Continuity in constructive mathematics using Agda

Martín Hötzel Escardó

University of Birmingham, UK

Autumn school *Proof and Computation*, 16-22 September 2018, Fischbachau

Organizers request

"give a 3 hour tutorial on continuity in constructive analysis (using Agda)".

I take the liberty to move the emphasis from analysis to Agda.

Continuity in constructive mathematics using Agda

1. Continuity = finite amounts of output depend only on finite amounts of input.
   - Example. To know a finite decimal approximation of $f(x)$ for a continuous function $f : \mathbb{R} \to \mathbb{R}$, it is enough to know a finite decimal approximation of x.
   - This is equivalent to the usual $\epsilon - \delta$ notion of continuity for the function f.
2. Constructive = implicit computational content.

   Knowledge comes with implicit algorithms.
3. Agda = computer implementation of a Martin-Löf type theory. u
   - Programming language (computes by evaluating terms to normal form).

     Dependently typed functional programming language.
   - So-called proof assistant.

     I prefer to see it as a languague for writing definitions, theorems, proofs and constructions.

     Perhaps computer referee would be a more faithful description of what Agda is.

Plan (or maybe wish list)

1. Learn Agda mainly on the fly, after a short introduction.
2. Based on "Continuity of Godel's system T functionals via effectful forcing" (MFPS'2013).
   - Define Gödel's system T in Agda.
   - Define its set-theoretical model in Agda.
   - Define an alternative dialogue model.
   - Establish a so-called logical relation between them.
   - Use this to conclude that the definable functions $(\mathbb{N} \to \mathbb{N}) \to \mathbb{N}$ of the set model are continuous.
   - Run some examples.
   - Internalize the model to get a translation of T into iself, to show that the dialogue tree of such a T-definable function is itself T definable, and hence the modulus of continuity is T-definable too.

Plan

1. Agda.
2. "Continuity of Godel's system T functionals via effectful forcing" (MFPS'2013).
3. "The inconsistency of a Brouwerian continuity principle with the Curry-Howard interpretation" (TLCA'2015).
   - Prove the negation of "all functions $(\mathbb{N} \to \mathbb{N}) \to \mathbb{N}$ are continuous" in Curry-Howard logic in Agda.
   - Observe that there are (topos) models of MLTT in which all functions are continuous.
   - Understand that there is no contradiction.
   - Show how univalent logic gives a consistent formulation of "all functions are $(\mathbb{N} \to \mathbb{N}) \to \mathbb{N}$ are continuous".
   - Things are radically different with functions $(\mathbb{N} \to 2) \to \mathbb{N}$.

     "all functions $(\mathbb{N} \to 2) \to \mathbb{N}$ are uniformly continuous" is consistent in both Curry-Howard and univalent/topos logic, and the two statements are logically equivalent types (but not isomorphic types).

Plan

1. Agda.
2. "Continuity of Godel's system T functionals via effectful forcing" (MFPS'2013).
3. "The inconsistency of a Brouwerian continuity principle with the Curry-Howard interpretation" (TLCA'2015, with Chuangjie Xu).
4. "A constructive manifestation of the Kleene-Kreisel continuous functionals" (APAL'2016, with Chuangjie Xu)
   - A model of type theory in type theory that validates "all functions $(\mathbb{N} \to 2) \to \mathbb{N}$ are uniformly continuous".
   - Classically equivalently to the Kleene-Kreisel continuous functionals.
   - But constructively developed.
   - Coquand et al. recently showed how to add universes to this model.

Plan summary

1. Agda.
2. "Continuity of Godel's system T functionals via effectful forcing" (MFPS'2013).
3. "The inconsistency of a Brouwerian continuity principle with the Curry-Howard interpretation" (TLCA'2015, with Chuangjie Xu).
4. "A constructive manifestation of the Kleene-Kreisel continuous functionals" (APAL'2016, with Chuangjie Xu)

Probably too ambitious.

Let's see how far we get in three lectures.

Materials

`http://www.cs.bham.ac.uk/~mhe/pc2018/`