Syntactical Strong Normalization of Intersection Types with Term Rewriting Rules

Andreas Abel

Department of Computer Science
Ludwig-Maximilians-University Munich

4th International Workshop on Higher-Order Rewriting
Paris, France
June 25, 2007

Normalization Proofs in Weak Meta-Theories

- Weak systems can be shown consistent in (almost as) weak meta-theories.
- Normalization implies usually consistency.
- Tait's *semantical* normalization proof for the simply-typed λ-calculus (STL) is beyond Peano Arithmetic.
- Combinatorial/arithmetical/*syntactical* normalization proofs for the STL have long been known, e.g., J.J.Levy (1974), Girard, Lafont, Taylor (1989).
- But were better understood for sequent calculus (cut elimination).
- No such proof is known for System F; for Gödel's T, ε_0-induction is necessary.

Recent Syntactical Normalization Proofs

- Intersection types: Bucciarelli et al. (LICS'99), Matthes (ITRS'00), Valentini (AML 2001), David (APAL 2001).
- Classical natural deduction: David, Nour.
- Positive recursive types: David, Nour (TLCA'07).
- This work: singleton types with term rewriting.

My Interest on Normalization in Weak Metatheories

- Research: normalization proofs in Twelf.
- Twelf: higher-order abstract syntax.
- Comfortable variable handling, but no recursive functions.
- Only Π_2 statements ($\forall x \exists y A$).
- Termination orders: lexicographic extension of structural order, i.e., $< \omega^\omega$.

Content

- A normalizer for simply-typed λ
- A normalization proof for intersection types
- Extension to term rewriting

A Normalizer for Simply-Typed Lambda-Calculus

- A structurally recursive normalizer:

$$
\begin{array}{lcl}
\mathsf{nf}(x) & = & x \\
\mathsf{nf}(\lambda x\!:\!A.t) & = & \lambda x\!:\!A.\,\mathsf{nf}(t) \\
\mathsf{nf}(r\,s) & = & \mathsf{nf}(r)@\mathsf{nf}(s) \\
\\
x\,\vec{w}@w & = & x\,\vec{w}\,w \\
(\lambda x\!:\!A.v)@w & = & [w^A/x]v
\end{array}
$$

- “Hereditary” subst. of one normal form into another terminates.
- $[(\lambda y\!:\!A.\lambda z\!:\!B.w)^{A\to B\to C}/x]x\,u\,v$ triggers two new substitutions

$$
\begin{array}{l}
[u^A/y]\lambda z\!:\!B.w \\
[v^B/z]w'
\end{array}
$$

  but A and B are smaller than $A \to B \to C$.
- $[w^A/x]v$ structurally recursive in (A, v).

Hereditary Substitutions

- Normalizing substitution of normal forms: $[s^A/x]t$

$$
\begin{array}{lcll}
[s^A/x]x & = & s^A & \\
[s^A/x]y & = & y & \text{if } x \neq y \\
[s^A/x](\lambda y\!:\!B.r) & = & \lambda y\!:\!B.\,[s^A/x]r & \text{where } y \text{ fresh for } s, x \\
[s^A/x](t\,u) & = & ([\hat{u}^B/y]r')^C & \text{if } \hat{t} = (\lambda y\!:\!B'.r')^{B\to C} \\
 & & \hat{t}\,\hat{u} & \text{otherwise} \\
\text{where } \hat{t} & = & [s^A/x]t & \\
\hat{u} & = & [s^A/x]u &
\end{array}
$$

- Invariant: $|B \to C| \leq |A|$ in line 4.

Inductive Characterization of Strongly Normalizing Terms

- Following Joachimski and Matthes (2003)
- $\Gamma \vdash t \Uparrow A$ means t *is strongly normalizing of type* A.
- $\Gamma \vdash t \downarrow^x A$ means t *is sn and neutral of type* A.
- Rules:

$$\frac{(x : A) \in \Gamma}{\Gamma \vdash x \downarrow^x A} \qquad \frac{\Gamma \vdash r \downarrow^x A \to B \qquad \Gamma \vdash s \Uparrow A}{\Gamma \vdash r\,s \downarrow^x B}\ \texttt{sne\_app}$$

$$\frac{\Gamma \vdash r \downarrow^x A}{\Gamma \vdash r \Uparrow A}\ \texttt{sn\_ne}$$

$$\frac{\Gamma, x : A \vdash t \Uparrow B}{\Gamma \vdash \lambda x.t \Uparrow A \to B}\ \texttt{sn\_lam} \qquad \frac{\Gamma \vdash s \Uparrow A \qquad \Gamma \vdash [s/x]r\,\vec{s} \Uparrow C}{\Gamma \vdash (\lambda x.r)\,s\,\vec{s} \Uparrow C}\ \texttt{sn\_exp}$$

Closure of S.N. Terms under Application

- Lemma: Let $\mathcal{D} :: \Gamma \vdash s \Uparrow A$.
  1. If $\mathcal{E} :: \Gamma \vdash r \Uparrow A \to C$ then $\Gamma \vdash r\, s \Uparrow C$.
  2. If $\mathcal{E} :: \Gamma, x\!:\!A \vdash t \Uparrow C$, then $\Gamma \vdash [s/x]t \Uparrow C$.
  3. If $\mathcal{E} :: \Gamma, x\!:\!A \vdash t \downarrow^x C$, then $\Gamma \vdash [s/x]t \Uparrow C$ and C is a subexpression of A.
  4. If $\mathcal{E} :: \Gamma, x\!:\!A \vdash t \downarrow^y C$ with $x \neq y$, then $\Gamma \vdash [s/x]t \downarrow^y C$.
- Proof: Simultaneously by main induction on type A (for part 3) and side induction on the derivation $\mathcal{E}$.
- Similar to Girard, Lafont and Taylor (1989): Lexicographic induction on highest degree (=type) of a redex and the number of redexes of highest degree.

Intersection Types

- STL + additional typing rules:

$$\frac{\Gamma \vdash t : A \qquad \Gamma \vdash t : B}{\Gamma \vdash t : A \cap B} \qquad \frac{\Gamma \vdash t : A \cap B}{\Gamma \vdash t : A} \qquad \frac{\Gamma \vdash t : A \cap B}{\Gamma \vdash t : B}$$

- Exactly the s.n. terms are typable.
- Additional rules for inductive characterization of s.n.:

$$\frac{\Gamma \vdash n \downarrow^{x} A \cap B}{\Gamma \vdash n \downarrow^{x} A} \qquad \frac{\Gamma \vdash n \downarrow^{x} A \cap B}{\Gamma \vdash n \downarrow^{x} B}$$

$$\frac{\Gamma \vdash t \Uparrow A \qquad \Gamma \vdash t \Uparrow B}{\Gamma \vdash t \Uparrow A \cap B}$$

Strong Normalization Using Domain Theory

- Berger (CiE'05) proves s.n. of bar recursion using domain theory.
- Main lemma: t is s.n. if $[\![t]\!] \neq \bot$.
- Coquand, Spiwack (LICS'06) simplify his argument.
- They use a λ filter model: $[\![t]\!]$ is the set of types t can be assigned in a simple intersection type system.
- Construction of model requires strong metatheory.
- This work: proof of main lemma in weak metatheory.

Type Assignment for Term Rewriting

- Example:

$$
\begin{array}{lcl}
\mathsf{add}\, y\, 0 & \longrightarrow & y \\
\mathsf{add}\, y\, (\$x) & \longrightarrow & \$(\mathsf{add}\, y\, x) \\
\\
\mathsf{add} & : & 0 \to 0 \to 0 \\
 & \cap & 0 \to \$0 \to \$0 \\
 & \cap & \$0 \to 0 \to \$0 \\
 & \cap & \$0 \to \$0 \to \$\$0 \\
 & \cap & \dots
\end{array}
$$

Types Approximating Function Behavior

Ground types

a, b, c	$::=$	E	exception
	$\mid$	$0 \mid \$a$	zero and successor singletons

Types

A, B, C	$::=$	a	ground type
	$\mid$	$\bigcap_{i \in I}(A_i \to B_i)$	finite funct. descr., all A_i different

- Intersection and subtyping definable.
- Measure: $|a| = 0$ and $|\bigcap_{i \in I}(A_i \to B_i)| = \max\{|A_i| + 1, |B_i| \mid i \in I\}$.

Typing

$$\frac{}{\Gamma \vdash 0 : 0} \qquad \frac{\Gamma \vdash r : a}{\Gamma \vdash \$r : \$a}$$

$$\frac{\Gamma \vdash r : 0 \qquad \Gamma \vdash \underline{z} : C}{\Gamma \vdash f(r) : C}\ f(0) \longrightarrow \underline{z}$$

$$\frac{\Gamma \vdash r : \$a \qquad \Gamma, x\!:\!a \vdash \underline{s} : C}{\Gamma \vdash f(r) : C}\ f(\$x) \longrightarrow \underline{s}$$

$$\frac{\Gamma \vdash r : A}{\Gamma \vdash f(r) : \mathrm{E}}\ A \neq 0, \$a$$

$$\frac{\Gamma \vdash r : A \qquad \Gamma \vdash r : B}{\Gamma \vdash r : A \cap B} \qquad \frac{\Gamma \vdash r : A \qquad A \subseteq B}{\Gamma \vdash r : B}$$

What about our Termination Argument!?

- Neutral terms in STL: The types of the s_i in $x\, s_1 \dots s_n$ are smaller than the type of x.
- With TR: The type of $f(x)$ might be bigger than the type of x.
- Problematic for substituting into $f(x)\, s_1 \dots s_n$.
- Solution: Distinguish *atomic terms* $x\, \vec{s}$ from *neutral terms* $E[f(x\, \vec{s})]$.
- Evaluation contexts:

$$E[] ::= [] \mid E[]\, s \mid f(E[]).$$

S.N. Atomic and Neutral Terms

- SN: Atomic terms.

$$\frac{}{\Gamma \vdash x \downarrow \Gamma(x)} \qquad \frac{\Gamma \vdash r \downarrow \bigcap_{i \in I}(A_i \to B_i) \qquad \Gamma \vdash s \Uparrow A_j \text{ for all } j \in J}{\Gamma \vdash r\, s \downarrow \bigcap_{j \in J} B_j}$$

- SN: Neutral terms.

$$\frac{\Gamma \vdash r \downarrow A \qquad A \subseteq B}{\Gamma \vdash r \Downarrow B} \qquad \frac{\Gamma \vdash r \Downarrow 0 \qquad \Gamma \vdash \underline{z}\, \vec{s} \Uparrow C}{\Gamma \vdash f(r)\, \vec{s} \Downarrow C}\ f(0) \longrightarrow \underline{z}$$

$$\frac{\Gamma \vdash r \Downarrow \$a \qquad \Gamma, x : a \vdash \underline{s}\, \vec{s} \Uparrow C}{\Gamma \vdash f(r)\, \vec{s} \Downarrow C}\ f(\$x) \longrightarrow \underline{s}$$

S.N. Terms

- Neutral terms.

$$\frac{\Gamma \vdash r \Downarrow A \qquad A \subseteq B}{\Gamma \vdash r \Uparrow B}$$

- Introductions.

$$\frac{\Gamma, x : A_i \vdash t \Uparrow B_i \text{ for all } i \in I}{\Gamma \vdash \lambda x t \Uparrow \bigcap_{i \in I}(A_i \to B_i)} \qquad \frac{}{\Gamma \vdash 0 \Uparrow 0} \qquad \frac{\Gamma \vdash r \Uparrow a}{\Gamma \vdash \$r \Uparrow \$a}$$

- Blocked terms.

$$\frac{\Gamma \vdash r \Uparrow A}{\Gamma \vdash f(r) \Uparrow \mathrm{E}}\ A \neq 0, \$a \qquad \frac{\Gamma \vdash r \Uparrow \mathrm{E} \qquad \Gamma \vdash s \Uparrow A}{\Gamma \vdash r\, s \Uparrow \mathrm{E}}$$

S.N. Terms (continued)

- Weak head expansions.

$$\frac{\Gamma \vdash s \Uparrow A \qquad \Gamma \vdash E[[s/x]t] \Uparrow C}{\Gamma \vdash E[(\lambda x t)\, s] \Uparrow C}$$

$$\frac{\Gamma \vdash E[\underline{z}] \Uparrow C}{\Gamma \vdash E[f(0)] \Uparrow C}\ f(0) \longrightarrow \underline{z}$$

$$\frac{\Gamma \vdash r \Uparrow A \qquad \Gamma \vdash E[[r/x]\underline{s}] \Uparrow C}{\Gamma \vdash E[f(\$r)] \Uparrow C}\ f(\$x) \longrightarrow \underline{s}$$

- Cannot treat higher-order datatypes like tree ordinals (yet!?)
- But sufficient for bar recursion example.

Conclusion

- Technique extends also to predicative polymorphism.
- Current work: primitive recursion (needs ordinals up to ω^ω).
- Leads into “Munich” proof theory (ordinal analysis).

References

- David, Nour: Arithmetical normalization proofs of λ, $\lambda\mu$, positive recursive types (TLCA 2007).
- Matthes, Joachimski, AML 2003: Syntactic normalization.
- Watkins et al, TYPES 2003: Hereditary substitutions.
- Abel, Weak Normalization for λ in Twelf (LFM 2004).
- Schürmann, Sarnat: Logical Relation Proofs in Twelf (HOR 2007 invited talk).