Advertisement

- 2 postdoc grants in Munich, from April & Sep. 2003

$\leq$ 2 years, within Graduiertenkolleg "Logic in Computer Science", `http://www.mathematik.uni-muenchen.de/~gkli/`

Apply before March 1, 2003

- Marktoberdorf Summer School, 29. July - 9. Aug. 2003

(Constable, Dowek, Grumberg, Harrison, Jones, Nipkow, Rathjen, Schwichtenberg, Urzykzyn, Vardi, Wainer)

"Proof Technology". Apply before March 8, 2003

An arithmetic for polynomial-time computation

Helmut Schwichtenberg

Mathematisches Institut, Universität München

Part 1: Feasible computation with higher types

Goal: Restriction $\mathbf{LT}$ of the system $\mathbf{T}$ of Gödel '58

Part 2: Transfer to proofs via Curry-Howard corresp.

- Two sorts of variables: complete $\overline{x}^{\rho}$ and incomplete x^{ρ}.
- (affine) linearity: higher type incomplete variables occur at most once.
- ramification:

$$\begin{cases} \rho \to \sigma \\ \lambda \overline{x}^{\rho} r \end{cases} \quad \text{as well as} \quad \begin{cases} \rho \multimap \sigma \\ \lambda x^{\rho} r \end{cases}$$

Two recursions

$$d(1) := \mathsf{S}_0(1) \qquad e(1) := 1$$
$$d(\mathsf{S}_i(x)) := \mathsf{S}_0(\mathsf{S}_0(d(x))) \qquad e(\mathsf{S}_i(x)) := d(e(x))$$

Then $|d(x)| = 2|x|$, $e(x) = d^{(|x|-1)}(1)$.

Problem: Previous value of 2nd recursion is rec. argument

Cure: recursion arguments "complete", i.e., d: $\mathbf{W} \to \mathbf{W}$, not $\mathbf{W} \multimap \mathbf{W}$.

Similar:

"safe" vs. "normal" arg. (Simmons, Bellantoni/Cook),

"tiering" (Leivant).

Recursion with parameter substitution (1)

$$e(1,y) := \mathsf{S}_0(y) \qquad e(\mathsf{S}_i(x),y) := e(x,e(x,y))$$

or

$$e(1) := \mathsf{S}_0 \qquad e(\mathsf{S}_i(x)) := e(x) \circ e(x)$$

Then $e(x) = \mathsf{S}_0^{(2^{|x|-1})}$.

Problem: Previous higher type value of rec. used twice.

Cure: (affine) linearity, i.e., higher type incomplete variables occur at most once.

Recursion with parameter substitution (2)

$$e(1, y) := y \qquad e(\mathsf{S}_i(x), y) := e(x, d(y))$$

or

$$e(1) := \mathsf{id} \qquad e(\mathsf{S}_i(x)) := e(x) \circ d$$

Then $e(x) = d^{(|x|-1)}$.

Problem: Higher result type with marked argument types (marked as recursion arguments).

Cure: Result types linear, i.e., without $\rightarrow$.

Higher argument type: iteration (1)

$$I(1, f) := \mathsf{id}$$

$$I(\mathsf{S}_i(x), f) := f \circ I(x, f)$$

Then $I(x, d) = d^{(|x|-1)}$.

Problem: Recursively defined d in function parameter.

Cure: Result types linear, i.e., without $\rightarrow$.

Higher argument type: iteration (2)

$$e(1) := \mathsf{S}_0$$

$$e(\mathsf{S}_i(x)) := I(\mathsf{S}_0(\mathsf{S}_0(1)), e(x))$$

Then: $e(x) = \mathsf{S}_0^{(2^{|x|-1})}$.

Problem: Previous higher type value $e(x)$ in step of I.

Cure: no higher type incomplete parameters in step.

Related work

Bounded recursion (Cobham, Cook/Urquhart)

Safe & ramified rec. (Simmons, Bellantoni/Cook, Leivant)

Bounded linear logic (Girard/Scedrov/Scott)

Light linear logic (Girard, Asperti, Roversi)

Soft linear logic (Lafont)

Typed λ-calculi for poly-time computation (Hofmann)

Modal type theory (Pfenning)

Bellantoni/Niggl/S

Types

$$\rho, \sigma ::= \mathbf{U} \mid \mathbf{B} \mid \mathbf{L}(\rho) \mid \rho \multimap \sigma \mid \rho \to \sigma \mid \rho \otimes \sigma \mid \rho \times \sigma$$

$\to$-free types are called linear. The level of a type:

$$\begin{aligned} l(\mathbf{U}) := l(\mathbf{B}) &:= 0 \\ l(\mathbf{L}(\rho)) &:= l(\rho) \\ l(\rho \multimap \sigma) := l(\rho \to \sigma) &:= \max\{l(\sigma), 1 + l(\rho)\} \\ l(\rho \otimes \sigma) &:= \max\{l(\rho), l(\sigma)\} \\ l(\rho \times \sigma) &:= \max\{l(\rho), l(\sigma), 1\} \end{aligned}$$

A higher type is any type of level at least 1.

Constants

ε : $\mathbf{U}$

tt, ff : $\mathbf{B}$

nil_ρ : $\mathbf{L}(\rho)$

cons_ρ : $\rho \multimap \mathbf{L}(\rho) \multimap \mathbf{L}(\rho)$

if_τ : $\mathbf{B} \multimap \tau \times \tau \multimap \tau$ (τ linear)

$c_\tau^{\mathbf{L}(\rho)}$: $\mathbf{L}(\rho) \multimap \tau \times (\rho \multimap \mathbf{L}(\rho) \multimap \tau) \multimap \tau$ (τ linear)

$\mathcal{R}_\tau^{\mathbf{L}(\rho)}$: $\mathbf{L}(\rho) \to (\rho \to \mathbf{L}(\rho) \to \tau \multimap \tau) \to \tau \multimap \tau$

(ρ ground, τ linear)

Constants (ctd.)

$$\otimes^{+}_{\rho\sigma} \;:\; \rho \multimap \sigma \multimap \rho \otimes \sigma$$

$$\otimes^{-}_{\rho\sigma\tau} \;:\; \rho \otimes \sigma \multimap (\rho \multimap \sigma \multimap \tau) \multimap \tau$$

$$\times^{+}_{\rho\sigma\vec{\tau}} \;:\; (\vec{\tau} \multimap \rho) \multimap (\vec{\tau} \multimap \sigma) \multimap \vec{\tau} \multimap \rho \times \sigma$$

$$\mathsf{fst}_{\rho\sigma} \;:\; \rho \times \sigma \multimap \rho$$

$$\mathsf{snd}_{\rho\sigma} \;:\; \rho \times \sigma \multimap \sigma$$

Abbreviations for $\mathbf{N} := \mathbf{L(U)}$ and $\mathbf{W} := \mathbf{L(B)}$

$$0 := \mathsf{nil}_{\mathbf{U}}$$

$$\mathsf{S} := \lambda l^{\mathbf{L(U)}}.\mathsf{cons}_{\mathbf{U}}\varepsilon l$$

$$1 := \mathsf{nil}_{\mathbf{B}}$$

$$\mathsf{S}_0 := \lambda l^{\mathbf{L(B)}}.\mathsf{cons}_{\mathbf{B}}\mathsf{ff}\, l$$

$$\mathsf{S}_1 := \lambda l^{\mathbf{L(B)}}.\mathsf{cons}_{\mathbf{B}}\mathsf{tt}\, l$$

Terms

c^ρ (constant) |

x^ρ (incomplete variable) |

$\overline{x}^\rho$ (complete variable) |

$(\lambda x^\rho r^\sigma)^{\rho \multimap \sigma}$ |

$(r^{\rho \multimap \sigma} s^\rho)^\sigma$ with h.t. incomplete vars in r, s distinct |

$(\lambda \overline{x}^\rho r^\sigma)^{\rho \to \sigma}$ |

$(r^{\rho \to \sigma} s^\rho)^\sigma$ with s complete

A term s is complete if all its free variables are complete.

Conversions

$$(\lambda x r)s \mapsto r[x := s]$$

$$\mathsf{if}_\tau \mathsf{tt} s \mapsto \mathsf{fst}_{\tau\tau} s, \quad \mathsf{if}_\tau \mathsf{ff}\, s \mapsto \mathsf{snd}_{\tau\tau} s$$

$$\boldsymbol{c}_\tau \mathsf{nil}_\rho s \mapsto \mathsf{fst}\, s, \quad \boldsymbol{c}_\tau (\mathsf{cons}_\rho r l) s \mapsto \mathsf{snd}\, s r l$$

$$\mathcal{R}_\tau \mathsf{nil}_\rho s t \mapsto t, \quad \mathcal{R}_\tau (\mathsf{cons}_\rho r l) s t \mapsto s r l (\mathcal{R}_\tau l s t)$$

$$\otimes^-_{\rho\sigma\tau} (\otimes^+_{\rho\sigma} r s) t \mapsto t r s$$

$$\mathsf{fst}_{\rho\sigma} (\times^+_{\rho\sigma\tau} r s t) \mapsto r t, \quad \mathsf{snd}_{\rho\sigma} (\times^+_{\rho\sigma\tau} r s t) \mapsto s t$$

Two recursions

$$d(1) := \mathsf{S}_0(1) \qquad e(1) := 1$$
$$d(\mathsf{S}_i(x)) := \mathsf{S}_0(\mathsf{S}_0(d(x))) \qquad e(\mathsf{S}_i(x)) := d(e(x))$$

Then $e(x) = d^{(|x|-1)}(1)$.

Problem: Previous value of 2nd recursion is rec. argument

Cure: recursion arguments "complete", i.e., $d\colon \mathbf{W} \to \mathbf{W}$, not $\mathbf{W} \multimap \mathbf{W}$.

Terms:

$$d := \lambda\bar{x}.\mathcal{R}_{\mathbf{W}}\bar{x}(\lambda\bar{z}\lambda\bar{l}\lambda p^{\mathbf{W}}.\mathsf{S}_0(\mathsf{S}_0 p))(\mathsf{S}_0 1)\colon \mathbf{W} \to \mathbf{W},$$
$$e := \lambda\bar{x}.\mathcal{R}_{\mathbf{W}}\bar{x}(\lambda\bar{z}\lambda\bar{l}\lambda p^{\mathbf{W}}.dp)1.$$

Recursion with parameter substitution (1)

$$e(1,y) := \mathsf{S}_0(y) \qquad e(\mathsf{S}_i(x),y) := e(x,e(x,y))$$

or

$$e(1) := \mathsf{S}_0 \qquad e(\mathsf{S}_i(x)) := e(x) \circ e(x)$$

Then $e(x) = \mathsf{S}_0^{(2^{|x|-1})}$.

Problem: Previous higher type value of rec. used twice.

Cure: (affine) linearity. Term:

$$\lambda \bar{x}.\mathcal{R}_{\mathbf{W} \multimap \mathbf{W}} \bar{x} (\lambda \bar{z} \lambda \bar{l} \lambda p^{\mathbf{W} \multimap \mathbf{W}} \lambda y.p(py)) \mathsf{S}_0$$

Recursion with parameter substitution (2)

$$e(1, y) := y \qquad e(1) := \mathsf{id}$$
$$e(\mathsf{S}_i(x), y) := e(x, d(y)) \quad \text{or} \quad e(\mathsf{S}_i(x)) := e(x) \circ d$$

Then $e(x) = d^{(|x|-1)}$.

Problem: Higher result type with marked argument types (marked as recursion arguments).

Cure: Result types linear, i.e., without $\to$. Term:

$$\lambda\bar{x}.\mathcal{R}_{\mathbf{W}\to\mathbf{W}}\bar{x}(\lambda\bar{z}\lambda\bar{l}\lambda p^{\mathbf{W}\to\mathbf{W}}\lambda\bar{x}.p(d\bar{x}))(\lambda\bar{y}\bar{y})$$

Higher argument type: iteration (1)

$$I(1, f) := \mathsf{id}$$

$$I(\mathsf{S}_i(x), f) := f \circ I(x, f)$$

Then $I(x, d) = d^{(|x|-1)}$.

Problem: Recursively defined d in function parameter.

Cure: Result types linear, i.e., without $\to$. Terms:

$$I_f := \lambda\overline{x}.\mathcal{R}_{\mathbf{W}\multimap\mathbf{W}}\overline{x}(\lambda\overline{z}\lambda\overline{l}\lambda p^{\mathbf{W}\multimap\mathbf{W}}\lambda y.f(py))(\lambda yy)$$

$$e := \lambda x.I_d x1 \quad \text{with } d\colon \mathbf{W}\to\mathbf{W}$$

Higher argument type: iteration (2)

$$e(1) := \mathsf{S}_0$$
$$e(\mathsf{S}_i(x)) := I(\mathsf{S}_0(\mathsf{S}_0(1)), e(x))$$

Then: $e(x) = \mathsf{S}_0^{(2^{|x|-1})}$.

Problem: Previous higher type value $e(x)$ in step of I.

Cure: no higher type incomplete parameters in step.

Terms:

$$I_f := \lambda x.\mathcal{R}_{\mathbf{W}\multimap\mathbf{W}} x(\lambda x \lambda p^{\mathbf{W}\multimap\mathbf{W}} \lambda y.f(py))(\lambda yy)$$
$$e := \lambda x.\mathcal{R}_{\mathbf{W}\multimap\mathbf{W}} x(\lambda x \lambda q^{\mathbf{W}\multimap\mathbf{W}}.I_q(\mathsf{S}_0(\mathsf{S}_0 1)))\mathsf{S}_0$$

Polynomials

$\oplus\colon \mathbf{W} \to \mathbf{W} \multimap \mathbf{W}$. $x \oplus y$ concatenates $|x|$ bits onto y.

$$1 \oplus y = \mathsf{S}_0 y$$

$$(\mathsf{S}_i x) \oplus y = \mathsf{S}_0(x \oplus y)$$

$\overline{x} \oplus y := \mathcal{R}_{\mathbf{W} \multimap \mathbf{W}} \overline{x} (\lambda \overline{z} \lambda \overline{l} \lambda p^{\mathbf{W} \multimap \mathbf{W}} \lambda y.\mathsf{S}_0(py))\mathsf{S}_0$.

$\odot\colon \mathbf{W} \to \mathbf{W} \to \mathbf{W}$. $x \odot y$ has output length $|x| \cdot |y|$.

$$x \odot 1 = x$$

$$x \odot (\mathsf{S}_i y) = x \oplus (x \odot y)$$

$\overline{x} \odot \overline{y} := \mathcal{R}_{\mathbf{W}} \overline{y} (\lambda \overline{z} \lambda \overline{l} \lambda p^{\mathbf{W}} \overline{x} \oplus p) \overline{x}$.

Feasible computation with higher types

Theorem. The closed $\mathbf{LT}$ terms of type $\vec{\mathbf{W}} \multimap \mathbf{W}$ denote exactly the functions computable in polynomial time.

(Computation model: parse dags, where every node with in-degree > 1 is of ground type; S. & Bellantoni 2002)

But: already $\mathcal{R}$-free normalization is superexponential.

Solution: this is only a constant w.r.t. to the length of integers $\vec{n}$ at which we evaluate the denoted function.

Part 2: Restriction LHA of HA

Goal: all extracted programs feasible. How?

- Three sorts of (assumption and object) variables: passive, complete and incomplete ones.
- (affine) linearity: higher type incomplete variables occur at most once.
- ramification:

$\tilde{\forall} x^\rho A$ as well as $\begin{cases} A \to B \\ \bar{\forall} x^\rho A \end{cases}$ as well as $\begin{cases} A \multimap B \\ \forall x^\rho A \end{cases}$

Goal (ctd.)

LHA solves

$$\frac{\text{Gödel's } \mathbf{T}}{\mathbf{HA}} = \frac{\text{Restriction } \mathbf{LT} \text{ of } \mathbf{T}}{?}$$

Motivation: Examples of arithmetical existence proofs exhibiting exponential growth.

Double use of assumptions

$$e(1,y) := \mathsf{S}_0(y) \qquad e(1) := \mathsf{S}_0$$
$$e(\mathsf{S}_i(x),y) := e(x,e(x,y)) \quad \text{or} \quad e(\mathsf{S}_i(x)) := e(x) \circ e(x)$$

Then $e(x) = \mathsf{S}_0^{(2^{|x|-1})}$, i.e., e grows exponentially.

Corresponding existence proof:

$$\forall x, y \exists z \, |z| = 2^{|x|-1} + |y|$$

Ind(x): ("functional") IH used twice.

Cure: (affine) linearity.

Substitution in function parameters

$$I(1, f) := \mathsf{id}$$

$$I(\mathsf{S}_i(x), f) := f \circ I(x, f)$$

$I(x, d) = d^{(|x|-1)}$: exponential.

Corresponding proofs

for I: $\forall x.\forall y_1 \exists y_2\, |y_2| = 2|y_1| \to \forall z \exists y\, |y| = 2^{|x|-1} + |z|$

for d: $\forall y_1 \exists y_2\, |y_2| = 2|y_1|$

are unproblematic, but we need to forbid applying a cut.

Cure: ramification: we can only prove $\bar{\forall} y_1 \exists y_2\, |y_2| = 2|y_1|$.

Formulas

Assume fixed predicate symbols $P, Q, \ldots$ of fixed arity.

Special predicate symbols: $=_\rho$.

$$A, B ::= P(\vec{r}) \mid A \to B \mid A \multimap B \mid A \otimes B \mid A \wedge B \mid$$
$$\tilde{\forall} x^\rho A \mid \overline{\forall} x^\rho A \mid \forall x^\rho A \mid \exists x^\rho A$$

Define $\bot :\equiv (\mathsf{tt} = \mathsf{ff})$, $\neg A :\equiv A \multimap \bot$.

Notice: conjunction is "weak"; $A \wedge B \multimap A$ is provable, but $(A \multimap B \multimap C) \multimap (A \wedge B \multimap C)$ is not.

Computational content $\tau(A)$ of a formula A

$$\begin{array}{ll}
\tau(P(\vec{r})) & := \mathbf{U} \\
\tau(A \to B) & := \tau(A) \to \tau(B) \\
\tau(A \multimap B) & := \tau(A) \multimap \tau(B) \\
\tau(A_0 \otimes A_1) & := \tau(A_0) \otimes \tau(A_1) \\
\tau(A_0 \wedge A_1) & := \tau(A_0) \times \tau(A_1)
\end{array}$$

$$\begin{array}{ll}
\tau(\tilde{\forall} x^\rho A) & := \tau(A) \\
\tau(\overline{\forall} x^\rho A) & := \rho \to \tau(A) \\
\tau(\forall x^\rho A) & := \rho \multimap \tau(A) \\
\tau(\exists x^\rho A) & := \rho \otimes \tau(A)
\end{array}$$

Computationally irrelevant (c.i.) formulas

A is c.i. $\Longleftrightarrow$

A contains no $\exists x^{\rho} B$ in a strictly positive position.

C.i. formulas are also called "Harrop formulas".

Notice: A is c.i. iff $\tau(A)^{c} = \mathbf{U}$ (^{c} means "cleaning").

Derivations $\Pi \mid \Gamma \mid \Delta \mid \Sigma \vdash M : A$

Raw proof terms:

$$M, N ::= u^A \mid c \mid \lambda u^A M \mid \lambda x^\rho M \mid MN \mid Mr$$

Split context: $\Theta = \Pi \mid \Gamma \mid \Delta \mid \Sigma$.

Π	passive part
Γ	complete part
Δ	incomplete higher type part
Σ	incomplete ground type part

Proof rules

The relation $\Pi \mid \Gamma \mid \Delta \mid \Sigma \vdash M : A$ is defined inductively:

- If $u^A \in \Gamma, \Delta, \Sigma$ and $\mathsf{FV}(A) \subseteq \Pi, \Gamma, \Delta, \Sigma$, then

$$\Pi \mid \Gamma \mid \Delta \mid \Sigma \vdash u^A : A.$$

- Also, for any axiom c,

$$\Pi \mid \Gamma \mid \Delta \mid \Sigma \vdash c : A.$$

Proof rules (ctd.)

$$\frac{\Pi \mid \Gamma, u^A \mid \Delta \mid \Sigma \vdash M : B}{\Pi \mid \Gamma \mid \Delta \mid \Sigma \vdash \lambda u^A M : A \to B} \quad (\to^+)$$

$$\frac{\Pi \mid \Gamma \mid \Delta, u^A \mid \Sigma \vdash M : B}{\Pi \mid \Gamma \mid \Delta \mid \Sigma \vdash \lambda u^A M : A \multimap B} \quad (\multimap^+_{\geq 1})$$

$$\frac{\Pi \mid \Gamma \mid \Delta \mid \Sigma, u^A \vdash M : B}{\Pi \mid \Gamma \mid \Delta \mid \Sigma \vdash \lambda u^A M : A \multimap B} \quad (\multimap^+_0)$$

Proof rules (ctd.)

$$\frac{\Pi \mid \Gamma \mid \Delta \mid \Sigma \vdash M : A \to B \qquad \Pi \mid \Gamma \mid \cdot \mid \cdot \vdash N : A}{\Pi \mid \Gamma \mid \Delta \mid \Sigma \vdash MN : B} \; (\to^-)$$

$$\frac{\Pi \mid \Gamma \mid \Delta_1 \mid \Sigma \vdash M : A \multimap B \qquad \Pi \mid \Gamma \mid \Delta_2 \mid \Sigma \vdash N : A}{\Pi \mid \Gamma \mid \Delta_1, \Delta_2 \mid \Sigma \vdash MN : B} \; (\multimap^-)$$

Proof rules (ctd.)

$$\frac{\Pi, x \mid \Gamma \mid \Delta \mid \Sigma \vdash M : A \qquad \texttt{VarCond}}{\Pi \mid \Gamma \mid \Delta \mid \Sigma \vdash \lambda x^{\rho} M : \tilde{\forall} x^{\rho} A} \qquad (\tilde{\forall}^{+})$$

$$\frac{\Pi \mid \Gamma, x \mid \Delta \mid \Sigma \vdash M : A \qquad \texttt{VarCond}}{\Pi \mid \Gamma \mid \Delta \mid \Sigma \vdash \lambda x^{\rho} M : \bar{\forall} x^{\rho} A} \qquad (\bar{\forall}^{+})$$

$$\frac{\Pi \mid \Gamma \mid \Delta, x \mid \Sigma \vdash M : A \qquad \texttt{VarCond}}{\Pi \mid \Gamma \mid \Delta \mid \Sigma \vdash \lambda x^{\rho} M : \forall x^{\rho} A} \qquad (\forall^{+}_{\geq 1})$$

$$\frac{\Pi \mid \Gamma \mid \Delta \mid \Sigma, x \vdash M : A \qquad \texttt{VarCond}}{\Pi \mid \Gamma \mid \Delta \mid \Sigma \vdash \lambda x^{\rho} M : \forall x^{\rho} A} \qquad (\forall^{+}_{0})$$

Proof rules (ctd.)

$$\frac{\Pi \mid \Gamma \mid \Delta \mid \Sigma \vdash M : \tilde{\forall} x^{\rho} A \qquad \mathsf{FV}(r) \subseteq \Pi, \Gamma, \Delta, \Sigma}{\Pi \mid \Gamma \mid \Delta \mid \Sigma \vdash Mr : A[x{:=}r]} \; (\tilde{\forall}^{-})$$

$$\frac{\Pi \mid \Gamma \mid \Delta \mid \Sigma \vdash M : \bar{\forall} x^{\rho} A \qquad \Gamma \mid \cdot \mid \cdot \vdash r : \rho}{\Pi \mid \Gamma \mid \Delta \mid \Sigma \vdash Mr : A[x{:=}r]} \; (\bar{\forall}^{-})$$

$$\frac{\Pi \mid \Gamma \mid \Delta_1 \mid \Sigma \vdash M : \forall x^{\rho} A \qquad \Gamma \mid \Delta_2 \mid \Sigma \vdash r : \rho}{\Pi \mid \Gamma \mid \Delta_1, \Delta_2 \mid \Sigma \vdash Mr : A[x{:=}r]} \; (\forall^{-})$$

Proof rules (ctd.)

Passification rule:

$$\frac{\Pi \mid \Gamma \mid \Delta \mid \Sigma \vdash M : A \qquad \tau(A)^c = \mathbf{U}}{\Pi, \Gamma, \Delta, \Sigma \mid \Gamma' \mid \Delta' \mid \Sigma' \vdash M : A}$$

Contraction rule:

$$\frac{\Pi, u^A \mid \Gamma, v^A \mid \Delta \mid \Sigma \vdash M : B}{\Pi \mid \Gamma, v^A \mid \Delta \mid \Sigma \vdash M[u^A := v^A] : B}$$

and similarly for Δ and Σ.

Induction and cases axioms

Induction axioms $\mathsf{Ind}_{l,A}$, for A linear and ρ ground:

$$\bar{\forall} l^{\mathbf{L}(\rho)}.(\bar{\forall} x^{\rho} \bar{\forall} l^{\mathbf{L}(\rho)}.A \multimap A[l{:=}\mathsf{cons}_{\rho}(x,l)]) \to A[l{:=}\mathsf{nil}_{\rho}] \multimap A$$

Cases axioms $\mathsf{Cases}_{l,A}$ and If_A, for A linear:

$$\forall l^{\mathbf{L}(\rho)}.A[l{:=}\mathsf{nil}_{\rho}] \wedge \forall x^{\rho} \forall l^{\mathbf{L}(\rho)} A[l{:=}\mathsf{cons}_{\rho}(x,l)] \multimap A$$

$$\forall p^{\mathbf{B}}.A[p{:=}\mathsf{tt}] \wedge A[p{:=}\mathsf{ff}] \multimap A$$

Logical axioms

$$A \multimap B \multimap A \otimes B$$

$$A \otimes B \multimap (A \multimap B \multimap C) \multimap C$$

$$(\vec{C} \multimap A) \multimap (\vec{C} \multimap B) \multimap \vec{C} \multimap A \wedge B$$

$$A_0 \wedge A_1 \multimap A_i$$

$$\forall x.A \multimap \exists x A,$$

$$\exists x A \multimap (\forall x.A \multimap B) \multimap B \quad \text{if } x \notin \mathsf{FV}(B)$$

$$\bot \multimap P(\vec{r})$$

and equality axioms, elimination axioms for $\otimes$.

Derived rules

$$\frac{\Pi \mid \Gamma \mid \Delta \mid \Sigma \vdash A}{\Pi, \Pi' \mid \Gamma, \Gamma' \mid \Delta, \Delta' \mid \Sigma, \Sigma' \vdash A} \quad \text{(Weakening)}$$

$$\frac{\Pi, \Pi' \mid \Gamma \mid \Delta \mid \Sigma \vdash A}{\Pi \mid \Gamma, \Pi' \mid \Delta \mid \Sigma \vdash A} \quad \text{(Activation)}$$

$$\frac{\Pi \mid \Gamma \mid \Delta, \Delta' \mid \Sigma, \Sigma' \vdash A}{\Pi \mid \Gamma, \Delta', \Sigma' \mid \Delta \mid \Sigma \vdash A} \quad \text{(Completion)}$$

Remarks

$A \multimap B$ stronger than $A \to B$: $(\multimap^-)$ has fewer restrictions than $(\to^-)$.

$\tilde{\forall}$-propositions are stronger than $\forall$-propositions, which in turn are stronger than $\bar{\forall}$-propositions.

$$\vdash (A \multimap B) \multimap (A \to B),$$

$$\vdash \tilde{\forall} x A \multimap \forall x A,$$

$$\vdash \forall x A \multimap \bar{\forall} x A.$$

Extracted term of a derivation

Given $\Theta \vdash M : A$, want $[\![\Theta \vdash M : A]\!]$ (abbreviated $[\![M]\!]$) such that form

$$\Pi \mid \Gamma, u_1^{B_1}, \ldots, u_n^{B_n} \mid \Delta, v_1^{C_1}, \ldots, v_m^{C_m} \mid \Sigma, w_1^{D_1}, \ldots, w_k^{D_k}$$
$$\vdash M : A$$

we can conclude

$$\Gamma, x_{u_1}, \ldots, x_{u_n} \mid \Delta, x_{v_1}, \ldots, x_{v_m} \mid \Sigma, x_{w_1}, \ldots, x_{w_k}$$
$$\vdash [\![M]\!] : \tau(A).$$

Extracted term $[\![M]\!]$

$$[\![u^A]\!] := x_u^{\tau(A)} \quad (x_u^{\tau(A)} \text{ associated with } u^A)$$

$$[\![\lambda u^A M]\!] := \lambda x_u^{\tau(A)} [\![M]\!]$$

$$[\![MN]\!] := [\![M]\!][\![N]\!]$$

$$[\![\lambda x^\rho M]\!] := \begin{cases} [\![M]\!] & \text{if the last rule is } (\tilde{\forall}^+) \\ \lambda x^\rho [\![M]\!] & \text{if the last rule is } (\bar{\forall}^+) \text{ or } (\forall^+) \end{cases}$$

$$[\![Mr]\!] := \begin{cases} [\![M]\!] & \text{if the last rule is } (\tilde{\forall}^-) \\ [\![M]\!]r & \text{if the last rule is } (\bar{\forall}^-) \text{ or } (\forall^-) \end{cases}$$

$$[\![M]\!] := \varepsilon^{\tau(A)} \quad \text{if the last rule of } M\colon A \text{ is (Passif.)}$$

and hence $\tau(A)^c = \mathbf{U}$

Extracted term for induction and cases axioms

$\bar{\forall} l^{\mathbf{L}(\rho)}.(\bar{\forall} x^{\rho} \bar{\forall} l^{\mathbf{L}(\rho)}.A \multimap A[l{:=}\mathsf{cons}_{\rho}(x,l)]) \to A[l{:=}\mathsf{nil}_{\rho}] \multimap A$

$\forall l^{\mathbf{L}(\rho)}.A[l{:=}\mathsf{nil}_{\rho}] \wedge \forall x^{\rho} \forall l^{\mathbf{L}(\rho)} A[l{:=}\mathsf{cons}_{\rho}(x,l)] \multimap A$

$\forall p^{\mathbf{B}}.A[p{:=}\mathsf{tt}] \wedge A[p{:=}\mathsf{ff}] \multimap A$

have the extracted terms (for A linear, $\tau := \tau(A)$, and ρ ground in the induction axiom)

$$[\![\mathsf{Ind}_{l,A}]\!] := \mathcal{R}_{\tau}^{\mathbf{L}(\rho)} : \mathbf{L}(\rho) \to (\rho \to \mathbf{L}(\rho) \to \tau \multimap \tau) \to \tau \multimap \tau$$

$$[\![\mathsf{Cases}_{l,A}]\!] := c_{\tau} : \mathbf{L}(\rho) \multimap \tau \times (\rho \multimap \mathbf{L}(\rho) \multimap \tau) \multimap \tau$$

$$[\![\mathsf{If}_{A}]\!] := \mathsf{if}_{\tau} : \mathbf{B} \multimap \tau \times \tau \multimap \tau$$

Extracted terms for axioms (ctd.)

Write $[\![A]\!]$ for $[\![c\colon A]\!]$. If $\tau(A)^c = \mathbf{U}$, then $[\![A]\!] := \varepsilon^{\tau(A)}$ (ε^ρ some closed term of type ρ).

$$
\begin{aligned}
[\![A \multimap B \multimap A \otimes B]\!] &:= \otimes^+_{\tau(A),\tau(B)} \\
[\![A \otimes B \multimap (A \multimap B \multimap C) \multimap C]\!] &:= \otimes^-_{\tau(A),\tau(B),\tau(C)} \\
[\![(\vec{C} \multimap A) \multimap (\vec{C} \multimap B) \multimap \vec{C} \multimap A \wedge B]\!] &:= \times^+_{\tau(A),\tau(B),\tau(\vec{C})} \\
[\![A \wedge B \multimap A]\!] &:= \mathsf{fst}_{\tau(A),\tau(B)} \\
[\![A \wedge B \multimap B]\!] &:= \mathsf{snd}_{\tau(A),\tau(B)}
\end{aligned}
$$

Extracted terms for axioms (ctd.)

$$
\begin{array}{ll}
[\![\forall x.A \multimap \exists x A]\!] & := \otimes^{+}_{\rho,\tau(A)} \\
[\![\exists x A \multimap (\forall x.A \multimap B) \multimap B]\!] & := \otimes^{-}_{\rho,\tau(A),\tau(B)} \\
[\![\forall x^{\rho} \forall y^{\sigma} A[z := \otimes^{+} xy] \multimap \forall z^{\rho \otimes \sigma} A]\!] & := \lambda f \lambda z.\, \otimes^{-} zf \\
[\![\tilde{\forall} x^{\rho} \tilde{\forall} y^{\sigma} A[z := \otimes^{+} xy] \multimap \tilde{\forall} z^{\rho \otimes \sigma} A]\!] & := \mathsf{id}_{\tau(A)}
\end{array}
$$

Soundness of typing

Theorem. Assume

$$\Pi \mid \Gamma, u_1^{B_1}, \ldots, u_n^{B_n} \mid \Delta, v_1^{C_1}, \ldots, v_m^{C_m} \mid \Sigma, w_1^{D_1}, \ldots, w_k^{D_k}$$
$$\vdash M : A.$$

Then

$$\Gamma, x_{u_1}, \ldots, x_{u_n} \mid \Delta, x_{v_1}, \ldots, x_{v_m} \mid \Sigma, x_{w_1}, \ldots, x_{w_k}$$
$$\vdash [\![M]\!] : \tau(A).$$

Proof. Inspection of the proof rules and the extracted terms for the c.r. axioms. □

Realizability

$$
\begin{aligned}
r\,\underline{\mathbf{mr}}\,P(\vec{s}) &:= P(\vec{s}) \\
r\,\underline{\mathbf{mr}}\,(A{\to}B) := r\,\underline{\mathbf{mr}}\,(A{\multimap}B) &:= \forall x.x\,\underline{\mathbf{mr}}\,A \to rx\,\underline{\mathbf{mr}}\,B \\
r\,\underline{\mathbf{mr}}\,(A \otimes B) &:= r\pi_0\,\underline{\mathbf{mr}}\,A \wedge r\pi_1\,\underline{\mathbf{mr}}\,B \\
r\,\underline{\mathbf{mr}}\,(A \wedge B) &:= r\mathrm{tt}\,\underline{\mathbf{mr}}\,A \wedge r\mathrm{ff}\,\underline{\mathbf{mr}}\,B. \\
r\,\underline{\mathbf{mr}}\,\tilde{\forall}xA &:= \forall x.r\,\underline{\mathbf{mr}}\,A \\
r\,\underline{\mathbf{mr}}\,\bar{\forall}xA := r\,\underline{\mathbf{mr}}\,\forall xA &:= \forall x.rx\,\underline{\mathbf{mr}}\,A \\
r\,\underline{\mathbf{mr}}\,\exists xA &:= r\pi_0\,\underline{\mathbf{mr}}\,A[x{:=}r\pi_1]
\end{aligned}
$$

Remark: $r\,\underline{\mathbf{mr}}\,A$ is an $\mathbf{HA}$-formula, and $\mathbf{HA} \vdash A \leftrightarrow r\,\underline{\mathbf{mr}}\,A$ if A is $\exists$-free.

Soundness

Theorem. Assume $\Theta \vdash M : A$. Then there is an $\mathbf{HA}$-derivation of

$$[\![M]\!]\,\underline{\mathbf{mr}}\,A$$

from assumptions $x_u\,\underline{\mathbf{mr}}\,B$ for $u^B \in \Theta$.

Proof. By induction on $\Theta \vdash M : A$. □

$\mathbf{LHA}$ and its provably recursive functions

Recall the induction scheme:

$$\bar{\forall} l^{\mathbf{L}(\rho)}.(\bar{\forall} x^{\rho} \bar{\forall} l^{\mathbf{L}(\rho)}.A \multimap A[l{:=}\mathsf{cons}_{\rho}(x,l)]) \to A[l{:=}\mathsf{nil}_{\rho}] \multimap A$$

provided A linear and ρ ground (A is linear if A contains no complete quantification $\bar{\forall}$ and no $\to$.) For instance, the proofs sketched above can easily be formalized in $\mathbf{LHA}$.

Theorem. A function is provably recursive in $\mathbf{LHA}$ if and only if it is computable in polynomial time.

References

Hofmann. Typed λ-calculi for poly-time computation. Habilitation thesis, TU Darmstadt, 1998

Bellantoni & Niggl & S., Higher Type Recursion, Ramification and Polynomial Time. APAL 2000

S. & Bellantoni, Feasible Computation w. Higher Types. 2002

Aehlig & Berger & Hofmann & S., An arithmetic for non-size-increasing polynomial-time computation. Submitted