

Computational content of coinductive proofs

Helmut Schwichtenberg
(j.w.w. Kenji Miyamoto)

Mathematisches Institut, LMU, München

Isaac Newton Institute, Cambridge, 15. March 2012

A theory of computable functionals, TCF

- ▶ A variant of HA^ω .
- ▶ Variables range over arbitrary partial continuous functionals.
- ▶ Constants for (partial) computable functionals, defined by equations.
- ▶ Inductively and coinductively defined predicates. Totality for ground types inductively defined.
- ▶ Induction $:=$ elimination (or least-fixed-point) axiom for a totality predicate.
- ▶ Coinduction $:=$ greatest-fixed-point for a coinductively defined predicate.
- ▶ Minimal logic: $\rightarrow, \forall$ only. $=$ (Leibniz), $\exists, \vee, \wedge$ (Martin-Löf) inductively defined.
- ▶ $\perp := (\text{False} = \text{True})$. Ex-falso-quodlibet: $\perp \rightarrow A$ provable.

Realizability interpretation

- ▶ Define a formula $t \mathbf{r} A$, for A a formula and t a term in T^+ .
- ▶ **Soundness theorem**:
If M proves A , then $\text{et}(M) \mathbf{r} A$ can be proved.
- ▶ **Decorations** ($\rightarrow^c, \forall^c$ and $\rightarrow^{\text{nc}}, \forall^{\text{nc}}$) for removal of abstract data, and fine-tuning:

$$t \mathbf{r} (A \rightarrow^c B) := \forall_x (x \mathbf{r} A \rightarrow tx \mathbf{r} B),$$

$$t \mathbf{r} (A \rightarrow^{\text{nc}} B) := \forall_x (x \mathbf{r} A \rightarrow t \mathbf{r} B),$$

$$t \mathbf{r} (\forall_x^c A) := \forall_x (tx \mathbf{r} A),$$

$$t \mathbf{r} (\forall_x^{\text{nc}} A) := \forall_x (t \mathbf{r} A).$$

Example: decorating the existential quantifier

- ▶ $\exists_x A$ is inductively defined by the clause

$$\forall_x (A \rightarrow \exists_x A)$$

with least-fixed-point axiom

$$\exists_x A \rightarrow \forall_x (A \rightarrow P) \rightarrow P.$$

- ▶ Decoration leads to variants $\exists^d, \exists^l, \exists^r, \exists^u$ (d for “double”, l for “left”, r for “right” and u for “uniform”).

$$\begin{array}{ll} \forall_x^c (A \rightarrow^c \exists_x^d A), & \exists_x^d A \rightarrow^c \forall_x^c (A \rightarrow^c P) \rightarrow^c P, \\ \forall_x^c (A \rightarrow^{nc} \exists_x^l A), & \exists_x^l A \rightarrow^c \forall_x^c (A \rightarrow^{nc} P) \rightarrow^c P, \\ \forall_x^{nc} (A \rightarrow^c \exists_x^r A), & \exists_x^r A \rightarrow^c \forall_x^{nc} (A \rightarrow^c P) \rightarrow^c P, \\ \forall_x^{nc} (A \rightarrow^{nc} \exists_x^u A), & \exists_x^u A \rightarrow^{nc} \forall_x^{nc} (A \rightarrow^{nc} P) \rightarrow^c P. \end{array}$$

Program extraction

- ▶ We need formalized proofs.
- ▶ Can't take a proof assistant from the shelf: none fits TCF.

Minlog (<http://www.minlog-system.de>)

- ▶ Natural deduction for $\rightarrow, \forall$, plus inductively and coinductively defined predicates.
- ▶ Partial functionals are first class citizens.
- ▶ Allows type and predicate parameters (for abstract developments: groups, fields, reals, ...).

Example: average of two reals

Berger and Seisenberger (2009, 2010).

- ▶ Extraction from a proof dealing with abstract reals.
- ▶ Proof involving coinduction of the proposition that any two reals in $[-1, 1]$ have their average in the same interval.
- ▶ B & S informally extract a Haskell program from this proof, which works with stream representations of reals.

Aim here: discuss formalization of the proof, and machine extract its computational content.

Free algebra $\mathbf{J}$ of intervals

- ▶ $\mathbf{SD} := \{-1, 0, 1\}$ signed digits (or $\{L, M, R\}$).
- ▶ $\mathbf{J}$ free algebra of intervals. Constructors

$\mathbb{I}$ the interval $[-1, 1]$,
 $C: \mathbf{SD} \rightarrow \mathbf{J} \rightarrow \mathbf{J}$ left, middle, right half.

- ▶ $C_1\mathbb{I}$ denotes $[0, 1]$.
- ▶ $C_0\mathbb{I}$ denotes $[-\frac{1}{2}, \frac{1}{2}]$.
- ▶ $C_0(C_{-1}\mathbb{I})$ denotes $[-\frac{1}{2}, 0]$.

$C_{d_0}(C_{d_1} \dots (C_{d_{k-1}}\mathbb{I}) \dots)$ denotes the interval in $[-1, 1]$ whose reals have a signed digit representation starting with $d_0 d_1 \dots d_{k-1}$.

Information systems $\mathbf{C}_\rho$ for partial continuous functionals

- ▶ Types ρ, σ, τ : from algebras ι by $\rho \rightarrow \sigma$.
- ▶ $\mathbf{C}_\rho := (C_\rho, \text{Con}_\rho, \vdash_\rho)$.
- ▶ **Tokens** $a \in C_\rho$ (= atomic pieces of information): constructor trees $Ca_1^*, \dots, a_n^*$ with a_i^* a token or $*$. Example: $S(S^*)$.
- ▶ **Formal neighborhoods** $U \in \text{Con}_\rho$: $\{a_1, \dots, a_n\}$, consistent.
- ▶ **Entailment** $U \vdash_\rho a$.

Ideals $x \in |\mathbf{C}_\rho|$ (“points”, here: partial continuous functionals): consistent deductively closed sets of tokens.

- ▶ We consider ideals $x \in |\mathbf{C}_J|$.

Total and cototal ideals of base type

Generally:

- ▶ **Cototal** ideals x : every token (i.e., constructor tree) $P(*) \in x$ has a “ $\succ_1$ -successor” $P(C*) \in x$.
- ▶ **Total** ideals: the cototal ones with $\succ_1$ well-founded.

Examples:

- ▶ Total ideals of **J**:

$$\mathbb{I}_{\frac{i}{2^k}, k} := \left[\frac{i}{2^k} - \frac{1}{2^k}, \frac{i}{2^k} + \frac{1}{2^k} \right] \quad \text{for } -2^k < i < 2^k.$$

- ▶ Cototal ideals of **J**: reals in $[-1, 1]$, in (non-unique) stream representation using signed digits $-1, 0, 1$.

Inductive and coinductive definitions

- ▶ Inductively define a set I of (abstract) reals, by the clauses

$$I0, \quad \forall_x^{\text{nc}} \forall_d (Ix \rightarrow I \frac{x+d}{2}).$$

Witnesses are intervals (total ideals in $\mathbf{J}$).

- ▶ Coinductively define ${}^{\text{co}}I$, by the (single) clause

$$\forall_x^{\text{nc}} ({}^{\text{co}}Ix \rightarrow x = 0 \vee \exists_y^{\text{r}} \exists_d (x = \frac{y+d}{2} \wedge {}^{\text{co}}Iy)).$$

Witnesses are streams of signed digits (cototal ideals in $\mathbf{J}$).

Inductive and coinductive definitions (continued)

- ▶ We express that I is the **least** predicate satisfying the two clauses, by the least-fixed-point axiom

$$\forall_x^{\text{nc}} (Ix \rightarrow P0 \rightarrow \forall_x^{\text{nc}} \forall_d (Ix \rightarrow Px \rightarrow P^{\frac{x+d}{2}}) \rightarrow Px).$$

- ▶ Dually we express that ${}^{\text{co}}I$ is the **greatest** predicate satisfying its clause, by the greatest-fixed-point (or coinduction) axiom

$$\forall_x^{\text{nc}} (Px \rightarrow \forall_x^{\text{nc}} (Px \rightarrow x = 0 \vee \exists_y^{\text{r}} \exists_d (x = \frac{y+d}{2} \wedge ({}^{\text{co}}Iy \vee Py))) \rightarrow {}^{\text{co}}Ix).$$

Theorem (Average)

$$\forall_{x,y}^{\text{nc}} \exists_{z \in \text{col}}^{\text{r}} z = \frac{x+y}{2}.$$

Proof. Let $i, j, \dots \in \mathbf{SD}_2 := \{-2, -1, 0, 1, 2\}$, and

$$Xz := \exists_{x,y \in \text{col}}^{\text{r}} z = \frac{x+y}{2}, \quad Yz := \exists_{x,y \in \text{col}}^{\text{r}} \exists_i z = \frac{x+y+i}{4}.$$

We show $X \subseteq Y$:

Pf. Wlog $x = \frac{x'+d}{2}$ and $y = \frac{y'+e}{2}$. Then $\frac{x+y}{2} = \frac{x'+y'+d+e}{4} \in Y$.

It suffices to prove $Y \subseteq \text{col}$. Want to apply Gfp-axiom for col to Y :

$$\begin{aligned} & \forall_z^{\text{nc}} (Yz \rightarrow \forall_z^{\text{nc}} (Yz \rightarrow z = 0 \vee \exists_{z'}^{\text{r}} \exists_d (z = \frac{z'+d}{2} \wedge (\text{col}_{y'} \vee Yz')))) \\ & \rightarrow \text{col}_z). \end{aligned}$$

Recall: Want to apply Gfp-axiom for col to Y :

$$\forall_z^{nc} (Yz \rightarrow \forall_z^{nc} (Yz \rightarrow z = 0 \vee \exists_{z'}^r \exists_d (z = \frac{z' + d}{2} \wedge (^{col}ly \vee Yz')))) \\ \rightarrow ^{col}lz).$$

It suffices to prove that Y satisfies col 's clause. We show

$$\forall_{z \in Y}^{nc} \exists_{z' \in Y}^r \exists_d z = \frac{z' + d}{2}.$$

Pf. Let $z = \frac{x+y+i}{4}$. Wlog $x = \frac{x'+d'}{2}$ and $y = \frac{y'+e'}{2}$. Then

$$z = \frac{x' + y' + d' + e' + 2i}{8}.$$

Solve $d' + e' + 2i = j + 4d$. Then

$$z = \frac{x' + y' + j + 4d}{8} = \frac{z' + d}{2} \quad \text{with} \quad z' := \frac{x' + y' + j}{4}.$$

Formalization

- ▶ Use a **type variable** ρ to denote an abstract type of reals.
- ▶ Need functions P (plus) of type $\rho \rightarrow \rho \rightarrow \rho$ for addition, and H (half) of type $\rho \rightarrow \rho$ for division by 2, with axioms

$$(x + k)/2 + l = (x + (k +_Z 2l))/2,$$

$$(x + k)/4 + l = (x + (k +_Z 4l))/4,$$

$$(x + k)/2 + (y + l)/2 = ((x + y) + (k +_Z l))/2,$$

$$x + 0 = x, \quad 0 + y = y,$$

$$0/2 = 0, \quad 2k/2 = k, \quad k + l = k +_Z l.$$

- ▶ In the proof of lemma `YSatClause` we had to solve $d' + e' + 2i = j + 4d$ for given $d', e' \in \mathbf{SD}$ and $i \in \mathbf{SD}_2$. This is a finite problem and hence can be solved by defining $J: \mathbf{SD} \rightarrow \mathbf{SD} \rightarrow \mathbf{SD}_2 \rightarrow \mathbf{SD}_2$ and $D: \mathbf{SD} \rightarrow \mathbf{SD} \rightarrow \mathbf{SD}_2 \rightarrow \mathbf{SD}$ explicitly. The validity of $d' + e' + 2i = J(d', e', i) + 4D(d', e', i)$ is proved by cases.

Extraction from lemma XSubY

The constant cXSubY of type $\mathbf{J} \rightarrow \mathbf{J} \rightarrow \mathbf{SD}_2 \times \mathbf{J} \times \mathbf{J}$ satisfies the equations

$$\text{cXSubY}(\mathbb{I}, \mathbb{I}) = \langle 0, \mathbb{I}, \mathbb{I} \rangle,$$

$$\text{cXSubY}(\mathbb{I}, C_e w) = \langle e, \mathbb{I}, w \rangle,$$

$$\text{cXSubY}(C_d v, \mathbb{I}) = \langle d, v, \mathbb{I} \rangle,$$

$$\text{cXSubY}(C_d v, C_e w) = \langle d + e, v, w \rangle.$$

For the given two streams, cXSubY computes the sum of the two head digits (regarding $\mathbb{I}$ as $C_M \mathbb{I}$), and its tails. This sum of digits of type $\mathbf{SD}_2$ is a “carry” which contains intermediate information to compute the average.

Extraction from lemma YSatClause

The constant `cYSatClause` of type

$\mathbf{SD}_2 \rightarrow \mathbf{J} \rightarrow \mathbf{J} \rightarrow \mathbf{SD}_2 \times \mathbf{SD} \times \mathbf{J} \times \mathbf{J}$ satisfies the equations

$$\text{cYSatClause}(i, \mathbb{I}, \mathbb{I}) = \langle J(0, 0, i), D(0, 0, i), \mathbb{I}, \mathbb{I} \rangle,$$

$$\text{cYSatClause}(i, \mathbb{I}, C_e w) = \langle J(0, e, i), D(0, e, i), \mathbb{I}, w \rangle,$$

$$\text{cYSatClause}(i, C_d v, \mathbb{I}) = \langle J(d, 0, i), D(d, 0, i), v, \mathbb{I} \rangle,$$

$$\text{cYSatClause}(i, C_d v, C_e w) = \langle J(d, e, i), D(d, e, i), v, w \rangle.$$

For the given carry and two signed digit streams, `cYSatClause` computes the carry for the next step, the first signed digit of the average of the streams, and the tails of the streams.

Extraction from theorem Average

The term `eterm` extracted from the proof is

```
[v0,v1]
(CoRec sdtwo@@iv@@iv=>iv)(cXSubY v0 v1)
([ivw2]
  Inr
  [let jdvw3
    (cYSatClause left ivw2
                  left right ivw2
                  right right ivw2)
    (left right jdvw3@
     (InR sdtwo@@iv@@iv iv)
     (left jdvw3@right right jdvw3))])]
```

of type $\mathbf{J} \rightarrow \mathbf{J} \rightarrow \mathbf{J}$. It calls `cXSubY` to compute the first carry and the tails of the inputs. Then `CoRec` repeatedly calls `cYSatClause`, to compute the average step by step.

Corecursion

- ▶ The conversion rules for $\mathcal{R}$ with **total ideals as recursion arguments** work from the leaves towards the root, and terminate because total ideals are well-founded.
- ▶ For cototal ideals (streams) a similar operator is available to define functions with **cototal ideals as values**: corecursion.
- ▶ ${}^{\text{co}}\mathcal{R}_{\mathbf{J}}^{\tau} : \tau \rightarrow (\tau \rightarrow \mathbf{U} + \mathbf{SD} \times (\mathbf{J} + \tau)) \rightarrow \mathbf{J}$ ($\mathbf{U}$ unit type).
- ▶ Conversion rule

$$\begin{aligned} {}^{\text{co}}\mathcal{R}_{\mathbf{J}}^{\tau} NM &\mapsto [\text{case } (MN)^{\mathbf{U} + \mathbf{SD} \times (\mathbf{J} + \tau)} \text{ of} \\ &\quad \text{inl } _ \mapsto \mathbb{I} \mid \\ &\quad \text{inr } \langle d, z \rangle \mapsto C_d[\text{case } z^{\mathbf{J} + \tau} \text{ of} \\ &\quad \quad \text{inl } _ \mapsto \mathbb{I} \mid \\ &\quad \quad \text{inr } u^{\tau} \mapsto {}^{\text{co}}\mathcal{R}_{\mathbf{J}}^{\tau} uM]]. \end{aligned}$$

An experiment

- ▶ Apply `eterm` to $1/2 + 1/8 = 5/8$ and $1/2 + 1/4 = 3/4$.

- ▶ Type the commands

```
(define test (nt (mk-term-in-app-form
  eterm (pt "C R(C M(C R II))" (pt "C R(C R II)"))))
(define neterm10 (nt (undelay-delayed-corec test 10)))
(pp neterm10)
```

- ▶ The result is

```
C R (C R (C M (C L (C M (C M (C M (C M (C M (C M
  ((CoRec sdtwo@@iv@@iv=>iv) ...))))))))))
```

- ▶ The result is correct, as

$$(5/8 + 3/4)/2 = 11/16 = 1/2 + 1/4 - 1/16.$$

References

- ▶ U. Berger, From coinductive proofs to exact real arithmetic. CSL 2009.
- ▶ U. Berger, K. Miyamoto, H.S. and M. Seisenberger, The interactive proof system Minlog. Calco-Tools 2011.
- ▶ U. Berger and M. Seisenberger, Proofs, programs, processes. CiE 2010.
- ▶ H.S., Realizability interpretation of proofs in constructive analysis. Theory of Computing Systems, 2008.
- ▶ H.S. and S.S. Wainer, Proofs and Computations. Perspectives in Logic, ASL & Cambridge UP, 2012.