightarrow\; \forall\, n \in \mathbb{N} : s(n) \text{ is defined},$$

and thus by (8),

$$length^B(s) = n \;\Leftrightarrow\; obs_n^B(s) = * \wedge \forall\, i < n : obs_i^B(s) \neq *, \tag{9}$$

$$length^B(s) = \infty \;\Leftrightarrow\; \forall\, n \in \mathbb{N} : obs_n^B(s) \neq *. \tag{10}$$

Let $s, s' \in B_{list}$ such that $length^B(s) \neq length^B(s')$. Then $length^B(s) = n$ or $length^B(s') = n$ for some $n \in \mathbb{N}$. W.l.o.g. suppose that the first case holds true. By

(9), $obs_n^B(s) = *$. If $length^B(s') = \infty$, then (10) implies a contradiction: $obs_n^B(s) \neq * = obs_n^B(s)$. Otherwise $length^B(s') = n'$ for some $n' \in \mathbb{N}$ with $n' \neq n$. Let $m = min(n, n')$. If $n < n'$, then by (9), $obs_m^B(s) = obs_n^B(s) = * \neq obs_n^B(s') = obs_m^B(s')$. Otherwise $n' < n$ and thus by (9), $obs_m^B(s') = obs_{n'}^B(s') = * \neq obs_{n'}^B(s) = obs_m^B(s)$. Hence (4) is valid for $t = obs_m$. ❑

Let $C = \{subtree\}$. $\nu coBintree'$ is isomorphic to the $coBintree'$-coalgebra

$$B =_{def} Tree_{coBintree,C}(BA)$$

of C-colored $coBintree$-trees over BA (see Colored Σ-trees).

Let $Z = Btree(X)^\infty \rightarrow Btree(X)^\infty$. B_{btree} can be represented as the set of partial functions

$$t : 2^* \rightarrow X \times Z$$

such that for all $w \in 2^*$ and $b \in 2$, if $t(wb)$ is defined, then $t(w)$ is defined.

With respect to this interpretation, the destructors of $coBintree'$ are interpreted as follows: For all $t \in B_{tree}$,

$$fork^B(t) = \begin{cases} * & \text{if } t = \Omega, \\ (\lambda w.t(0w), \pi_1(t(\epsilon)), \lambda w.t(1w)) & otherwise, \end{cases}$$
$$subtree^B(t) = \pi_2(t(\epsilon)).$$

Let AX be given by the $coBintree'$-formulas

$$\in_{btree}(t) \Rightarrow \in_{1+btree\times entry\times btree}(fork\langle t\rangle) \wedge \in_{btree^{blist}}(subtree\langle t\rangle) \quad (1)$$
$$\in_{btree\times entry\times btree}(p) \Rightarrow \in_{btree}(\pi_1\langle p\rangle) \wedge \in_{btree}(\pi_3\langle p\rangle) \quad (2)$$
$$\in_{btree^{blist}}(f) \Rightarrow \in_{btree}(\$w\langle f\rangle) \quad (3)$$
$$\begin{array}{rl} \in_{btree}(t) \Rightarrow & \exists\, p, q : ([[x,y]fork]t = [x]p \wedge \$\epsilon\langle subtree\langle t\rangle\rangle = t) \vee \\ & \exists\, p, q : ([[x,y]fork]t = [y]p \wedge \\ & \quad \$0w\langle subtree\langle t\rangle\rangle = \$w\langle subtree\langle \pi_1\langle p\rangle\rangle\rangle \wedge \\ & \quad \$1w\langle subtree\langle t\rangle\rangle = \$w\langle subtree\langle \pi_3\langle p\rangle\rangle\rangle) \end{array} \quad (4)$$

for all $w \in 2^*$. AX consists of inverse Horn clauses over $coBintree'$ that satisfy the assumptions of Restriction with a greatest invariant. Hence $gfp(\overline{AX}) = B$. Let $inv = \in^B$.

For all $t, t' \in inv_{btree}$,
$subtree^B(t) \neq subtree^B(t')$ implies $u^B(t) \neq u^B(t')$ for some $u \in Obs_{coBintree,btree}$. (5)

Proof.
Since B satisfies (4), inv satisfies the conclusion of (4) or, equivalently, the equations (1)-(3) of 2.13. Hence $t \in inv_{btree}$ iff for all $w \in 2^*$,

$$subtree^B(t)(\epsilon) = t, \tag{6}$$
$$fork^B(t) = (u, e, u') \quad \text{implies} \quad subtree^B(t)(0{:}w) = subtree^B(u)(w), \tag{7}$$
$$fork^B(t) = (u, e, u') \quad \text{implies} \quad subtree^B(t)(1{:}w) = subtree^B(u')(w). \tag{8}$$

It is easy to see that

- $Obs_{coBintree,btree} = \{obs_w \mid w \in 2^+\}$ where $obs_\epsilon = [0, [10]\pi_2]fork$ and for all $w \in \mathbb{N}^+$, $obs_{0w} = [0, [10 \cdot obs_w]\pi_1]fork$ and $obs_{1w} = [0, [10 \cdot obs_w]\pi_3]fork$,
- for all $t \in B_{tree}$ and $w \in \mathbb{N}^*$, $obs^B_w(t) = t(w)$ if $t(w)$ is defined, and $obs_w(t) = *$ otherwise. (9)

By (6)-(8) and the definition of B, for all $t \in inv_{btree}$ and $v \in 2^*$,

$$subtree^B(t)(v) = \lambda w.t(vw),$$

and thus by (9),

$$subtree^B(t)(v) = \lambda w.obs_{vw}(t). \tag{10}$$

Let $t, t' \in B_{btree}$ and $w \in 2^*$ such that $subtree^B(t) \neq subtree^B(t')$. Then there are $v, w \in 2^*$ such that$subtree^B(t)(v)(w) \neq subtree^B(t)(v)(w)$. Hence by (10), $\lambda w.obs_{vw}(t) \neq \lambda w.obs_{vw}(t)$, and thus (5) is valid for $u = obs^B_{vw}$. ❏

Let $\in^A = \nu\, coTree$. Then A satisfies AX. Hence $A \in Alg^{\in}_{coTree',AX}$ and thus by Lemma DESEXT (2), (6) and (7) imply $\in^B|_{coTree} \cong \nu\, coTree$.

Bibliography

References

[1] P. Aczel, *An Introduction to Inductive Definitions*, in: J. Barwise, ed., Handbook of Mathematical Logic, North-Holland (1977) 739-782

[2] J. Adámek, *Free algebras and automata realizations in the language of categories*, Commentat. Math Univers. Carolinae 15 (1974) 589-602

[3] J. Adámek, *Introduction to Coalgebra*, Theory and Applications of Categories 14 (2005) 157-199

[4] J. Adámek, *Final coalgebras are ideal completions of initial algebras*, Journal of Logic and Computation 12 (2002) 217-242

[5] P. Aczel, J. Adámek, J. Velebil, *A Coalgebraic View of Infinite Trees and Iteration*, Proc. Coalgebraic Methods in Computer Science, Elsevier ENTCS 44 (2001) 1-26

[6] M.A. Arbib, *Free dynamics and algebraic semantics*, Proc. Fundamentals of Computation Theory, Springer LNCS 56 (1977) 212-227

[7] M.A. Arbib, E.G. Manes, *Arrows, Structures, and Functors*, Academic Press 1975

[8] M.A. Arbib, E.G. Manes, *Parametrized Data Types Do Not Need Highly Constrained Parameters*, Information and Control 52 (1982) 139-158

[9] J. Adámek, S. Milius, L.S. Moss, *Initial algebras and terminal coalgebras: a survey*, draft of Feb. 7, 2011, TU Braunschweig

[10] J. Adámek, H.–E. Porst, *From varieties of algebras to covarieties of coalgebras*, Proc. Coalgebraic Methods in Computer Science, Elsevier ENTCS 44 (2001) 27–46

[11] J. Adámek, H.–E. Porst, *On Tree Coalgebras and Coalgebra Presentations*, Theoretical Computer Science 311 (2004) 257-283

[12] M. Barr, *Terminal coalgebras in well-founded set theory*, Theoretical Computer Science 114 (1993) 299-315

[13] M. Barr, *Terminal coalgebras for endofunctors on sets*, ftp://ftp.math.mcgill.ca/pub/barr/pdffiles/trmclg.pdf, McGill University, Montreal 1999

[14] M. Barr, *Coequalizers and Free Triples*, Math. Zeitschrift 116 (1970) 307-322

[15] M. Barr, Ch. Wells, *Category Theory*, Lecture Notes for ESSLLI, 1999, *link*

[16] J.A. Brzozowski, *Derivatives of regular expressions*, Journal ACM 11 (1964) 481–494

[17] J.R.B. Cockett, D. Spencer, *Strong categorical datatypes II: A term logic for categorical programming*, Theoretical Computer Science 139 (1995) 69-113, *link*

[18] C. Cîrstea, *A coalgebraic equational approach to specifying observational structures*, Theoretical Computer Science 280 (2002) 35–68, link

[19] M. Erwig, *Categorical Programming with Abstract Data Types*, Proc. AMAST'98, Springer LNCS 1548, 406-421, link

[20] M.M. Fokkinga, *Program Calculation Properties of Continuous Algebras*, CWI Report CS-R9104, Amsterdam 1991

[21] J. Gibbons, G. Hutton, Th. Altenkirch, *When is a function a fold or an unfold?*, ENTCS 44 (2001), link

[22] J.A. Goguen, R. Burstall, *Institutions: Abstract Model Theory for Specification and Programming*, J. ACM 39 (1992) 95-146, link

[23] J.A. Goguen, J.W. Thatcher, E.G. Wagner, J.B. Wright, *Initial Algebra Semantics and Continuous Algebras*, J. ACM 24 (1977) 68-95, link

[24] H.P. Gumm, T. Schröder, *Coalgebras of bounded type*, Math. Structures in Computer Science 12 (2002),565-578

[25] H.P. Gumm, *State Based Systems are Coalgebras*, Cubo - Matematica Educacional 5 (2003) 239-262

[26] H.P. Gumm, *Universelle Coalgebra*, in: Th. Ihringer, *Allgemeine Algebra*, Heldermann Verlag 2003

[27] G. Gupta et al., *Infinite Computation, Co-induction and Computational Logic*, Proc. CALCO 2011, Springer LNCS 6859 (2011) 40-54

[28] R. Hinze, *Adjoint Folds and Unfolds*, Proc. Mathematics of Program Construction 2010, Springer LNCS 6120, 195–228, *link*

[29] B. Jacobs, *Invariants, Bisimulations and the Correctness of Coalgebraic Refinements*, Proc. Algebraic Methodology and Software Technology, Springer LNCS 1349 (1997) 276-291

[30] B. Jacobs, *Introduction to Coalgebra*, Radboud University Nijmegen 2005

[31] B. Jacobs, *Exercises in Coalgebraic Specification*, Springer LNCS 2297 (2002) 237-280, *link*

[32] B. Jacobs, *A Bialgebraic Review of Deterministic Automata, Regular Expressions and Languages*, in: K. Futatsugi et al. (eds.), Goguen Festschrift, Springer LNCS 4060 (2006) 375–404

[33] B. Jacobs, J. Rutten, *A Tutorial on (Co)Algebras and (Co)Induction*, EATCS Bulletin 62 (1997) 222-259, *link*

[34] G. Jarzembski, *A new proof of Reiterman's theorem*, Cahiers de topologie et géométrie différentielle catégoriques 35 (1994) 239-247

[35] S. Kamin, *Final data types an their specification extension*, ACM Trans. on Prog. Lang. and Systems Comp. Syst. Sci. 5 (1983) 97-123

[36] D. Kozen, *Realization of Coinductive Types*, Proc. Math. Foundations of Prog. Lang. Semantics 27, Carnegie Mellon University, Pittsburgh 2011

[37] S.C. Kleene, *Introduction to Metamathematics*, Van Nostrand 1952

[38] B. Klin, *Bialgebraic Methods and Modal Logic in Structural Operational Semantics*, Elsevier 2007

[39] J. Lambek, *A fixpoint theorem for complete categories*, Math. Zeitschrift 103 (1968) 151-161

[40] J.-L. Lassez, V.L. Nguyen, E.A. Sonenberg, *Fixed Point Theorems and Semantics: A Folk Tale*, Information Processing Letters 14 (1982) 112-116

[41] F.W. Lawvere, *Diagonal arguments in cartesian closed categories*, Reprints in Theory and Applications of Categories 15 (2006) 1–13

[42] D.J. Lehmann, M.B. Smyth, *Algebraic Specification of Data Types: A Synthetic Approach*, Math. Systems Theory 14 (1981) 97-139

[43] G. Markowsky, *Chain-complete posets and directed sets with applications*, Algebra Universalis 6 (1976) 53-68

[44] E. Meijer, G. Hutton, *Bananas in Space: Extending Fold and Unfold to Exponential Types*, Proc. FPCA'95, ACM Publications, 324-333, *link*

[45] T. Mossakowski, L. Schröder, M. Roggenbach, H. Reichel, *Algebraic-coalgebraic specification in CoCASL*, J. Logic and Algebraic Programming 67 (2006) 146-197, *link*

[46] P. Padawitz, *Algebraic Model Checking and more*, *slides* and *short paper*

[47] P. Padawitz, *Expander2 as a Prover and Rewriter*, *link*

[48] P. Padawitz, *Formale Methoden des Systementwurfs*, *link*

[49] P. Padawitz, *Übersetzerbau*, course notes, TU Dortmund 2010, *link*

[50] P. Padawitz, *Church-Rosser-Eigenschaften von Graphgrammatiken und Anwendungen auf die Semantik von LISP*, Diplomarbeit, TU Berlin 1978

[51] S. Phillips, W.H. Wilson, G.S. Halford, *What Do Transitive Inference and Class Inclusion Have in Common? Categorical (Co)Products and Cognitive Development*, PLoS Computational Biology 5 (2009), *link*

[52] S. Phillips, W.H. Wilson, *Categorial Compositionality: A Category Theory Explanation for the Systematicity of Human Cognition*, PLoS Computational Biology 6 (2010), *link*

[53] B. Pierce, *Basic Category Theory for Computer Scientists*, MIT Press 1991

[54] J. Reiterman, *The Birkhoff theorem for finite algebras*, Algebra Universalis 14 (1982) 1-10

[55] J.J.M.M. Rutten, *Universal coalgebra: a theory of systems*, Theoretical Computer Science 249 (2000) 3-80

[56] J.J.M.M. Rutten, *Automata and coinduction (an exercise in coalgebra)*, Proc. CONCUR '98, Springer LNCS 1466 (1998) 194–218

[57] J.J.M.M. Rutten, *Behavioural differential equations: a coinductive calculus of streams, automata, and power series*, Theoretical Computer Science 308 (2003) 1-53

[58] J.J.M.M. Rutten, *A coinductive calculus of streams*, Math. Struct. in Comp. Science 15 (2005) 93-147

[59] J. Soto-Andrade, F.J. Varela, *Self-Reference and Fixed Points: A Discussion and an Extension of Lawvere's Theorem*, Acta Applicandae Mathematicae 2 (1984) 1-19

[60] K. Sen, G. Rosu, *Generating Optimal Monitors for Extended Regular Expressions*, Proc. Runtime Verification 2003, Elsevier ENTCS 89 (2003) 226-245

[61] L. Simon, *Coinductive Logic Programming*, Ph.D. thesis, University of Texas at Dallas (2006)

[62] A. Tarski, *A lattice-theoretical fixpoint theorem and its applications*, Pacific J. Math. 5 (1955), 285-309

[63] D. Turi, G. Plotkin, *Towards a Mathematical Operational Semantics*, Proc. Logic in Computer Science, IEEE Computer Society Press (1997) 280–291, *link*

[64] J.W. Thatcher, E.G. Wagner, J.B. Wright, *More on Advice on Structuring Compilers and Proving Them Correct*, Theoretical Computer Science 15 (1981) 223-249

[65] T. Uustalu, V. Vene, *Primitive (Co)Recursion and Course-of-Value (Co)Iteration*, INFORMATICA 10 (1999) 5-26, *link*

[66] Ph. Wadler, *Theorems for free!*, Proc. FPLCA '89, ACM Press (1989) 347-359

[67] M. Wand, *Final algebra semantics and data type extension*, J. Comp. Syst. Sci. 19 (1979) 27-44

[68] R.F.C. Walters, *Categories and Computer Science*, Cambridge University Press 1992

[69] J. Winter, M.M. Bonsague, J. Rutten, *Context-Free Languages, Coalgebraically*, Proc. CALCO 2011

[70] N.S. Yanofsky, *A Universal Approach to Self-Referential Paradoxes, Incompleteness and Fixed Points*, The Bulletin of Symbolic Logic 9 (2003) 362-386