Proofs and computations

Helmut Schwichtenberg

Mathematisches Institut, LMU, München

Philosophy of Science Meeting, Kyoto University, Japan, 24. May 2009

Mathematical objects

- $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ finite.
- $\mathbb{R}$ given by $\mathbb{N} \to \mathbb{Q}$ (Cauchy sequence).
- $\mathbb{R} \to \mathbb{R}$ (uniformly) continuous.

Here:

- $\mathbb{N}$,
- $\mathbb{N} \to \mathbb{N}$,
- $(\mathbb{N} \to \mathbb{N}) \to \mathbb{N} \to \mathbb{N}$,
- etc.

Gödel's "Erweiterung des finiten Standpunkts"

- Gödel (1958) proposed the unexplained notion of a total computable functional as an extension of the finitary standpoint. Example: higher type primitive recursion

$$\begin{cases} \mathcal{R}(0, r, s) = r, \\ \mathcal{R}(\mathrm{S}n, r, s) = s(n, \mathcal{R}(n, r, s)). \end{cases}$$

- This concept underlies Martin-Löf's type theory (1984).
- It is also the basis of many contemporary proof assistants (Coq, Agda, NuPrl).
- Alternative: theory of computable partial functionals, with the partial continuous functionals as their domains (Kreisel 1959, Scott 1970, Ershov 1974).

Theses

(C1) Computations are finite.

(C2) Computable functionals are recursively enumerable limits of finite partial functionals ("formal neighbourhoods").

(C3) Computable functionals are defined on "partial continuous functionals", i.e., arbitrary limits of finite partial functionals.

Total functionals can be recovered as a (dense) subset of the partial continuous functionals.

(Coherent) information systems (Scott 1982)

$\mathbf{A} = (A, \smile, \vdash)$ such that

- A is a non-empty (countable) set (the "tokens").
- $\smile$ is a reflexive and symmetric relation on A ("consistency").
- $\vdash$ is a relation between Con and A ("entailment"), where

$$\mathrm{Con} := \{\, U \mid U \subseteq^{\mathrm{fin}} A \land \forall_{a,b\in U}(a \smile b) \,\}$$

such that

$$a \in U \to U \vdash a,$$
$$\forall_{b\in V}(U \vdash b) \to V \vdash a \to U \vdash a,$$
$$a \in U \to U \vdash b \to a \smile b.$$

Examples

- Any (countable) set A can be turned into a flat information system. Tokens: all $a \in A$.

$$a \smile b :\leftrightarrow a = b \quad \text{and} \quad U \vdash a :\leftrightarrow a \in U,$$

  Then $\mathrm{Con} = \{\emptyset\} \cup \{\, \{a\} \mid a \in A \,\}$. Objects: elements of Con.

- Approximations of functions from A to B. Tokens: pairs (a, b) with $a \in A$ and $b \in B$.

$$(a_1, b_1) \smile (a_2, b_2) :\leftrightarrow (a_1 = a_2 \to b_1 = b_2),$$
$$U \vdash (a, b) :\leftrightarrow (a, b) \in U.$$

  Objects: (the graphs of) all partial functions from A to B.

Ideals, Scott topology

The ideals ("objects", "limits") of an information system $\mathbf{A} = (A, \smile, \vdash)$ are subsets x of A which satisfy

$$U \subseteq x \to U \in \mathrm{Con} \quad (x \text{ in consistent}),$$
$$x \supseteq U \vdash a \to a \in x \quad (x \text{ is deductively closed}).$$

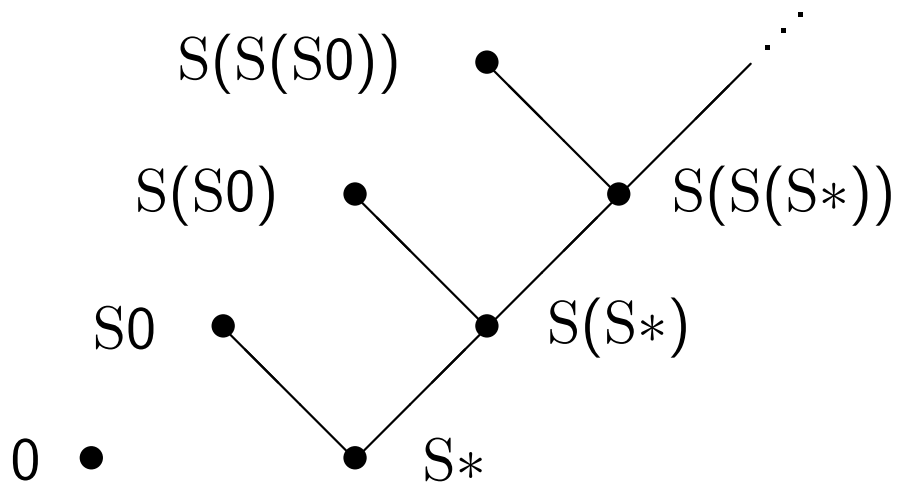
The set $|\mathbf{A}|$ of ideals for $\mathbf{A}$ carries a natural topology. Basis: cones $\tilde{U} := \{ z \mid z \supseteq U \}$ generated by the formal neighborhoods U. The continuous maps $f : |\mathbf{A}| \to |\mathbf{B}|$ and the ideals $r \in |\mathbf{A} \to \mathbf{B}|$ are in a bijective correspondence:

$$|r|(x) := \{ b \in B \mid \exists_{U \subseteq x}((U, b) \in r) \},$$
$$(U, b) \in \hat{f} :\leftrightarrow b \in f(\overline{U}) \quad \text{with } \overline{U} := \{ a \in A \mid U \vdash_A a \}.$$

These assignments are inverse to each other:

$$f = |\hat{f}| \quad \text{and} \quad r = \widehat{|r|}.$$

Tokens and entailment for **N**



Another example: Expressions **E** with constructors 0 (nullary) and C (binary). Then

$$C0* \smile C*0, \qquad \{C0*, C*0\} \vdash C00.$$

Constructors are injective and have disjoint ranges

- Let C be a constructor of ι. Then

$$|r_{\mathrm{C}}|(\vec{x}\,) = |r_{\mathrm{C}}|(\vec{y}\,) \to \vec{x} = \vec{y}.$$

- If $\mathrm{C}_1, \mathrm{C}_2$ are distinct constructors of ι then

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

These properties are mandatory for equational reasoning. They do not hold for flat $\mathbf{A}_\iota$'s:

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

(with $\bot := \emptyset$), since for flat $\mathbf{A}_\iota$'s constructors need to be strict.

The typed λ-calculus T^+

Common extension of Gödel's T and Plotkin's PCF.

$$M, N ::= x^\rho \mid \mathrm{C}^\rho \mid D^\rho \mid (\lambda_{x^\rho} M^\sigma)^{\rho\to\sigma} \mid (M^{\rho\to\sigma} N^\rho)^\sigma.$$

Every defined constant D^ρ comes with a system of computation rules, consisting of equations

$$D\vec{P}_i(\vec{y}_i) = M_i(\vec{y}_i)$$

with constructor patterns $\vec{P}_i$.

Examples of constants D with their computation rules

- $\begin{cases} D0 := 0, \\ D(\mathrm{S}n) := \mathrm{S}(\mathrm{S}(Dn)). \end{cases}$
- Addition, multiplication.
- Gödel's higher type primitive recursion operators $\mathcal{R}$:

$$\begin{cases} \mathcal{R}(0, r, s) = r, \\ \mathcal{R}(\mathrm{S}n, r, s) = s(n, \mathcal{R}(n, r, s)). \end{cases}$$

- The fixed point operators Y:

$$Yw = w(Yw).$$

Inductive definition of $(\vec{U}, a) \in [\![\lambda_{\vec{x}} M]\!]$

$$\frac{U_i \vdash a}{(\vec{U}, a) \in [\![\lambda_{\vec{x}} x_i]\!]}(V), \qquad \frac{(\vec{U}, V) \subseteq [\![\lambda_{\vec{x}} N]\!] \qquad (\vec{U}, V, a) \in [\![\lambda_{\vec{x}} M]\!]}{(\vec{U}, a) \in [\![\lambda_{\vec{x}}(MN)]\!]}(A).$$

For every constructor C and defined constant D

$$\frac{\vec{V} \vdash \vec{a}}{(\vec{U}, \vec{V}, \mathrm{C}\vec{a}) \in [\![\lambda_{\vec{x}} \mathrm{C}]\!]}(C), \qquad \frac{(\vec{U}, \vec{V}, a) \in [\![\lambda_{\vec{x},\vec{y}} M]\!] \qquad \vec{W} \vdash \vec{P}(\vec{V})}{(\vec{U}, \vec{W}, a) \in [\![\lambda_{\vec{x}} D]\!]}(D)$$

with one such rule (D) for every computation rule $D\vec{P}(\vec{y}\,) = M$.

Properties of $[\![M]\!]$

- $[\![M]\!]$ is an ideal, i.e., consistent and deductively closed.
- $[\![MN]\!] = [\![M]\!][\![N]\!]$.
- $[\![M]\!]$ is preserved under reduction, generated from

$(\lambda_x M(x))N \mapsto M(N)$ β-conversion,

$\lambda_x(Mx) \mapsto M \quad (x \notin \mathrm{FV}(M))$ η-conversion,

$D\vec{P}(\vec{N}) \mapsto M(\vec{N})$ for $D\vec{P}(\vec{y}\,) = M(\vec{y}\,)$ a computation rule.

Extension of known results to the non-flat setting

- (Definability: Plotkin 1977). F is computable iff it can be defined by a T^+-term involving the parallel conditional pcond, a continuous approximation $\exists$ to the existential quantifier, and valmax:

$$\frac{U \vdash a_n \qquad U \vdash a}{(U, \{\mathrm{S}^n 0\}, a) \in [\![\mathrm{valmax}]\!]}(M_1) \qquad \frac{\{a_n\} \vdash a}{(U, \{\mathrm{S}^n 0\}, a) \in [\![\mathrm{valmax}]\!]}(M_2).$$

  (j.w.w. Basil Karadais).
- (Adequacy: Plotkin 1977). Suppose $[\![M]\!]$ is a "numeral" (i.e., a closed constructor expression). Then M reduces to it.
- (Density: Kreisel 1959, Ershov 1974). The total functionals are dense (w.r.t. the Scott topology).

Theses

(P1) The only (basic) logical connectives are $\to$, $\forall$.

(P2) Proofs have two aspects: (a) They guarantee correctness. (b) They may have computational content.

(P3) Computational content only enters a proof via inductively (or coinductively) defined predicates.

(P4) To fine tune the computational content of a proof, distinguish $\to^{\mathrm{c}}$, $\forall^{\mathrm{c}}$ (computational) and $\to$, $\forall$ (non-computational).

Formulas as computational problems

- Kolmogorov (1925) proposed to view a formula A as a computational problem, of type $\tau(A)$, the type of a potential solution or "realizer" of A.
- $\tau(A)$ should be the type of the term (or "program") to be extracted from a proof of A.
- Formally: $A \mapsto \tau(A)$ (a type or the nulltype symbol ε).
- In case $\tau(A) = \varepsilon$ proofs of A have no computational content; such formulas A are called computationally irrelevant (c.i.); the other ones computationally relevant (c.r.).

Restrictions to $\to^+$ and $\forall^+$

$\mathrm{CV}(M) :=$ the set of "computational variables" of a derivation M.
Consider

$$\dfrac{\begin{array}{c}[u\colon A]\\ \mid M\\ B\end{array}}{A \to B}\to^+ u \qquad \text{or as term} \quad (\lambda_{u^A} M)^{A\to B}.$$

$(\lambda_{u^A} M)^{A\to B}$ is correct if M is and $x_u \notin \mathrm{CV}(M)$. Consider

$$\dfrac{\begin{array}{c}\mid M\\ A\end{array}}{\forall_x A}\forall^+ x \qquad \text{or as term} \quad (\lambda_x M)^{\forall_x A} \qquad \text{(with usual var. condition).}$$

$(\lambda_x M)^{\forall_x A}$ is correct if M is and $x \notin \mathrm{CV}(M)$.

Example: totality

Totality predicates T_ρ are defined by induction on ρ. For base types, e.g. for $\mathbf{N}$: inductive definition, by the clauses

$$T0, \qquad \forall_n(Tn \to^c T(\mathrm{S}n)).$$

Elimination (or least fixed point) axiom (writing $\forall^c_{n\in T}A$ for $\forall_n(Tn \to^c A)$):

$$\forall^c_{n\in T}(A(0) \to^c \forall^c_{n\in T}(A(n) \to^c A(\mathrm{S}n)) \to^c A(n)).$$

This is the induction scheme.

Example: Leibniz equality Eq

- Inductively defined by the introduction axiom

$$\forall_x \mathrm{Eq}_\rho(x^\rho, x^\rho).$$

- Elimination axiom:

$$\forall^{\mathrm{c}}_{x,y}(\mathrm{Eq}(x,y) \to \forall^{\mathrm{c}}_x C(x,x) \to^{\mathrm{c}} C(x,y)).$$

- With $C(x,y) := A(x) \to^{\mathrm{c}} A(y)$ this implies

$$\forall^{\mathrm{c}}_{x,y}(\mathrm{Eq}(x,y) \to A(x) \to^{\mathrm{c}} A(y)) \quad \text{(compatibility of } \mathrm{Eq}).$$

- Compatibility gives symmetry and transitivity of Eq.

Example: $\exists$

- For $\exists_x A$ one may decorate in its single clause $\forall_x(A \to \exists_x A)$ independently both, $\forall$ and $\to$.
- This gives four (only) computationally different variants $\exists^{\mathrm{d}}, \exists^{\mathrm{l}}, \exists^{\mathrm{r}}, \exists$ of the existential quantifier, with axioms

$$\begin{array}{ll}
\forall_x^{\mathrm{c}}(A \to^{\mathrm{c}} \exists_x^{\mathrm{d}} A), & \exists_x^{\mathrm{d}} A \to^{\mathrm{c}} \forall_x^{\mathrm{c}}(A \to^{\mathrm{c}} B) \to^{\mathrm{c}} B, \\
\forall_x^{\mathrm{c}}(A \to \exists_x^{\mathrm{l}} A), & \exists_x^{\mathrm{l}} A \to^{\mathrm{c}} \forall_x^{\mathrm{c}}(A \to B) \to^{\mathrm{c}} B, \\
\forall_x(A \to^{\mathrm{c}} \exists_x^{\mathrm{r}} A), & \exists_x^{\mathrm{r}} A \to^{\mathrm{c}} \forall_x(A \to^{\mathrm{c}} B) \to^{\mathrm{c}} B, \\
\forall_x(A \to \exists_x A), & \exists_x A \to \forall_x(A \to B) \to^{\mathrm{c}} B.
\end{array}$$

Similarly for $\wedge$, $\vee$.

Ex-Falso-Quodlibet

need not be assumed, but can be proved.

$$\mathbf{F} \to A, \text{ with } \mathbf{F} := \mathrm{Eq}(\mathrm{ff}, \mathrm{tt}) \text{ ("falsity").}$$

The proof is in 2 steps. (1) $\mathbf{F} \to \mathrm{Eq}(x^\rho, y^\rho)$, since from $\mathrm{Eq}(\mathrm{ff}, \mathrm{tt})$ by compatibility

$$\mathrm{Eq}\,\underbrace{[\textbf{if } \mathrm{tt} \textbf{ then } x \textbf{ else } y]}_{x}\,\underbrace{[\textbf{if } \mathrm{ff} \textbf{ then } x \textbf{ else } y]}_{y}.$$

(2) Induction on (the sim. definition of) predicates and formulas.

- Case $I\vec{s}$. Let K_0 be the nullary clause $A_1 \to \cdots \to A_n \to I\vec{t}$. By IH: $\mathbf{F} \to A_i$. Hence $I\vec{t}$. From $\mathbf{F}$ we also obtain $\mathrm{Eq}(s_i, t_i)$, by (1). Hence $I\vec{s}$ by compatibility.
- The cases $A \to B$, $A \to^{\mathrm{c}} B$, $\forall_x A$ and $\forall^{\mathrm{c}}_x A$ are easy.

Embedding classical arithmetic

- Let $\neg A := (A \to \mathbf{F})$, and

$$\tilde{\exists}_x A := \neg\forall_x \neg A, \qquad A \,\tilde{\vee}\, B := (\neg A \to \neg B \to \mathbf{F}).$$

- Consider a total boolean term $r^{\mathbf{B}}$ as representing a decidable predicate: $\mathrm{Eq}(r, \mathrm{tt})$.
- Prove $\forall_{p\in T}(\neg\neg \mathrm{Eq}(p, \mathrm{tt}) \to \mathrm{Eq}(p, \mathrm{tt}))$ by boolean induction.
- Lift this via $\to, \forall$ using

$$\vdash (\neg\neg B \to B) \to \neg\neg(A \to B) \to A \to B,$$
$$\vdash (\neg\neg A \to A) \to \neg\neg\forall_x A \to \forall_x A.$$

- For formulas A built from $\mathrm{Eq}(\cdot, \mathrm{tt})$ by $\to, \forall_{x\in T}$ prove stability

$$\forall_{\vec{x}\in T}(\neg\neg A \to A) \qquad (\mathrm{FV}(A) \text{ among } \vec{x}).$$

Soundness

Let M be a derivation of A from assumptions $u_i : C_i$ $(i < n)$. Then we can find a derivation of $[\![M]\!]\ \mathbf{r}\ A$ from assumptions

$$\begin{cases} x_{u_i}\ \mathbf{r}\ C_i & \text{for } \tau(C_i) \neq \varepsilon \text{ and } x_{u_i} \in \mathrm{CV}(M) \\ \exists_x(x\ \mathbf{r}\ C_i) & \text{for } \tau(C_i) \neq \varepsilon \text{ and } x_{u_i} \notin \mathrm{CV}(M) \\ \varepsilon\ \mathbf{r}\ C_i & \text{for } \tau(C_i) = \varepsilon. \end{cases}$$

Decoration can simplify extracts

- Suppose that a proof M uses a lemma $L^{\mathrm{d}}\colon A \vee^{\mathrm{d}} B$.
- Then the extract $[\![M]\!]$ will contain the extract $[\![L^{\mathrm{d}}]\!]$.
- Suppose that the only computationally relevant use of L^{d} in M was which one of the two alternatives holds true, A or B.
- Express this by using a weakened $L\colon A \vee B$.
- Since $[\![L]\!]$ is a boolean, the extract of the modified proof is "purified": the (possibly large) extract $[\![L^{\mathrm{d}}]\!]$ has disappeared.

Decorating proofs

Goal: Insertion of as few as possible decorations into a proof.
Write $\forall^{\mathrm{c}}_{n\in T} A$ for $\forall_n(T_{\mathbf{N}} n \to^{\mathrm{c}} A)$.

- $\mathrm{Seq}(M)$ of a proof M consists of its context and end formula.
- The uniform proof pattern $U(M)$ of a proof M is the result of changing in c.r. formulas of M (i.e., not above a c.i. formula) all $\to^{\mathrm{c}}$, $\forall^{\mathrm{c}}$ into $\to$, $\forall$, except "uninstantiated" formulas of axioms, e.g., $\forall^{\mathrm{c}}_{n\in T}(P0 \to^{\mathrm{c}} \forall^{\mathrm{c}}_{n\in T}(Pn \to^{\mathrm{c}} P(\mathrm{S}n)) \to^{\mathrm{c}} Pn)$.
- A formula D extends C if D is obtained from C by changing some $\to$, $\forall$ into $\to^{\mathrm{c}}$, $\forall^{\mathrm{c}}$.
- A proof N extends M if (1) N and M are the same up to variants of $\to$, $\forall$ in their formulas, and (2) every c.r. formula of M is extended by the corresponding one in N.

Decoration algorithm

Assumption: We have an algorithm assigning to every axiom A and every decoration variant C of A another axiom whose formula D extends C, and D is the least among those extensions.

Theorem (Ratiu, H.S.)

Under the assumption above, for every uniform proof pattern U and every extension of its sequent $\mathrm{Seq}(U)$ we can find a decoration M_∞ of U such that

(a) $\mathrm{Seq}(M_\infty)$ *extends the given extension of* $\mathrm{Seq}(U)$, *and*

(b) M_∞ *is optimal in the sense that any other decoration M of U whose sequent $\mathrm{Seq}(M)$ extends the given extension of $\mathrm{Seq}(U)$ has the property that M also extends M_∞.*

Example: list reversal (U. Berger 2005)

- Give "weak" existence proof ($\tilde{\exists}$, but with a predicate variable $\bot$ instead of **F**).
- A-translate into an existence proof. Extracted algorithm: $f(v_1) := h(v_1, \mathrm{nil}, \mathrm{nil})$ with

$$h(\mathrm{nil}, v_2, v_3) := v_3, \quad h(xv_1, v_2, v_3) := h(v_1, v_2x, xv_3).$$

  The second argument of h is not needed, but makes the algorithm quadratic. (In each recursion step v_2x is computed, and list append is defined by recursion over its first argument.)
- Compute optimal decoration of existence proof. Extracted algorithm: $f(v_1) := g(v_1, \mathrm{nil})$ with

$$g(\mathrm{nil}, v_2) := v_2, \quad g(xv_1, v_2) := g(v_1, xv_2).$$

  This is the well-known linear algorithm, with an accumulator.

Example: maximal segments (Bates & Constable 1985)

- Let X be linearly ordered by $\leq$. Given

$$\mathrm{seg}\colon \mathbf{N} \to \mathbf{N} \to X.$$

(Example: $X = \mathbb{Z}$ and $\mathrm{seg}(i,k) = f(i) + \cdots + f(k)$ for some $f\colon \mathbf{N} \to \mathbb{Z}$.)

- Want: maximal segment

$$\forall_n^{\mathrm{c}} \exists_{i \leq k \leq n}^{\mathrm{l}} \forall_{i' \leq k' \leq n} (\mathrm{seg}(i',k') \leq \mathrm{seg}(i,k)).$$

$(n, i, j, k \in T_{\mathbf{N}})$.

- Special case: maximal end segment

$$\forall_n^{\mathrm{c}} \exists_{j \leq n}^{\mathrm{l}} \forall_{j' \leq n} (\mathrm{seg}(j',n) \leq \mathrm{seg}(j,n)).$$

Example: maximal segment problem (ctd.)

2 proofs of the existence of a maximal end segment for $n+1$

$$\forall_n^{\mathrm{c}} \exists_{j \le n+1}^{\mathrm{l}} \forall_{j' \le n+1} (\mathrm{seg}(j', n+1) \le \mathrm{seg}(j, n+1)).$$

- Introduce an auxiliary parameter m; prove by induction on m

$$\forall_n \forall_{m \le n+1}^{\mathrm{c}} \exists_{j \le n+1}^{\mathrm{l}} \forall_{j' \le m} (\mathrm{seg}(j', n+1) \le \mathrm{seg}(j, n+1)).$$

- Use $\mathtt{ES}_n$: $\exists_{j \le n}^{\mathrm{l}} \forall_{j' \le n} (\mathrm{seg}(j', n) \le \mathrm{seg}(j, n))$ and the additional assumption of monotonicity

$$\forall_{i,j,n} (\mathrm{seg}(i, n) \le \mathrm{seg}(j, n) \to \mathrm{seg}(i, n+1) \le \mathrm{seg}(j, n+1)).$$

Proceed by cases on $\mathrm{seg}(j, n+1) \le \mathrm{seg}(n+1, n+1)$. If $\le$, take $n+1$, else the previous j.

Example: maximal segment problem (ctd.)

Prove the existence of a maximal segment by induction on n, simultaneously with the existence of a maximal end segment.

$$\forall_n^{\mathrm{c}}(\exists^{\mathrm{l}}_{i\le k\le n}\forall_{i'\le k'\le n}(\mathrm{seg}(i',k')\le \mathrm{seg}(i,k))\wedge^{\mathrm{d}}$$
$$\exists^{\mathrm{l}}_{j\le n}\forall_{j'\le n}(\mathrm{seg}(j',n)\le \mathrm{seg}(j,n)))$$

In the step:

- Compare the maximal segment i,k for n with the maximal end segment $j,n+1$ proved separately.
- If $\le$, take the new i,k to be $j,n+1$. Else take the old i,k.

Depending on how the existence of a maximal end segment was proved, we obtain a quadratic or a linear algorithm.

Example: maximal segment problem (ctd.)

How could the better proof be found? We have

$$\mathtt{L1}: \forall_n^{\mathrm{c}} \exists_{j \le n+1}^{\mathrm{l}} \forall_{j' \le n+1} (\mathrm{seg}(j', n+1) \le \mathrm{seg}(j, n+1)),$$
$$\mathtt{L2}: \forall_n (\mathtt{ES}_n \to^{\mathrm{c}} \mathtt{Mon} \to \exists_{j \le n+1}^{\mathrm{l}} \forall_{j' \le n+1} (\mathrm{seg}(j', n+1) \le \mathrm{seg}(j, n+1))).$$

- The decoration algorithm arrives at L1 with

$$\exists_{j \le n+1}^{\mathrm{l}} \forall_{j' \le n+1} (\mathrm{seg}(j', n+1) \le \mathrm{seg}(j, n+1)).$$

- L2 fits as well, its assumptions $\mathtt{ES}_n$ and Mon are in the context, and it has $\forall_n$ rather than $\forall_n^{\mathrm{c}}$, hence is preferred.

References

- H.S., Recursion on the partial continuous functionals. In: Logic Colloquium 2005 (ASL Lecture Notes in Logic 28).
- U. Berger, Program extraction from normalization proofs. In: Proc. TLCA 1993 (Springer LNCS 664).
- U. Berger, Uniform Heyting arithmetic. APAL 133 (2005).
- D. Ratiu and H.S., Decorating proofs. To appear, Mints volume (ed. S. Feferman and W. Sieg), 2009.