Towards guarded recursion in HoTT

Rasmus Ejlers Møgelberg

IT University of Copenhagen

September 25, 2015

Overview

- Guarded recursion applications
  - Computing with streams
  - Modelling programming languages
- Model: the topos of trees
- Intensional models
- Coinduction via guarded recursion
- Problems combining HoTT and GR

Motivation

Motivation 1: Computing with streams

- Which of the following streams are well-defined:

```
zeros = 0::zeros
   xs = xs
   ys = 0::(tail(ys))
 nats = 0::(map (+1) nats)
```

Capturing productivity in types

- Introduce modal operator $\blacktriangleright$

$$\begin{aligned} \mathtt{S(int)}\ \mathtt{=}\ \mu\mathtt{X.int} \times \blacktriangleright \mathtt{X} \\ \mathtt{hd}\colon \mathtt{S(int)} \to \mathtt{int} \\ \mathtt{tail}\colon \mathtt{S(int)} \to \blacktriangleright \mathtt{S(int)} \\ \mathtt{cons}\colon \mathtt{int} \times \blacktriangleright \mathtt{S(int)} \to \mathtt{S(int)} \end{aligned}$$

- Fixed points

$$\mathtt{fix}\colon (\blacktriangleright \mathtt{S(int)} \to \mathtt{S(int)}) \to \mathtt{S(int)}$$

$$\mathtt{zeros}\ \mathtt{=}\ \mathtt{fix}(\lambda \mathtt{xs.0::xs})$$

Capturing productivity in types

- $\blacktriangleright$ is an applicative functor

$$\mathtt{next} : \mathtt{X} \to \blacktriangleright \mathtt{X}$$

$$\circledast : \blacktriangleright (\mathtt{X} \to \mathtt{Y}) \to \blacktriangleright \mathtt{X} \to \blacktriangleright \mathtt{Y}$$

- Typing `nats`

$$\mathtt{nats\ =\ fix(}\lambda\mathtt{xs.0::(next(map\ (+1))} \circledast \mathtt{xs))}$$

- Fixed point property

$$\mathtt{fix} : \big(\blacktriangleright \mathtt{S(int)} \to \mathtt{S(int)}\big) \to \mathtt{S(int)}$$

$$\mathtt{fix(f)} = \mathtt{f(next(fix(f)))}$$

Motivation 2: Modelling higher-order store

- Would like to solve (but can not)

$$\mathcal{W} = N \to_{\mathrm{fin}} \mathcal{T} \qquad\qquad \mathcal{T} = \mathcal{W} \to_{\mathrm{mon}} \mathcal{P}(\mathrm{Value})$$

- Suffices to solve this equation

$$\widehat{\mathcal{T}} \cong \blacktriangleright((N \to_{\mathrm{fin}} \widehat{\mathcal{T}}) \to_{\mathrm{mon}} \mathcal{P}(\mathrm{Value}))$$

- Can model higher-order store in expressive type theory with guarded recursion
- Synthetic presentation of step-indexed model!
- (Birkedal, M. et al LICS 2011)

Synthetic guarded domain theory

- Lifting monad

$$LX \cong X + \blacktriangleright LX$$

- Model of PCF in type theory with guarded recursion

$$Y\ M \Downarrow^{k+1} v =_{\text{def}} \blacktriangleright (M(Y\ M) \Downarrow^{k} v)$$

- Proved (intensional) computational adequacy
- (Paviotti, M, Birkedal, MFPS 2015)

The topos of trees

The topos of trees ($\mathbf{Set}^{\omega^{\mathrm{op}}}$)

- Objects

$$X(1) \xleftarrow{r_1} X(2) \xleftarrow{r_2} X(3) \longleftarrow \ldots$$

- Example: object of streams of integers `S(int)`

$$\mathbb{Z} \xleftarrow{\pi} \mathbb{Z}^2 \xleftarrow{\pi} \mathbb{Z}^3 \xleftarrow{\pi} \ldots$$

The topos of trees ($\mathbf{Set}^{\omega^{\mathrm{op}}}$)

- Objects

$$X(1) \xleftarrow{r_1} X(2) \xleftarrow{r_2} X(3) \longleftarrow \dots$$

- Example: object of streams of integers `S(int)`

$$\mathbb{Z} \xleftarrow{\ \pi\ } \mathbb{Z}^2 \xleftarrow{\ \pi\ } \mathbb{Z}^3 \xleftarrow{\ \pi\ } \dots$$

- Define $\blacktriangleright X$

$$\{*\} \longleftarrow X(1) \longleftarrow X(2) \longleftarrow \dots$$

- Note that `S(int)` $\cong \mathbb{Z} \times \blacktriangleright$ `S(int)`:

$$\mathbb{Z} \times 1 \xleftarrow{\mathbb{Z}\times !} \mathbb{Z} \times \mathbb{Z} \xleftarrow{\mathbb{Z} \times \pi} \mathbb{Z} \times \mathbb{Z}^2 \xleftarrow{\mathbb{Z} \times \pi} \dots$$

Construction of fixed points

- Given $f : \blacktriangleright X \to X$:

$$\begin{tikzcd}
\{*\} & X(1) \arrow[l] & X(2) \arrow[l, "r_1"'] & \dots \arrow[l] \\
X(1) \arrow[from=u, "f_1"'] & X(2) \arrow[l, "r_1"'] \arrow[from=u, "f_2"'] & X(3) \arrow[l, "r_2"'] \arrow[from=u, "f_3"'] & \dots \arrow[l]
\end{tikzcd}$$

- Construct $\mathrm{fix}_X(f) : 1 \to X$:

$$\begin{tikzcd}
1 \arrow[d, "f_1"] & 1 \arrow[l] \arrow[d, "f_2 \circ f_1"] & 1 \arrow[l] \arrow[d, "f_3 \circ f_2 \circ f_1"] & \dots \arrow[l] \\
X(1) & X(2) \arrow[l, "r_1"'] & X(3) \arrow[l, "r_2"'] & \dots \arrow[l]
\end{tikzcd}$$

- Fixed points are unique

Guarded recursive types as fixed points

- Universe closed under $\blacktriangleright$

$$\frac{\Gamma \vdash A : U}{\Gamma \vdash \triangleright A : U}$$

$$\mathrm{El}(\triangleright(A)) = \blacktriangleright \mathrm{El}(A)$$

Guarded recursive types as fixed points

- Universe closed under $\blacktriangleright$

$$\frac{\Gamma \vdash A : \blacktriangleright U}{\Gamma \vdash \triangleright A : U}$$

$$\mathrm{El}(\triangleright(\mathrm{next}(A))) = \blacktriangleright \mathrm{El}(A)$$

Guarded recursive types as fixed points

- Universe closed under $\blacktriangleright$

$$\frac{\Gamma \vdash A : \blacktriangleright U}{\Gamma \vdash \triangleright A : U}$$

$$\mathrm{El}(\triangleright(\mathrm{next}(A))) = \blacktriangleright \mathrm{El}(A)$$

- Type of streams as fixed point for universe map

$$\mathtt{S(int)} = \mathrm{fix}(\lambda X : \blacktriangleright U. \mathbb{Z} \times \triangleright X)$$

- Then

$$\begin{aligned} \mathrm{El}(\mathtt{S(int)}) &= \mathrm{El}(\mathbb{Z} \times \triangleright(\mathrm{next}(\mathtt{S(int)}))) \\ &= \mathbb{Z} \times \blacktriangleright \mathrm{El}(\mathtt{S(int)}) \end{aligned}$$

Guarded recursive types as fixed points

- Universe closed under $\blacktriangleright$

$$\frac{\Gamma \vdash A : \blacktriangleright U}{\Gamma \vdash \triangleright A : U}$$

$$\mathrm{El}(\triangleright(\mathrm{next}(A))) = \blacktriangleright \mathrm{El}(A)$$

- Type of streams as fixed point for universe map

$$\mathtt{S(int)} = \mathrm{fix}(\lambda X : \blacktriangleright U . \mathbb{Z} \times \triangleright X)$$

- Then

$$\begin{aligned} \mathrm{El}(\mathtt{S(int)}) &= \mathrm{El}(\mathbb{Z} \times \triangleright(\mathrm{next}(\mathtt{S(int)}))) \\ &= \mathbb{Z} \times \blacktriangleright \mathrm{El}(\mathtt{S(int)}) \end{aligned}$$

- Model using Hofmann-Streicher universe construction

Intensional models

Intensional models

- **Theorem** (Shulman). If $\mathbb{C}$ is a model of intensional type theory, so is $\mathbb{C}^{\omega^{\mathrm{op}}}$. Univalent universes in $\mathbb{C}$ lift to univalent universes in $\mathbb{C}^{\omega^{\mathrm{op}}}$ by Hofmann-Streicher

Intensional models

- **Theorem** (Shulman). If $\mathbb{C}$ is a model of intensional type theory, so is $\mathbb{C}^{\omega^{\mathrm{op}}}$. Univalent universes in $\mathbb{C}$ lift to univalent universes in $\mathbb{C}^{\omega^{\mathrm{op}}}$ by Hofmann-Streicher
- Model construction uses Reedy model structure on $\mathbb{C}^{\omega^{\mathrm{op}}}$
- Closed types are sequences

$$A(1) \xleftarrow{r_1^A} A(2) \xleftarrow{r_2^A} A(3) \xleftarrow{r_3^A} A(4) \longleftarrow \ldots$$

  where each r_n^A is a *fibration*
- i.e., closed types modelled as infinite contexts

Uniqueness of fixed points, propositionally

- **Theorem**. $\mathbb{C}^{\omega^{\mathrm{op}}}$ models guarded recursion plus uniqueness of fixed points

$$\frac{\Gamma \vdash f : \blacktriangleright A \to A \quad \Gamma \vdash M : A \quad \Gamma \vdash p : \mathrm{Id}_A(M, f(\mathrm{next}(M)))}{\Gamma \vdash \mathrm{UFP}(p) : \mathrm{Id}_A(M, \mathrm{fix}(f))}$$

- Univalence plus UFP implies

$$(F(\mathrm{next}(A)) \simeq A) \longleftrightarrow (A \simeq \mathrm{fix}(F))$$

- for $F : \blacktriangleright U \to U$
- (Birkedal and M, LICS 2013)

Coinductive types via guarded recursive types

A problem

$$
\begin{aligned}
\mathtt{S(int)} \;\mathtt{=}\; \mu\mathtt{X}.\mathtt{int} \times \blacktriangleright \mathtt{X} \\
\mathtt{hd}\colon \mathtt{S(int)} \to \mathtt{int} \\
\mathtt{tail}\colon \mathtt{S(int)} \to \blacktriangleright \mathtt{S(int)} \\
\mathtt{cons}\colon \mathtt{int} \times \blacktriangleright \mathtt{S(int)} \to \mathtt{S(int)} \\
\mathtt{next}\colon \mathtt{X} \to \blacktriangleright \mathtt{X} \\
\circledast\colon \blacktriangleright (\mathtt{X} \to \mathtt{Y}) \to \blacktriangleright \mathtt{X} \to \blacktriangleright \mathtt{Y}
\end{aligned}
$$

- Computing the second element

$$\mathtt{snd} \;\mathtt{=}\; (\lambda \mathtt{xs}.\mathtt{next(hd)} \circledast \mathtt{tail(xs)})\colon \mathtt{S(int)} \to \blacktriangleright \mathtt{int}$$

- Can not get rid of $\blacktriangleright$!

Guarded recursion vs coinduction in model

- All maps $\texttt{f}: \texttt{S(int)} \to \texttt{S(int)}$ are *causal*

$$\begin{tikzcd}
\mathbb{Z} \arrow[d, "f_1"'] & \mathbb{Z}^2 \arrow[l, "\pi"'] \arrow[d, "f_2"'] & \mathbb{Z}^3 \arrow[l, "\pi"'] \arrow[d, "f_3"'] & \dots \arrow[l, "\pi"'] \\
& & & \dots \\
\mathbb{Z} & \mathbb{Z}^2 \arrow[l, "\pi"'] & \mathbb{Z}^3 \arrow[l, "\pi"'] & \dots \arrow[l, "\pi"']
\end{tikzcd}$$

- Observation: Limit of guarded recursive streams is *set* of real (coinductive) streams!

$$\mathbb{Z} \xleftarrow{\pi} \mathbb{Z}^2 \xleftarrow{\pi} \mathbb{Z}^3 \xleftarrow{\pi} \dots$$

Syntax: multiple clocks

- Idea originally due to Atkey and McBride
- Clock variable context $\Delta = \kappa_1, \ldots, \kappa_n$

$$\frac{\Gamma \vdash_\Delta A : \mathrm{Type} \qquad \vdash_\Delta \kappa}{\Gamma \vdash_\Delta \overset{\kappa}{\blacktriangleright} A : \mathrm{Type}}$$

$$\mathtt{fix}^\kappa : (\overset{\kappa}{\blacktriangleright} X \to X) \to X$$

- etc

Universal quantification over clocks

$$\frac{\Gamma \vdash_{\Delta,\kappa} A : \mathsf{Type} \qquad \kappa \notin \mathsf{fc}(\Gamma)}{\Gamma \vdash_{\Delta} \forall\kappa.A : \mathsf{Type}}$$

$$\frac{\Gamma \vdash_{\Delta,\kappa} t : A \qquad \kappa \notin \mathsf{fc}(\Gamma)}{\Gamma \vdash_{\Delta} \Lambda\kappa.t : \forall\kappa.A}$$

$$\frac{\Gamma \vdash_{\Delta} t : \forall\kappa.A \qquad \vdash_{\Delta} \kappa'}{\Gamma \vdash_{\Delta} t[\kappa'] : A[\kappa'/\kappa]}$$

- Clock quantification is right adjoint to clock weakening

An extensional model (Bizjak and M, MFPS 2015)

- Type in context $\Delta = \emptyset$ is a set
- Type in context $\Delta = \kappa$ is an object in the topos of trees

$$X(1) \xleftarrow{r_1} X(2) \xleftarrow{r_2} X(3) \longleftarrow \ldots$$

- Type in context $\Delta = \kappa, \kappa'$ is a diagram of form

$$\begin{tikzcd}
\vdots \arrow[d] & \vdots \arrow[d] & \vdots \arrow[d] & & \vdots \arrow[d] \\
X(1,3) \arrow[d] & X(2,3) \arrow[l] \arrow[d] & X(3,3) \arrow[l] \arrow[d] \arrow[rr, bend right] & \cdots \arrow[l] & X(3) \arrow[d] \\
X(1,2) \arrow[d] & X(2,2) \arrow[l] \arrow[d] \arrow[rrr, bend right] & X(3,2) \arrow[l] \arrow[d, dashed] & \cdots \arrow[l] & X(2) \arrow[d] \\
X(1,1) \arrow[rrrr, bend right] & X(2,1) \arrow[l] & X(3,1) \arrow[l] & \cdots \arrow[l] & X(1)
\end{tikzcd}$$

An extensional model (Bizjak and M, MFPS 2015)

- Interpretation of $\blacktriangleright^{\kappa} \blacktriangleright^{\kappa'} \texttt{int}$

$$
\begin{tikzcd}
\vdots \arrow[d] & \vdots \arrow[d] & \vdots \arrow[d] & & \vdots \arrow[d] \\
1 \arrow[d] & \mathbb{Z} \arrow[l] \arrow[d] & \mathbb{Z} \arrow[l] \arrow[d] \arrow[rr, bend right] & \cdots \arrow[l] & \mathbb{Z} \arrow[d] \\
1 \arrow[d] & \mathbb{Z} \arrow[l] \arrow[d] \arrow[rrr, bend right] & \mathbb{Z} \arrow[l] \arrow[d] & \cdots \arrow[l] & 1 \arrow[d] \\
1 \arrow[rrrr, bend right] & 1 \arrow[l] & 1 \arrow[l] & \cdots \arrow[l] & 1
\end{tikzcd}
$$

Universes

- Model forces universes to be indexed by sets of clocks

$$\frac{\Delta' \subseteq \Delta}{\Gamma \vdash_\Delta \mathrm{U}_{\Delta'} : \mathsf{Type}}$$

- E.g. $[\![\vdash_\kappa \mathrm{U}_\emptyset : \mathsf{Type}]\!]$ universe of small constant presheaves in $\mathbf{Set}^{\omega^{\mathrm{op}}}$,
- and $[\![\vdash_\kappa \mathrm{U}_\kappa : \mathsf{Type}]\!]$ universe of all small presheaves in $\mathbf{Set}^{\omega^{\mathrm{op}}}$

Universes

- Model forces universes to be indexed by sets of clocks

$$\frac{\Delta' \subseteq \Delta}{\Gamma \vdash_{\Delta} \mathrm{U}_{\Delta'} : \mathsf{Type}}$$

- E.g. $[\![\vdash_{\kappa} \mathrm{U}_{\emptyset} : \mathsf{Type}]\!]$ universe of small constant presheaves in $\mathbf{Set}^{\omega^{\mathrm{op}}}$,
- and $[\![\vdash_{\kappa} \mathrm{U}_{\kappa} : \mathsf{Type}]\!]$ universe of all small presheaves in $\mathbf{Set}^{\omega^{\mathrm{op}}}$
- Implicit inclusion of universes

$$\frac{\Delta'' \subseteq \Delta' \subseteq \Delta \qquad \Gamma \vdash_{\Delta} A : \mathrm{U}_{\Delta''}}{\Gamma \vdash_{\Delta} A : \mathrm{U}_{\Delta'}}$$

- Requires universe inclusion to commute with operations on universe

Universes in $\mathbf{Set}^{\omega^{\mathrm{op}}}$

- Assume given universe U in **Set**
- Construct universes $V_\emptyset$, V_κ in $\mathbf{Set}^{\omega^{\mathrm{op}}}$

$$\begin{tikzcd}
V_\emptyset \arrow[d, "i"'] & = & U \arrow[d, "i_1"'] & U \arrow[l] \arrow[d, "i_2"'] & U \arrow[l] \arrow[d, "i_3"'] & U \arrow[l] \arrow[d, "i_4"'] & \dots \arrow[l] \\
V_\kappa & = & V_\kappa(1) & V_\kappa(2) \arrow[l] & V_\kappa(3) \arrow[l] & V_\kappa(4) \arrow[l] & \dots \arrow[l]
\end{tikzcd}$$

- Where $V_\kappa(1) = U$

$$V_\kappa(n+1) = \{X_1 \xleftarrow{f_1} X_2 \dots \xleftarrow{f_n} X_{n+1} \mid \forall i. X_i \in U\}$$

$$i_n(X) = (X \xleftarrow{id} X \dots \xleftarrow{id} X)$$

- (V_κ is Hofmann-Streicher universe)

Universes in $\mathbf{Set}^{\omega^{\mathrm{op}}}$

$$\begin{tikzcd}
V_\emptyset \arrow[d, "i"'] & = & U \arrow[d, "i_1"'] & U \arrow[l] \arrow[d, "i_2"'] & U \arrow[l] \arrow[d, "i_3"'] & U \arrow[l] \arrow[d, "i_4"'] & \dots \arrow[l] \\
V_\kappa & = & V_\kappa(1) & V_\kappa(2) \arrow[l] & V_\kappa(3) \arrow[l] & V_\kappa(4) \arrow[l] & \dots \arrow[l]
\end{tikzcd}$$

- Problem: i does not commute with forming function spaces.
- Solution: Quotient each $V_\kappa(n)$ by equivalence classes ensuring uniqueness of codes
- Requires choice in meta logic
- Will probably not work for intensional models

Conclusions

- Guarded recursive types useful for
  - modelling advanced programming languages: higher order store, non-determinism, concurrency etc
  - computing with streams (encoding productivity in types)
- Would like to combine this with HoTT
  - Expressive type theory for formalising programming language models
  - Better treatment of coinductive types in HoTT
- Topos of trees model exists in intensional variant
- Not so clear how to model clocks in intensional type theory

Thanks!