Proofs, computations and analysis

Helmut Schwichtenberg

Mathematisches Institut, LMU, München

Universität Trier, 2. Juni 2012

Formalization and extraction

One can extract from a (constructive) proof of a formula with computational content a term that "realizes" (Kleene, Kreisel, Troelstra) the formula. Why should one?

- It can be important to know for sure (and to be able to machine check) that in a proof nothing has been overlooked.
- The same applies to the algorithm implicit in the proof: even if the latter is correct, errors may occur in the implementation of the algorithm.
- Even if the algorithm is correctly implemented, for sensitive applications customers may (and do) require a formal proof that the code implementing the algorithm is correct.

Consequences

- The computational content of a proof should be machine extracted from a formalization of this proof.
- The extract should be a term in the underlying language of the formal system (here: T^+, a common extension of Gödel's T and Plotkin's PCF).
- A soundness theorem should be formally proved: the extract realizes the specification ($:=$ the formula being proved).

Computable functionals

- Types: $\iota \mid \rho \to \sigma$. Ground types ι: free algebras (e.g., **N**).
- Functionals seen as limits of finite approximations: ideals (Kreisel, Scott, Ershov).
- Computable functionals are r.e. sets of finite approximations (example: fixed point functional).
- Functionals are partial. Total functionals are defined (by induction over the types).

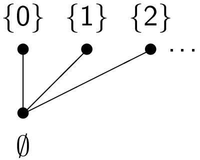
Information systems $\mathbf{C}_\rho$ for partial continuous functionals

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

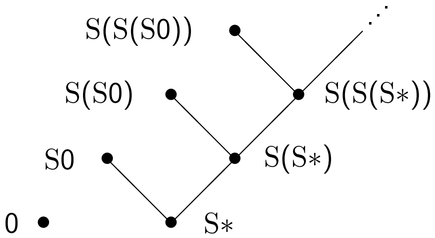
Ideals $x \in |\mathbf{C}_\rho|$ ("points", here: partial continuous functionals): consistent deductively closed sets of tokens.

Flat or non flat algebras?

- Flat:



- Non flat:



Non flat!

- Every constructor C generates an ideal in the function space: $r_{\mathrm{C}} := \{ (U, \mathrm{C}a^*) \mid U \vdash a^* \}$. Associated continuous map:

$$|r_{\mathrm{C}}|(x) = \{ \mathrm{C}a^* \mid \exists_{U \subseteq x}(U \vdash a^*) \}.$$

- Constructors are injective and have disjoint ranges:

$$|r_{\mathrm{C}}|(\vec{x}\,) \subseteq |r_{\mathrm{C}}|(\vec{y}\,) \leftrightarrow \vec{x} \subseteq \vec{y},$$
$$|r_{\mathrm{C}_1}|(\vec{x}\,) \cap |r_{\mathrm{C}_2}|(\vec{y}\,) = \emptyset.$$

- Both properties are false for flat information systems (for them, by monotonicity, constructors need to be strict).

$$|r_{\mathrm{C}}|(\emptyset, y) = \emptyset = |r_{\mathrm{C}}|(x, \emptyset),$$
$$|r_{\mathrm{C}_1}|(\emptyset) = \emptyset = |r_{\mathrm{C}_2}|(\emptyset).$$

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: $\to, \forall$ only. $=$ (Leibniz), $\exists$, $\lor$, $\land$ (Martin-Löf) inductively defined.
- $\bot := (\mathrm{False} = \mathrm{True})$. Ex-falso-quodlibet: $\bot \to A$ provable.
- Classical logic as a fragment: $\tilde{\exists}_x A$ defined by $\neg\forall_x\neg A$.

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 $\mathrm{et}(M)\ \mathbf{r}\ A$ can be proved.
- Decorations ($\to^{\mathrm{c}}, \forall^{\mathrm{c}}$ and $\to^{\mathrm{nc}}, \forall^{\mathrm{nc}}$) for removal of abstract data, and fine-tuning:

$$\begin{aligned}
t\ \mathbf{r}\ (A \to^{\mathrm{c}} B) &:= \forall_x (x\ \mathbf{r}\ A \ \to\ tx\ \mathbf{r}\ B),\\
t\ \mathbf{r}\ (A \to^{\mathrm{nc}} B) &:= \forall_x (x\ \mathbf{r}\ A \ \to\ t\ \mathbf{r}\ B),\\
t\ \mathbf{r}\ (\forall^{\mathrm{c}}_x A) &:= \forall_x (tx\ \mathbf{r}\ A),\\
t\ \mathbf{r}\ (\forall^{\mathrm{nc}}_x A) &:= \forall_x (t\ \mathbf{r}\ A).
\end{aligned}$$

Example: decorating the existential quantifier

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

$$\forall_x (A \to \exists_x A)$$

with least-fixed-point axiom

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

- Decoration leads to variants $\exists^{\mathrm{d}}, \exists^{\mathrm{l}}, \exists^{\mathrm{r}}, \exists^{\mathrm{u}}$ (d for "double", l for "left", r for "right" and u for "uniform").

$$\begin{array}{ll}
\forall^{\mathrm{c}}_x (A \to^{\mathrm{c}} \exists^{\mathrm{d}}_x A), & \exists^{\mathrm{d}}_x A \to^{\mathrm{c}} \forall^{\mathrm{c}}_x (A \to^{\mathrm{c}} P) \to^{\mathrm{c}} P, \\
\forall^{\mathrm{c}}_x (A \to^{\mathrm{nc}} \exists^{\mathrm{l}}_x A), & \exists^{\mathrm{l}}_x A \to^{\mathrm{c}} \forall^{\mathrm{c}}_x (A \to^{\mathrm{nc}} P) \to^{\mathrm{c}} P, \\
\forall^{\mathrm{nc}}_x (A \to^{\mathrm{c}} \exists^{\mathrm{r}}_x A), & \exists^{\mathrm{r}}_x A \to^{\mathrm{c}} \forall^{\mathrm{nc}}_x (A \to^{\mathrm{c}} P) \to^{\mathrm{c}} P, \\
\forall^{\mathrm{nc}}_x (A \to^{\mathrm{nc}} \exists^{\mathrm{u}}_x A), & \exists^{\mathrm{u}}_x A \to^{\mathrm{nc}} \forall^{\mathrm{nc}}_x (A \to^{\mathrm{nc}} P) \to^{\mathrm{c}} P.
\end{array}$$

Example: Supremum of an order located set of reals

- A real y is a supremum of a set S of reals if

$$\begin{aligned} &\forall_{x \in S}(x \leq y), \\ &\forall_{a<y} \exists_{x \in S}(a \leq x). \end{aligned}$$

- S is order located (above) if

$$\forall_{a,b;a<b}\big(\forall_{x \in S}(x \leq b) \vee \exists_{x \in S}(a \leq x)\big).$$

Theorem (LUB)

Assume that S is an inhabited set of reals that is bounded above. Then S has a supremum iff it is order located.

S order located $\to$ S has a supremum

- $\Pi_S(a, b)$: both $y \leq b$ for all $y \in S$ and $a < x$ for some $x \in S$.
- By assumption: $a, b \in \mathbb{Q}$ with $a < b$ such that $\Pi_S(a, b)$.
- Construct $(c_n)_n$ and $(d_n)_n$ (rationals) such that for all n

$$a = c_0 \leq c_1 \leq \cdots \leq c_n < d_n \leq \cdots \leq d_1 \leq d_0 = b, \tag{1}$$

$$\Pi_S(c_n, d_n), \tag{2}$$

$$d_n - c_n \leq (2/3)^n (b - a). \tag{3}$$

- Step: Have $c_0, \dots, c_n$ and $d_0, \dots, d_n$ such that (1)-(3).
- Let $c = c_n + \frac{1}{3}(d_n - c_n)$ and $d = c_n + \frac{2}{3}(d_n - c_n)$.
- Since S is order located, either $\forall_{y \in S}(y \leq d)$ or $\exists_{x \in S}(c < x)$.
- In the first case let $c_{n+1} := c_n$ and $d_{n+1} := d$, and in the second case let $c_{n+1} := c$ and $d_{n+1} := d_n$.
- (1)-(3) hold for $n + 1$, and the real $x = y$ given by the Cauchy sequences $(c_n)_n$ and $(d_n)_n$ is the least upper bound of S.

Nonnegative and k-positive reals

- A real number x is a pair $((a_n)_{n \in \mathbb{N}}, M)$ with $a_n \in \mathbb{Q}$ and $M \colon \mathbb{N} \to \mathbb{N}$ such that $(a_n)_n$ is a Cauchy sequence with modulus M, that is

$$|a_n - a_m| \leq 2^{-k} \quad \text{for } n, m \geq M(k).$$

- A real $x := ((a_n)_n, M)$ is *nonnegative* ($x \in \mathbb{R}^{0+}$) if

$$-2^{-k} \leq a_{M(k)} \quad \text{for all } k \in \mathbb{N}.$$

  It is *k-positive* ($x \in_k \mathbb{R}^+$) if

$$2^{-k} \leq a_{M(k+1)}.$$

- $(x \leq y) := (y - x \in \mathbb{R}^{0+})$.
- $(x < y) := \exists_k (x \in_k \mathbb{R}^+)$.

Formalization

$\forall_{y_0 \in S} \forall_{b_0} ($	S inhabited
$\forall_{x \in S}(x \leq b_0)$	b_0 upper bound of S
$\to \forall_{a,b;a<b}(\forall_{x \in S}(x \leq b) \vee \exists_{x \in S}(a \leq x))$	S order located
$\to \exists_y(\forall_{x \in S}(x \leq y) \wedge \forall_{a<y} \exists_{x \in S}(a \leq x)).$	

The type of a witness depends on the type τ of a witness for the formula defining S (example: $\mathbb{Z}$ for $S := \{ x \mid x^2 < 2 \}$):

$\mathbb{R} \to \tau \to \mathbb{Q}$	S inhabited, bound b_0 given
$\to (\mathbb{Q} \to \mathbb{Q} \to \mathbb{U} + \mathbb{R} \times \tau)$	S order located
$\to \mathbb{R} \times (\mathbb{Q} \to \mathbb{Z} \to \mathbb{R} \times \tau).$	

For a witness disregarding τ we "decorate" logical connectives:

Decoration

$$
\begin{aligned}
&\forall_{y_0}(y_0 \in S \to^{\mathrm{nc}} \forall_{b_0}(\\
&\forall_{x \in S}(x \leq b_0) \\
&\ \to \forall_{a,b;a<b}(\forall_{x \in S}(x \leq b) \vee \exists^{\mathrm{l}}_x(x \in S \wedge a \leq x)) \\
&\ \to \exists_y(\forall_{x \in S}(x \leq y) \wedge \forall_a(a < y \to \exists^{\mathrm{l}}_x(x \in S \wedge a \leq x))))).
\end{aligned}
$$

The type of a witness now is as desired

$\mathbb{R} \to \mathbb{Q}$	S inhabited, bound b_0 given
$\to (\mathbb{Q} \to \mathbb{Q} \to \mathbb{U} + \mathbb{R})$	S order located
$\to \mathbb{R} \times (\mathbb{Q} \to \mathbb{Z} \to \mathbb{R})$.	

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 extraction of its computational content.

Free algebra **J** of intervals

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

$$\begin{array}{ll} \mathbb{I} & \text{the interval } [-1,1], \\ C\colon \mathbf{SD} \to \mathbf{J} \to \mathbf{J} & \text{left, middle, right half.} \end{array}$$

- $\mathrm{C}_1\mathbb{I}$ denotes $[0, 1]$.
- $\mathrm{C}_0\mathbb{I}$ denotes $[-\frac{1}{2}, \frac{1}{2}]$.
- $\mathrm{C}_0(\mathrm{C}_{-1}\mathbb{I})$ denotes $[-\frac{1}{2}, 0]$.

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

- 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" $P(\mathrm{C}\vec{*}) \in x$.
- Total ideals: the cototal ones with $\succ_1$ well-founded.

Examples:

- Total ideals of **J**:

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

- 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

$$I0, \qquad \forall_x^{\mathrm{nc}} \forall_d \bigl(Ix \to I \frac{x+d}{2}\bigr).$$

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

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

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

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

- From a formalized proof of $\forall_{x,y}^{\mathrm{nc}}({}^{\mathrm{co}}Ix \to {}^{\mathrm{co}}Iy \to {}^{\mathrm{co}}I \frac{x+y}{2})$ extract a stream transformer, of type $\mathbf{J} \to \mathbf{J} \to \mathbf{J}$.

Arbitrary or fixed moduli

Reals:

- $((a_n)_n, M)$ Cauchy sequence plus modulus, or
- finite or infinite list of signed digits -1, 0, 1.

(Uniformly) continuous function:

- $(h_f, \alpha_f, \omega_f)$ approximating function, uniform modulus of Cauchyness plus modulus of uniform continuity, or
- possibly non well-founded labelled (with lists of signed digits -1, 0, 1) ternary tree.

Continuous functions

- Increment function $f^+ \colon \mathbf{L}(\mathbf{SD}) \to \mathbf{L}(\mathbf{SD})$.
- From f^+ we obtain $f \colon \mathbf{L}(\mathbf{SD}) \to \mathbf{L}(\mathbf{SD})$ by

$$\begin{aligned} f[] &= f^+[], \\ f(d :: a) &= f^+(d :: a) * f(a). \end{aligned}$$

- Example $\frac{x+d}{2}$:

$$\begin{aligned} f^+[] &= d, \\ f^+(d :: a) &= d. \end{aligned}$$

- Example $-x$:

$$\begin{aligned} f^+[] &= [], \\ f^+(d :: a) &= -d. \end{aligned}$$

Conclusion

- Decoration ($\to^{\mathrm{nc}}$, $\forall^{\mathrm{nc}}$, $\exists^{\mathrm{l}}$ etc.) needed to extract reasonable programs from proofs.
- Cototal ideals (type 0) to represent reals (as streams).
- Extract stream transformers from coinductive proofs.
- Work in progress (Kenji Miyamoto): continuous functions as possibly non well-founded labelled ternary trees (labels: lists of signed digits -1, 0, 1). Extract programs from coinductive proofs (e.g., composition).

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.
- 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.