Partiality, Revisited

The Partiality Monad as a Quotient Inductive-Inductive Type

Thorsten Altenkirch[1] Nils Anders Danielsson[2]
Nicolai Kraus[1]

[1]University of Nottingham [2]University of Gothenburg

FoSSaCS, 26 April 2017

Partiality

Task: Given $f : \mathbb{N} \to \mathbb{N}$, find $n : \mathbb{N}$ such that $f(n) = 0$.

In many languages (e.g. Haskell):
easy to write such a function $(\mathbb{N} \to \mathbb{N}) \to \mathbb{N}$

Our setting: intensional **Martin-Löf type theory** (e.g. Agda)

- formal system with Σ-, Π-, identity types, ...
- can be used for programming
- potential foundation of mathematics

All functions are total!

Partiality

Our goal: A *monad*

$$M : \mathsf{Type} \to \mathsf{Type}$$

such that $M(A)$ is a "type of partial elements" in MLTT.

Attempt: *Maybe* monad, $M(A) :\equiv \mathbf{1} + A$.
Not good, "too decidable":
cannot construct suitable function $(\mathbb{N} \to \mathbb{N}) \to \mathbf{1} + \mathbb{N}$.

Attempt: $M(A) :\equiv \Sigma_{Q:\mathsf{Prop}} (Q \to A)$.
(Cf. Escardó-Knapp 2017)
Here not good, "too undecidable".

Our goal: something "semidecidable".

Delay monad

Better attempt: **Delay Monad** (Capretta 2005).
$D(A)$ is the coinductive type generated by

- $\mathsf{now} : A \to D(A)$
- $\mathsf{later} : D(A) \to D(A)$.

Equivalent representation: functions $\mathbb{N} \to (\mathbf{1} + A)$ which become constant once they are $\mathsf{inr}(a)$.

Back to the problem "find a zero":
Yes, we can define a function $(\mathbb{N} \to \mathbb{N}) \to D(\mathbb{N})$.

But $D(A)$ is very intensional:
$\mathsf{later}(\mathsf{now}(a)) \neq \mathsf{later}(\mathsf{later}(\mathsf{now}(a)))$.
Our goal: more extensionality.

Delay monad, quotiented

Weak bisimilarity: binary relation $\approx$ on $D(A)$.
Intuition: $x \approx y$ iff x and y become equal after removing some "laters", thus: $\mathsf{later}(\mathsf{now}(a)) \approx \mathsf{later}(\mathsf{later}(\mathsf{now}(a)))$.

Chapman, Uustalu, Veltri:
Quotienting the delay monad by weak bisimilarity (2015).

Use $D(A)/\approx$ (quotient as introduced by Hofmann).

$D(\_)/\approx$ is a monad on Type assuming *countable choice*.

Countable choice: $\Pi_{n:\mathbb{N}} \|A(n)\| \to \|\Pi_{n:\mathbb{N}} A(n)\|$,
"for every n, there exists $A(n)$" → "there exists a function giving $A(n)$ for every n".

Our goal: avoiding choice.

Quotient inductive-inductive types

We use a combination of two concepts:

- *higher inductive types* from homotopy type theory: inductive types can have constructors for equalities
- *induction-induction*

This combination is also used in the HoTT book to define the Cauchy reals.

- Caveat: computational interpretation conjectured, but still experimental
- Only need special case (*quotient inductive-inductive types*), examined by Dijkstra (2016).

Partiality monad, the construction

Define type (set) $A_\perp$ and $\sqsubseteq : A_\perp \to A_\perp \to \mathsf{Prop}$ simultaneously:

Inductive type (set) $A_\perp$ with constructors:

$$\begin{aligned}
\eta &: A \to A_\perp \\
\perp &: A_\perp \\
\sqcup &: \left(\textstyle\sum_{s:\mathbb{N}\to A_\perp} \prod_{n:\mathbb{N}} s_n \sqsubseteq s_{n+1}\right) \to A_\perp \\
\alpha &: \textstyle\prod_{x,y:A_\perp} x \sqsubseteq y \to y \sqsubseteq x \to x = y
\end{aligned}$$

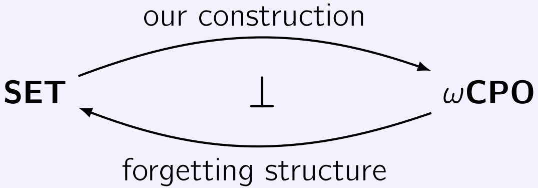
Inductive relation $\sqsubseteq$ given by the rules:

$$\frac{}{x \sqsubseteq x} \qquad \frac{x \sqsubseteq y \qquad y \sqsubseteq z}{x \sqsubseteq z} \qquad \frac{}{\perp \sqsubseteq x}$$

$$\frac{}{\prod_{n:\mathbb{N}} s_n \sqsubseteq \sqcup(s,p)} \qquad \frac{\prod_{n:\mathbb{N}} s_n \sqsubseteq x}{\sqcup(s,p) \sqsubseteq x}$$

Further characterisations

We get an adjunction:



Note: categories can be defined internally:

- **SET** – types (actually *sets*)
- ω**CPO** – types (sets) with structure making them ω-complete partial orders

Connection between the constructions

Assuming countable choice, our construction is equivalent to Chapman-Uustalu-Veltri's:

$$A_\perp \simeq D(A)/{\approx}$$

Why do we need countable choice?

- ✓ $w : D(A) \to A_\perp$
- ✓ w preserves weak bisimilarity ($\approx$)
- ✓ $(w(d) = w(e)) \to (d \approx e)$
- ! surjectivity: $\Pi_{x:A_\perp} \left\| \Sigma_{d:D(A)} w(d) = x \right\|$
  Induction on x; case $x \equiv \sqcup s$ needs countable choice

Final words

Some applications:

- non-terminating functions as fixed points, e.g. $(\mathbb{N} \to \mathbb{N}) \to \mathbb{N}_\bot$
- functions from the Cauchy reals, developed further by Gilbert (2017)
- topology with $\mathbf{1}_\bot$, ...

We have formalised this in Agda.

Take home message:
Constructing an inductive type simultaneously with its equalities is also useful "outside homotopy type theory".

Thank you for your attention!