Containers in Higher Kinds
jww Zhili Tian and Håkon Gylterud

Thorsten Altenkirch

Functional Programming Laboratory
School of Computer Science
University of Nottingham

April 16, 2025

Old paper

TLCA01

A. "Representations of first order function types as terminal coalgebras." Typed Lambda Calculi and Applications: 5th International Conference, TLCA 2001

Example : Streams

Stream A $\cong \mathbb{N} \to$ A

Given $\mathbb{N} \cong 1 + \mathbb{N}$

$$\begin{aligned} \mathbb{N} \to \mathsf{A} &\cong (1 + \mathbb{N}) \to \mathsf{A} \\ &\cong \mathsf{A} \times (\mathbb{N} \to \mathsf{A}) \end{aligned}$$

Hence:
$\mathbb{N} \to \mathsf{A} \cong \nu\,\mathsf{X} : \mathsf{Set}.\ \mathsf{A} \times \mathsf{X}$

What about trees?

$$
\begin{array}{l}
\textbf{data}\ \mathsf{BT}\ :\ \mathsf{Type}\ \textbf{where} \\
\quad \mathsf{leaf}\ :\ \mathsf{BT} \\
\quad \mathsf{node}\ :\ \mathsf{BT}\ \rightarrow\ \mathsf{BT}\ \rightarrow\ \mathsf{BT}
\end{array}
$$

Given $\mathsf{BT} \cong 1 + \mathsf{BT} \times \mathsf{BT}$

$$\begin{aligned}\mathsf{BT} \to \mathsf{A} &\cong (1 + \mathsf{BT} \times \mathsf{BT}) \to \mathsf{A} \\ &\cong \mathsf{A} \times (\mathsf{BT} \times \mathsf{BT} \to \mathsf{A}) \\ &\cong \mathsf{A} \times (\mathsf{BT} \to (\mathsf{BT} \to \mathsf{A}))\end{aligned}$$

Hence:

$\mathsf{BT} \to \mathsf{A} \cong (\nu\, \mathsf{F} : \mathsf{Set} \to \mathsf{Set} \,.\, \lambda\, \mathsf{X} \,.\, \mathsf{X} \times \mathsf{F}\,(\mathsf{F}\,\mathsf{X}))\,\mathsf{A}$

record Bush (A : Type) : Type **where**
  coinductive
  field
    head : A
    tail : Bush (Bush A)

app : Bush A $\rightarrow$ BT $\rightarrow$ A
app bsh leaf = head bsh
app bsh (node l r) = app (app (tail bsh) l) r

lam : (BT $\rightarrow$ A) $\rightarrow$ Bush A
head (lam f) = f leaf
tail (lam f) = ϕ (λ l $\rightarrow$ ϕ (λ r $\rightarrow$ f (node l r)))

Another old paper

TCS05

Abbott, Michael, A., and Neil Ghani.
"Containers: Constructing strictly positive types."
Theoretical Computer Science 342.1 (2005):

```
record Cont : Set₁ where
  constructor _ ◁ _
  field
    S : Set
    P : S → Set

⟦_⟧ : Cont → Set → Set
⟦ S ◁ P ⟧ X = Σ[ s ∈ S ] (P s → X)
```

Results on containers

- Initial and terminal algebras

$$\mu[\![\,\mathsf{S} \;\triangleleft\; \mathsf{P}\,]\!] \cong \mathsf{W}\;\mathsf{S}\;\mathsf{P}$$
$$\nu[\![\,\mathsf{S} \;\triangleleft\; \mathsf{P}\,]\!] \cong \mathsf{M}\;\mathsf{S}\;\mathsf{P}$$

- Container morphisms

$$\int_{X:\mathsf{Set}} [\![\,\mathsf{S} \;\triangleleft\; \mathsf{P}\,]\!]\;\mathsf{X} \;\rightarrow\; [\![\,\mathsf{T} \;\triangleleft\; \mathsf{Q}\,]\!]\;\mathsf{X} \cong$$
$$\Sigma[\,\mathsf{f} \;\in\; \mathsf{S} \;\rightarrow\; \mathsf{T}\,]\,(\mathsf{Q}\,(\mathsf{f}\;\mathsf{s}) \;\rightarrow\; \mathsf{P}\;\mathsf{s})$$

- Cont is cartesian closed.
  CIE 2010 (w Paul Levy and Sm Staton)
- Cont gives rise to a Category with families (CwF)
  TYPES 2021 abstract (w Ambrus Kaposi)

.

How to express the results from the TLCA01 paper in the language of TCS05?

What are containers in higher kinds?

$$\mathsf{H} : (\mathsf{Set} \to \mathsf{Set}) \to \mathsf{Set} \to \mathsf{Set}$$
$$\mathsf{H}\,\mathsf{F}\,\mathsf{X} = \mathsf{X} \times \mathsf{F}\,(\mathsf{F}\,\mathsf{X})$$

2nd order containers

record Cont_2 : Set_1 **where**
  constructor _ $\lhd$ _, _, _
  inductive
  pattern
  field
    $\mathsf{S} : \mathsf{Set}$
    $P_0 : \mathsf{S} \to \mathsf{Set}$
    $R_0 : (\mathsf{s} : \mathsf{S}) \to P_0\,\mathsf{s} \to \mathrm{Cont}_2$
    $P_1 : \mathsf{S} \to \mathsf{Set}$

$$\llbracket\_\rrbracket : \mathrm{Cont}_2 \to (\mathsf{Set} \to \mathsf{Set}) \to \mathsf{Set} \to \mathsf{Set}$$
$$\llbracket\, \mathsf{S} \lhd P_0 , R_0 , P_1 \,\rrbracket\, \mathsf{F}\, \mathsf{X} =$$
$$\Sigma[\, \mathsf{s} \in \mathsf{S} \,]\, ((\mathsf{p} : P_0\,\mathsf{s}) \to \mathsf{F}\, (\llbracket\, R_0\,\mathsf{s}\,\mathsf{p} \,\rrbracket\, \mathsf{F}\,\mathsf{X}))$$
$$\times\ (P_1\,\mathsf{s} \to \mathsf{X})$$

```
H : Cont₂
  -- H F X = X × F (F X)
H = ⊤ ◁ (λ _ → ⊤) ,
  (λ _ _ →    -- H' F X = F X
    ⊤ ◁ (λ _ → ⊤) ,
    (λ _ _ →    -- H'' F X = X
      ⊤ ◁ (λ _ → ⊥) , (λ _ ()) , λ _ → ⊤) ,
    λ _ → ⊤) ,
  λ _ → ⊤
```

Higher containers

data Ty : Set **where**
$\circ$: Ty
$\_ \Rightarrow \_$: Ty $\to$ Ty $\to$ Ty
data Con : Set **where**
$\bullet$: Con
$\_ \triangleright \_$: Con $\to$ Ty $\to$ Con
data Var : Con $\to$ Ty $\to$ Set **where**
vz : Var $(\Gamma \triangleright A)\ A$
vs : Var $\Gamma\ A \to$ Var $(\Gamma \triangleright B)\ A$

```
data Nf : Con → Ty → Set
record Ne (Γ : Con) (B : Ty) : Set
data Sp : Con → Ty → Ty → Set
data Nf where
  lam : Nf (Γ ▷ A) B → Nf Γ (A ⇒ B)
  ne : Ne Γ ∘ → Nf Γ ∘
record Ne Γ B where
  inductive
  field
    S : Set
    P : S → Var Γ A → Set
    R : (s : S) (x : Var Γ A) (p : P s x) → Sp Γ A B
data Sp where
  ε   : Sp Γ A A
  _,_ : Nf Γ A → Sp Γ B C → Sp Γ (A ⇒ B) C
```

Semantics

$$
\begin{aligned}
&[\![\_]\!]_{\mathrm{T}} : \mathsf{Ty} \to \mathsf{Set} \\
&[\![\circ]\!]_{\mathrm{T}} = \mathsf{Set} \\
&[\![\mathsf{A} \Rightarrow \mathsf{B}]\!]_{\mathrm{T}} = [\![\mathsf{A}]\!]_{\mathrm{T}} \to [\![\mathsf{B}]\!]_{\mathrm{T}} \\
&[\![\_]\!]_{\mathrm{C}} : \mathsf{Con} \to \mathsf{Set} \\
&[\![\bullet]\!]_{\mathrm{C}} = \top \\
&[\![\Gamma \rhd \mathsf{A}]\!]_{\mathrm{C}} = [\![\Gamma]\!]_{\mathrm{C}} \times [\![\mathsf{A}]\!]_{\mathrm{T}} \\
&[\![\_]\!]_{\mathrm{v}} : \mathsf{Var}\,\Gamma\,\mathsf{A} \to [\![\Gamma]\!]_{\mathrm{C}} \to [\![\mathsf{A}]\!]_{\mathrm{T}} \\
&[\![\mathsf{vz}]\!]_{\mathrm{v}}\,(\gamma\,,\mathsf{a}) = \mathsf{a} \\
&[\![\mathsf{vs}\,\mathsf{x}]\!]_{\mathrm{v}}\,(\gamma\,,\mathsf{a}) = [\![\mathsf{x}]\!]_{\mathrm{v}}\,\gamma
\end{aligned}
$$

$$
\begin{aligned}
&[\![\_]\!]_{\mathrm{nf}} : \mathsf{Nf}\,\Gamma\,\mathsf{A} \to [\![\,\Gamma\,]\!]_{\mathrm{C}} \to [\![\,\mathsf{A}\,]\!]_{\mathrm{T}} \\
&[\![\_]\!]_{\mathrm{ne}} : \mathsf{Ne}\,\Gamma\,\circ \to [\![\,\Gamma\,]\!]_{\mathrm{C}} \to \mathsf{Set} \\
&[\![\_]\!]_{\mathrm{sp}} : \mathsf{Sp}\,\Gamma\,\mathsf{A}\,\mathsf{B} \to [\![\,\Gamma\,]\!]_{\mathrm{C}} \to [\![\,\mathsf{A}\,]\!]_{\mathrm{T}} \to [\![\,\mathsf{B}\,]\!]_{\mathrm{T}}
\end{aligned}
$$

$$
\begin{aligned}
&[\![\,\mathsf{lam}\,\mathsf{x}\,]\!]_{\mathrm{nf}}\,\gamma\,\mathsf{a} = [\![\,\mathsf{x}\,]\!]_{\mathrm{nf}}\,(\gamma\,,\mathsf{a}) \\
&[\![\,\mathsf{ne}\,\mathsf{x}\,]\!]_{\mathrm{nf}}\,\gamma = [\![\,\mathsf{x}\,]\!]_{\mathrm{ne}}\,\gamma
\end{aligned}
$$

$$
\begin{aligned}
&[\![\_]\!]_{\mathrm{ne}}\,\{\Gamma\}\,\textbf{record}\,\{\mathsf{S} = \mathsf{S}; \mathsf{P} = \mathsf{P}; \mathsf{R} = \mathsf{R}\}\,\gamma = \\
&\quad \Sigma[\,\mathsf{s} \in \mathsf{S}\,]\,(\{\mathsf{A} : \mathsf{Ty}\}\,(\mathsf{x} : \mathsf{Var}\,\Gamma\,\mathsf{A}) \\
&\qquad\qquad (\mathsf{p} : \mathsf{P}\,\mathsf{s}\,\mathsf{x}) \to [\![\,\mathsf{R}\,\mathsf{s}\,\mathsf{x}\,\mathsf{p}\,]\!]_{\mathrm{sp}}\,\gamma\,([\![\,\mathsf{x}\,]\!]_{\mathrm{v}}\,\gamma))
\end{aligned}
$$

$$
\begin{aligned}
&[\![\,\epsilon\,]\!]_{\mathrm{sp}}\,\gamma\,\mathsf{a} = \mathsf{a} \\
&[\![\,\mathsf{n}\,,\mathsf{ns}\,]\!]_{\mathrm{sp}}\,\gamma\,\mathsf{f} = [\![\,\mathsf{ns}\,]\!]_{\mathrm{sp}}\,\gamma\,(\mathsf{f}\,([\![\,\mathsf{n}\,]\!]_{\mathrm{nf}}\,\gamma))
\end{aligned}
$$

Results?

- Interpretation of simply typed λ-calculus
  Using ideas from heredetary substitutions
  Still need to check equations.
- Functoriality
  Need to interpret using heredetary higher functors
  In progress
- Morphisms ?
  Seems to work for Cont_2, generalize?
- Initial algebras, terminal coalgebras

$$\mu : \mathsf{HCont}\ (\mathsf{A} \Rightarrow \mathsf{A}) \Rightarrow \mathsf{A}$$
$$\nu : \mathsf{HCont}\ (\mathsf{A} \Rightarrow \mathsf{A}) \Rightarrow \mathsf{A}$$

  Some issues (restrict universe?)
- Translation from TLCA01
  sketch on paper
  proof?

Heredetary functors

$$
\begin{array}{l}
\textbf{record}\ \mathsf{Cat}\ (\mathsf{Obj}\ :\ \mathsf{Set})\ :\ \mathsf{Set} \\
\textbf{record}\ \mathsf{Func}\ (\mathsf{C}\ :\ \mathsf{Cat}\ \mathsf{X})\ (\mathsf{D}\ :\ \mathsf{Cat}\ \mathsf{Y})\ (\mathsf{F}\ :\ \mathsf{X}\ \to\ \mathsf{Y})\ :\ \mathsf{Set} \\
[\![\_]\!]_{\mathrm{F}}\ :\ (\mathsf{A}\ :\ \mathsf{Ty})\ \to\ [\![\ \mathsf{A}\ ]\!]_{\mathrm{T}}\ \to\ \mathsf{Set} \\
[\![\_]\!]_{\mathrm{C}}\ :\ (\mathsf{A}\ :\ \mathsf{Ty})\ \to\ \mathsf{Cat}\ (\Sigma\ [\![\ \mathsf{A}\ ]\!]_{\mathrm{T}}\ [\![\ \mathsf{A}\ ]\!]_{\mathrm{F}}) \\
[\![\ \mathsf{set}\ ]\!]_{\mathrm{F}}\ \mathsf{X}\ =\ \top \\
[\![\ \mathsf{A}\ \Rightarrow\ \mathsf{B}\ ]\!]_{\mathrm{F}}\ \mathsf{H}\ = \\
\quad \Sigma[\ \mathsf{HH}\ \in\ ((\mathsf{F}\ :\ [\![\ \mathsf{A}\ ]\!]_{\mathrm{T}})\ \to\ [\![\ \mathsf{A}\ ]\!]_{\mathrm{F}}\ \mathsf{F}\ \to\ [\![\ \mathsf{B}\ ]\!]_{\mathrm{F}}\ (\mathsf{H}\ \mathsf{F}))\ ] \\
\qquad \mathsf{Func}\ [\![\ \mathsf{A}\ ]\!]_{\mathrm{C}}\ [\![\ \mathsf{B}\ ]\!]_{\mathrm{C}}\ (\lambda\ (\mathsf{F}\ ,\ \mathsf{FF})\ \to\ \mathsf{H}\ \mathsf{F}\ ,\ \mathsf{HH}\ \mathsf{F}\ \mathsf{FF})
\end{array}
$$

Example for μ

$\mathsf{H\,F\,X} = 1 + \mathsf{X} \times \mathsf{F\,(F\,X)}$
construct μ H

$$
\begin{aligned}
&\textbf{data}\ \mathsf{S} : \mathsf{Set}\\
&\mathsf{P} : \mathsf{S} \to \mathsf{Set}\\
&\textbf{data}\ \mathsf{S}\ \textbf{where}\\
&\quad \mathrm{s}_\perp : \mathsf{S}\\
&\quad \mathsf{node} : (\mathsf{s} : \mathsf{S}) \to (\mathsf{P\,s} \to \mathsf{S}) \to \mathsf{S}\\
&\mathsf{P}\,\mathrm{s}_\perp = \perp\\
&\mathsf{P\,(node\,s\,f)} = \mathsf{Maybe}\,(\Sigma[\,\mathsf{p} \in \mathsf{P\,s}\,]\,(\mathsf{P\,(f\,p)}))
\end{aligned}
$$

Example for ν

H F X $=$ X $\times$ F (F X)
construct nu H (we know it is 1 $\lhd$ BT)

```
record S : Set
data P : S → Set
record S where
  coinductive
  field
    s : S
    f : P s → S
data P where
  hd : (sf : S) → P sf
  tl : (sf : S) (p : P (S.s sf)) (q : P (S.f sf p)) → P sf
```

Problem : not strictly positive!