

Towards Type Theory with Continuity

Thorsten Altenkirch

School of Computer Science
University of Nottingham

March 17, 2008

- Extensional Type Theory with W and Quotient types
- In category speak: LCC pretopos with W predicative topos
- **Prop** = sets with at most one inhabitant.
- Define $\exists a : A.P a = [\Sigma a : A.P a]$ (bracket types).

$$\frac{A : \mathbf{Set}}{[A] : \mathbf{Prop}} \quad [A] = A / (\lambda x, y. \text{true})$$

- Logic, Set Theory and Programming Language
- As a Programming Language: purely functional, **total** !
- How to capture *real world* programming, i.e. *computational effects*?

Effects as Monads

Functional Programming

Effects = Monads (Moggi, Wadler)

Monad

$M : \mathbf{Set} \rightarrow \mathbf{Set}$

$$\frac{a : A}{\eta a : M A} \quad \frac{m : M A \quad f : A \rightarrow M B}{a \gg= f : M B}$$

+ equations ($A \rightarrow M B$ is a category)

Monads

Error $M_E X = 1 + X$

State $M_S X = S \rightarrow S \times X$

Cont. $M_C X = (X \rightarrow R) \rightarrow R$

Kleisli category

$A \rightarrow M B$ = effectful computations.

Delay M_D

Partial $M_P X = (M_D X) / \sim$

Idea

Partial functions from A to $B = A \rightarrow M_P B$.

Based on published work by Venanzio Capretta and unpublished work with Tarmo Uuustalu and Venanzio.

$$\text{codata} \frac{A : \mathbf{Set}}{M_D A : \mathbf{Set}} \quad \text{where} \quad \frac{a : A}{N a : M_D A} \quad \frac{d : M_D A}{L d : M_D A}$$

Divergent computation

$$\perp = M_D A$$

$$\perp = L \perp$$

Monad structure

$$\eta_D a = N a$$

$$N a \gg= f = f a$$

$$L d \gg= f = L(d \gg= f)$$

$$\mathbf{data} \frac{d : M_D A \quad a : A}{d \downarrow a : \mathbf{Prop}} \quad \text{where} \quad \frac{}{N a \downarrow a} \quad \frac{d \downarrow a}{L d \downarrow a}$$

Termination order

$$\begin{aligned} d \sqsubseteq d' &= \Pi a : A. d \downarrow a \rightarrow d' \downarrow a \\ d \sim d' &= d \sqsubseteq d' \wedge d' \sqsubseteq d \end{aligned}$$

- $M_P X = (M_D X) / \sim$
- Classically: $M_P X = X + \{\perp\}$
- Inherits monad structure (\gg_D stable under \sim).
- Lift order: $\sqsubseteq: M_D A \rightarrow M_D A \rightarrow \mathbf{Prop}$

How to construct ?

$$\frac{f : (A \rightarrow M_P B) \rightarrow A \rightarrow M_P B}{\text{fix } f : A \rightarrow M_P B}$$

$$\text{fix } f = \bigsqcup (\lambda n. f^n \perp)$$

directed completeness

$$\text{Chain} = \{\vec{d} : \mathbb{N} \rightarrow M_D A \mid \forall n : \mathbb{N}. \vec{d} n \sqsubseteq \vec{d} (n+1)\}$$

$$\frac{\vec{d} : \text{Chain}}{\bigsqcup \vec{d} : M_D A} \quad \bigsqcup \vec{d} = \vec{d} 0 \sqcup \text{L} \bigsqcup \vec{d} \circ (+1)$$

race

$$\begin{aligned} \text{N} a \sqcup d' &= \text{N} a \\ \text{L} d \sqcup \text{N} b &= b \\ \text{L} d \sqcup \text{L} d' &= \text{L}(d \sqcup d') \end{aligned}$$

Continuity?

- \sqcup works on M_P (but not \sqcup).
- ω -CPO structure lifts pointwise to $A \rightarrow M_P B$.
- We need that $f : (A \rightarrow M_P B) \rightarrow A \rightarrow M_P B$ is ω -continuous, i.e.

$$f(\bigsqcup \vec{d}) = \bigsqcup \lambda i. f(\vec{d} i)$$

- We have to prove ω -continuity again and again.
- We cannot define a non-continuous f !

Type Theory with Continuity ?

- How to add continuity to Type Theory?
- Consistent with extensionality.
- Computational (BHK).
- Explained by translation?

1st order continuity

- Consider $(\mathbb{N} \rightarrow \mathbb{N}) \rightarrow \mathbb{N}$
- What are the possible computations of this type?

Eating games (Hancock et al)

data $G : \mathbf{Set}$ where $\frac{n : \mathbb{N}}{Rn : G}$ $\frac{g : \mathbb{N} \rightarrow G}{Gg : G}$

$\frac{g : G}{[[G]] : (\mathbb{N} \rightarrow \mathbb{N}) \rightarrow \mathbb{N}}$ $[[Rn]]h = n$
 $[[Gg]]h = [[g(h0)]]h \circ (+1)$

$\frac{f : (\mathbb{N} \rightarrow \mathbb{N}) \rightarrow \mathbb{N}}{qf : G}$ $[[qf]] = f$
 $q[[g]] = g$

Too intensional!

$$g \sim g' = ([g] = [g'])$$

$$\frac{f : (\mathbb{N} \rightarrow \mathbb{N}) \rightarrow \mathbb{N}}{qf : G / \sim} \quad \begin{array}{l} [qf] = f \\ q[g] = g \end{array}$$

$$\frac{f : (\mathbb{N} \rightarrow \mathbb{N}) \rightarrow \mathbb{N} \quad h : \mathbb{N} \rightarrow \mathbb{N}}{\text{lc } f h : \exists n : \mathbb{N}. \Pi h' : \mathbb{N} \rightarrow \mathbb{N}. (\Pi i < n. h n = h' n) \rightarrow f h = f h'}$$

Derive lc using lc':

$$\frac{g : G \quad h : \mathbb{N} \rightarrow \mathbb{N}}{\text{lc}' g h : \Sigma n : \mathbb{N}. \Pi h' : \mathbb{N} \rightarrow \mathbb{N}. (\Pi i < n. h n = h' n) \rightarrow f h = f h'}$$

Higher order continuity?

What are games for:

$$((\mathbb{N} \rightarrow \mathbb{N}) \rightarrow \mathbb{N}) \rightarrow \mathbb{N}?$$

$$((\mathbb{N} \rightarrow \mathbb{N}) \rightarrow \mathbb{N}) \rightarrow \mathbb{N}$$

$$\simeq \mathbf{G} / \sim \rightarrow \mathbb{N}$$

$$\simeq \{f : \mathbf{G} \rightarrow \mathbb{N} \mid \prod g, g' : \mathbf{G}. g \sim g' \rightarrow f g = f g'\}$$

We need games for:

$$\mathbf{G} \rightarrow \mathbb{N}$$

Higher order games ($G \rightarrow \mathbb{N}$)

$S : \mathbf{Set}$

$Q : S \rightarrow \mathbf{Set}$

$R : \mathbf{Set}$

$n : \prod s : S, q : Q s. R \rightarrow S$

Synek-Petersson trees: $T : S \rightarrow \mathbf{Set}$

data $S : \mathbf{Set}$ **where** $\frac{n : \mathbb{N}}{R n : S} \quad \frac{\vec{s} : S^*}{N \vec{s} : S}$

data $\frac{s : S}{Q s : \mathbf{Set}}$ **where** $\frac{}{H : Q(N \vec{s})} \quad \frac{q : Q \vec{s}_i}{D i q : Q(N \vec{s})}$

data $\frac{}{R : \mathbf{Set}}$ **where** $\frac{n : \mathbb{N}}{R n : R} \quad \frac{}{N : R}$

$$\frac{t : T s}{\llbracket t \rrbracket : \{g : G \mid s < g\} \rightarrow \mathbb{N}}$$

$$t \sim t' = (\llbracket t \rrbracket = \llbracket t' \rrbracket)$$

$$\frac{f : \{g : G \mid s < g\} \rightarrow \mathbb{N}}{qf : (T s) / \sim}$$

We can use T to give an interpretation of a 3rd order type by 1st order games.

- Can we interpret all arithmetic types by 1st order games? (using Synek-Petersson trees).
- Such a construction should give rise to a translation justifying continuity in Type Theory.
- Has this been done in intuitionistic logic?
- Applications to the elimination of extensionality in Observational Type Theory.