Strong Dinaturality and Initial Algebras

Tarmo Uustalu

Inst. of Cybern., Tallinn, Estonia

U. of Minho, Braga, Portugal

www.cs.ioc.ee/∼tarmo/

NWPT'00, Bergen, Norway, 11–13 Oct. 2000

www.ii.uib.no/∼nwpt00/

Non-constructive, constructive Tarski thms.

- Given a preorder $(X, \leq)$ and a function $f : X \to X$ monotone wrt. $\leq$.
- Then, if an infimum of $\{x \mid x \in X, fx \leq x\}$ exists, then a least prefixed point of f also exists:

$$\mathrm{lpfp}\, f = \inf \{x \mid x \in X, fx \leq x\}$$

The converse is true as well:

$$\inf \{x \mid x \in X, fx \leq x\} = \mathrm{lpfp}\, f$$

- If X has a bottom element and f is upward ω-continuous, then if a supremum exists for $\{f_*^i \mid i \in \omega\}$, where $f_* : \omega \to X$, $f_*^0 = \bot$, $f_*^{i+1} = f\, f_*^i$, then a least prefixed point of f also exists:

$$\mathrm{lpfp}\, f = \sup \{f_*^i \mid i \in \omega\}$$

Again, the converse is true, too:

$$\sup \{f_*^i \mid i \in \omega\} = \mathrm{lpfp}\, f$$

From preorders to categories...

preorder	$\rightsquigarrow$	category
monotone function	$\rightsquigarrow$	functor
[least] prefixed point of a monot. fun.	$\rightsquigarrow$	[initial] algebra of a functor
"selective"	$\rightsquigarrow$	strongly (!) weighted
[greatest] lower bound [infimum] of a monot. fun.		[final] cone [limit] of a functor

Algebras, initial algebras

- Given a category $\mathcal{C}$ and a functor $F : \mathcal{C} \to \mathcal{C}$.
- An *F-algebra* is a pair (A, ϕ) formed of an object A of $\mathcal{C}$ and a morphism $\phi : F\,A \to A$ of $\mathcal{C}$.
- Given F-algebras (A, ϕ), (B, ψ),
  an *F-algebra map* $(A, \phi) \to (B, \psi)$ is a morphism $g : A \to B$ of $\mathcal{C}$ s.t.

$$\begin{array}{ccc} F\,A & \xrightarrow{\varphi} & A \\ {\scriptstyle F\,g}\downarrow & & \downarrow{\scriptstyle g} \\ F\,B & \xrightarrow{\psi} & B \end{array}$$

- $(\mu F, \mathsf{in}_F)$ denotes "the" initial F-algebra (if there is one).
- Given an F-algebra (A, ϕ),
  $(\!|\, \phi \,|\!)_{F,A}$ $[\mathsf{cata}_{F,A}\, \phi]$ denotes the unique F-algebra map
  $(\mu F, \mathsf{in}_F) \to (A, \phi)$.

- $(\mu F, \mathsf{in}_F)$ is an F-algebra.
- For any F-algebra (A, ϕ),

$$(\!|\, \phi \,|\!)_{F,A} \circ \mathsf{in}_F = \phi \circ F\, (\!|\, \phi \,|\!)_{F,A}$$

-

$$\mathsf{id}_{\mu F} = (\!|\, \mathsf{in}_F \,|\!)_{F,\mu F}$$

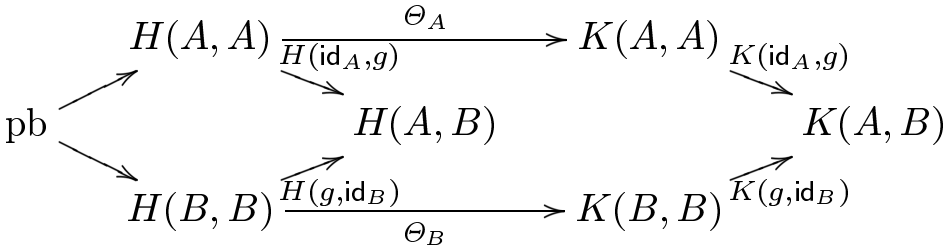
- For any F-algebras (A, ϕ), (B, ψ) and morphism $g : A \to B$ of $\mathcal{C}$,

$$g \circ \phi = \psi \circ F\, g \quad \Rightarrow \quad g \circ (\!|\, \phi \,|\!)_{F,A} = (\!|\, \psi \,|\!)_{F,B}$$

Strong dinaturals

- Given a category $\mathcal{C}$, a category $\mathcal{E}$ with pullbacks, and bifunctors $H, K : \mathcal{C}^{\mathrm{op}} \times \mathcal{C} \to \mathcal{E}$.
- A *strong dinat. transf.* $H \to K$ is a family Θ of morphisms $\Theta_A : H(A, A) \to K(A, A)$ in $\mathcal{E}$, one for each object A of $\mathcal{C}$, s.t.

  for any objects A, B and morphism $g : A \to B$ of $\mathcal{C}$,



- In the important special case $\mathcal{E} = \mathsf{Set}$, this means that Θ is a family of functions $\Theta_A : H(A,A) \to K(A,A)$, one for each object A of $\mathcal{C}$, s.t.

  for any objects A, B and morphism $g : A \to B$ of $\mathcal{C}$,
  for any $x \in H(A,A)$, $y \in H(B,B)$,

$$\begin{array}{ccc} H\,(\mathsf{id}_A, g)\,x & & K\,(\mathsf{id}_A, g)\,(\Theta_A\,x) \\ = & \Rightarrow & = \\ H\,(g, \mathsf{id}_B)\,y & & K\,(g, \mathsf{id}_B)\,(\Theta_B\,y) \end{array}$$

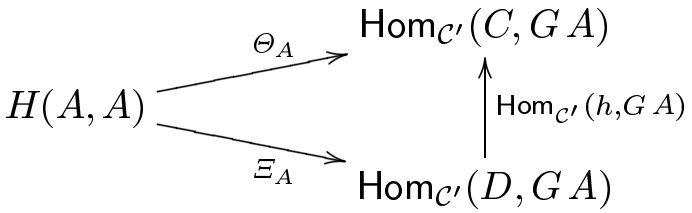
Strongly weighted cones, strongly weighted limits

- Given a category $\mathcal{C}$, a locally small category $\mathcal{C}'$, a bifunctor $H : \mathcal{C}^{\mathrm{op}} \times \mathcal{C} \to \mathsf{Set}$, and a functor $G : \mathcal{C} \to \mathcal{C}'$.
- A *strongly* H*-weighted* G*-cone* is a pair (C, Θ) formed of an object C of $\mathcal{C}'$ and a strong dinat. transf. $\Theta : H \to \mathsf{Hom}_{\mathcal{C}'}(C, G-)$, i.e., a family of functions $\Theta_A : H(A,A) \to \mathsf{Hom}_{\mathcal{C}'}(C, G\,A)$, one for each object A of $\mathcal{C}$, s.t.

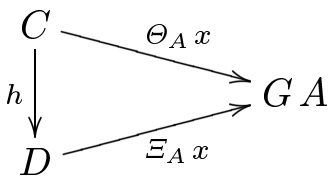
  for any objects A, B and morphism $g : A \to B$ of $\mathcal{C}$,
  for any $x \in H(A,A), y \in H(B,B)$,

$$\begin{array}{ccc} H\,(\mathsf{id}_A, g)\,x & & \\ = & \Rightarrow & \\ H\,(g, \mathsf{id}_B)\,y & & \end{array} \qquad \begin{array}{ccc} & \overset{\Theta_A\,x}{\nearrow} & G\,A \\ C & & \downarrow{\scriptstyle G\,g} \\ & \underset{\Theta_B\,y}{\searrow} & G\,B \end{array}$$

- Given strongly H-weighted G-cones (C, Θ), (D, Ξ),
  a *strongly H-weighted G-cone map* $(C, \Theta) \to (D, \Xi)$ is a morphism $h : C \to D$ of $\mathcal{C}'$ s.t.
  for any object A of $\mathcal{C}$,



  i.e., for any object A of $\mathcal{C}$,
  for any $x \in H(A, A)$,



- $(\Pi(H, G), \mathsf{Out}^H_G)$ denotes "the" <u>final</u> strongly H-weighted G-cone (strongly H-weighted G-limit) (if there is one).
- Given a strongly H-weighted G-cone (C, Θ),
  $\langle\, \Theta \,\rangle^H_{G,C}\ [\mathsf{Split}^H_{G,C}\ \Theta]$ denotes the unique strongly H-weighted G-cone map $(C, \Theta) \to (\Pi(H, G), \mathsf{Out}^H_G)$.

- $(\Pi(H,G), \mathsf{Out}_G^H)$ is a strongly H-weighted G-cone.
- For any strongly H-weighted G-cone (C, Θ),

$$\forall_A\, x \in H(A,A).\ (\mathsf{Out}_G^H)_A\, x \circ \langle\, \Theta \,\rangle_{G,C}^H = \Theta_A\, x$$

- $$\mathsf{id}_{\Pi(H,G)} = \langle\, \mathsf{Out}_G^H \,\rangle_{G,\Pi(H,G)}^H$$

- For any strongly H-weighted G-cones (C, Θ), (D, Ξ) and morphism $h : C \to D$ of $\mathcal{C}'$,

$$(\forall_A\, x \in H(A,A).\ \Theta_A\, x = \Xi_A\, x \circ h) \quad \Rightarrow \quad \langle\, \Theta \,\rangle_{G,C}^H = \langle\, \Xi \,\rangle_{G,D}^H \circ h$$

Non-constructive Tarski thm. for categories

- Given a locally small category $\mathcal{C}$ and a functor $F : \mathcal{C} \to \mathcal{C}$.
- If a $\mathsf{Hom}_{\mathcal{C}}(F-,-)$-weighted $\mathsf{Id}_{\mathcal{C}}$-limit exists, then an initial F-algebra also exists:

$$\begin{aligned}
\mu F &= \Pi(\mathsf{Hom}(F-,-), \mathsf{Id}) \\
(\!|\, \phi \,|\!)_{F,A} &= (\mathsf{Out}_{\mathsf{Id}}^{\mathsf{Hom}(F-,-)})_A\, \phi \\
\mathsf{in}_F &= \langle\, \lambda_A\, \phi.\ \phi \circ F\, (\!|\, \phi \,|\!)_{F,A} \,\rangle_{\mathsf{Id},F\,\mu F}^{\mathsf{Hom}(F-,-)}
\end{aligned}$$

- The converse is also true:

$$\begin{aligned}
\Pi(\mathsf{Hom}(F-,-), \mathsf{Id}) &= \mu F \\
\mathsf{Out}_{\mathsf{Id}}^{\mathsf{Hom}(F-,-)} &= \lambda_A\, \phi.\ (\!|\, \phi \,|\!)_{F,A} \\
\langle\, \Theta \,\rangle_{\mathsf{Id},C}^{\mathsf{Hom}(F-,-)} &= \Theta_{\mu F}\, \mathsf{in}_F
\end{aligned}$$

Strong dinaturality and initial algebras (cf. Yoneda)

- Given a locally small category $\mathcal{C}$ and a functor $K : \mathcal{C} \to \mathsf{Set}$.
- Then, given a functor $F : \mathcal{C} \to \mathcal{C}$ s.t. an initial F-algebra exists,
  - $\mathsf{SDinat}(\mathsf{Hom}_{\mathcal{C}}(F -, -), K)$ is a set
  - $$\begin{array}{rcl} i_F : \mathsf{SDinat}(\mathsf{Hom}(F -, -), K) & \to & K\,\mu F \\ \Theta & \mapsto & \Theta_{\mu F}\,\mathsf{in}_F \text{ is an iso} \\ \lambda_A\,\phi.\ K\,(\!|\,\phi\,|\!)_{F,A}\,x & \leftarrow\!\shortmid & x \end{array}$$

- Also, given functors $F, G : \mathcal{C} \to C$ s.t. initial F, G-algebras exist and a nat. transf. $\tau : F \to G$, we have

  $$K\,\mu\tau\,(i_F\,\Theta) = i_G\,(\Theta \circ \mathsf{Hom}(\tau -, -))$$

  where

  $$\begin{array}{rcl} \mu\tau & = & (\!|\,\mathsf{in}_G \circ \tau_{\mu G}\,|\!)_{F,\mu G} \\ \mathsf{Hom}(\tau -, -) & = & \lambda_A\,\phi.\ \phi \circ \tau_A \end{array}$$

  Thus, i_F is natural in F.

Non-constructive Tarski thm. in programming

- $\mathsf{Split}_{\mathsf{Id},1}^{\mathsf{Hom}(F-,-)}$ is the build_F operator from shortcut deforestation!
- Natural numbers, ver. 1: initial algebra:

```
data Nat = Suc Nat | Zer

foldnat :: (a -> a) -> a -> Nat -> a
foldnat s z (Suc n) = s (foldnat s z n)
foldnat s z Zer     = z

buildnat :: (forall a. (a -> a) -> a -> a) -> Nat
buildnat theta = theta Suc Zer
```


- Natural numbers, ver. 2: strongly weighted limit:

```
data Nat = Buildnat (forall a. (a -> a) -> a -> a)

foldnat :: (a -> a) -> a -> Nat -> a
foldnat s z (Buildnat theta) = theta s z

suc :: Nat -> Nat
suc n = Buildnat (\s -> \z -> s (foldnat s z n))
zer :: Nat
zer   = Buildnat (\s -> \z -> z)
```

- General inductive types, ver. 1: initial algebras:

```
data Mu f = In (f (Mu f))

cata :: Functor f => (f a -> a) -> Mu f -> a
cata phi (In x) = phi (fmap (cata phi) x)

build :: (forall a. (f a -> a) -> a) -> Mu f
build theta = theta In
```


- General inductive types, ver. 2: strongly weighted limits:

```
data Mu f = Build (forall a. (f a -> a) -> a)

cata :: (f a -> a) -> Mu f -> a
cata phi (Build theta) = theta phi

in :: Functor f => f (Mu f) -> Mu f
in x = Build (\phi -> phi (fmap (cata phi) x))
```