Hereditary Substitution for the λΔ-Calculus

Harley Eades and Aaron Stump
Computer Science
THE UNIVERSITY OF IOWA

The Big Picture

- Goal: Prove weak normalization of the $\lambda\Delta$-calculus.
- Tool of choice: hereditary substitution.
- Novelty: normalization by hereditary substitution has never been applied to any classical type theories.

Why Hereditary Substitution

- It provides a directly defined substitution which preserves normal forms.

Overview

- Hereditary substitution and STLC.
- The λΔ-calculus.
- A naive extension of hereditary substitution to the λΔ-calculus.
- The correct extension.
- Normalization of the λΔ-calculus.

The Simply Typed λ-Calculus

Syntax:

$$
\begin{array}{rrcl}
\text{(Types)} & T, A, B, C & ::= & \bot \mid b \mid A \to B \\
\text{(Terms)} & t & ::= & x \mid \lambda x : T.t \mid t_1\ t_2 \\
\text{(Normal Forms)} & n, m & ::= & x \mid \lambda x : T.n \mid h\ n \\
\text{(Heads)} & h & ::= & x \mid h\ n \\
\text{(Contexts)} & \Gamma & ::= & \cdot \mid x : A \mid \Gamma_1, \Gamma_2
\end{array}
$$

The Simply Typed λ-Calculus

Typing:

$$\frac{}{\Gamma, x : A \vdash x : A} \ \textsc{Ax} \qquad \frac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x : A.t : A \to B} \ \textsc{Lam}$$

$$\frac{\begin{array}{c}\Gamma \vdash t_2 : A \\ \Gamma \vdash t_1 : A \to B\end{array}}{\Gamma \vdash t_1\ t_2 : B} \ \textsc{App}$$

Reduction:

$$\frac{}{(\lambda x : T.t)\ t' \rightsquigarrow [t'/x]t} \ \textsc{Beta}$$

Hereditary Substitution

- Syntax: $[t/x]^A t$
- Usual termination order: (A, t')
- Like ordinary capture-avoiding substitution.
- Except, if the substitution introduces a redex, then that redex is recursively reduced.
- Example: $[\lambda z : \mathrm{b}.z/x]^{\mathrm{b}\to\mathrm{b}}(x\ y)(\approx ((\lambda z : \mathrm{b}.z)\ y \approx [y/z]^{\mathrm{b}} z) = y$

Hereditary Substitution

Definition. *We define the partial function* ctype *which computes the type of an application in head normal form. It is defined as follows:*

$$\mathsf{ctype}_T(x, x) = T$$

$$\mathsf{ctype}_T(x, t_1\ t_2) = T''$$
$$\textit{Where } \mathsf{ctype}_T(x, t_1) = T' \to T''.$$

Lemma (Properties of ctype).

i. *If* $\mathsf{ctype}_T(x, t) = T'$ *then* $\mathsf{head}(t) = x$ *and* $T' \leq T$.

ii. *If* $\Gamma, x : T, \Gamma' \vdash t : T'$ *and* $\mathsf{ctype}_T(x, t) = T''$ *then* $T' \equiv T''$.

Hereditary Substitution

$[t/x]^A x = t$

$[t/x]^A y = y$

$[t/x]^A (\lambda y : A'.t') = \lambda y : A'.([t/x]^A t')$

$[t/x]^A (t_1\ t_2) = ([t/x]^A t_1)\,([t/x]^A t_2)$
Where $([t/x]^A t_1)$ is not a λ-abstraction, or t_1 is a λ-abstraction.

$[t/x]^A (t_1\ t_2) = [s'_2/y]^{A''} s'_1$
Where $([t/x]^A t_1) = \lambda y : A''.s'_1$ for some y, s'_1 and A'',
$[t/x]^A t_2 = s'_2$, and $\mathsf{ctype}_A(x, t_1) = A'' \to A'$.

Properties of Hereditary Substitution

Lemma (Total and Type Preserving). *Suppose* $\Gamma \vdash t : T$ *and* $\Gamma, x : T, \Gamma' \vdash t' : T'$. *Then there exists a term* t'' *such that* $[t/x]^T t' = t''$ *and* $\Gamma, \Gamma' \vdash t'' : T'$.

Properties of Hereditary Substitution

Lemma (Normality Preserving). *If* $\Gamma \vdash n : T$ *and* $\Gamma, x : T \vdash n' : T'$ *then there exists a normal term* n'' *such that* $[n/x]^T n' = n''$.

Properties of Hereditary Substitution

Lemma (Soundness with Respect to Reduction). *If* $\Gamma \vdash t : T$ *and* $\Gamma, x : T, \Gamma' \vdash t' : T'$ *then* $[t/x]t' \leadsto^* [t/x]^T t'$.

The λΔ-Calculus

Syntax

Rehof:1994

$$
\begin{array}{rrcl}
\text{(Types)} & T, A, B, C & ::= & \cdots \\
\text{(Terms)} & t & ::= & \cdots \,|\, \Delta x : T.t \\
\text{(Normal Forms)} & n, m & ::= & \cdots \,|\, \Delta x : T.n \\
\text{(Heads)} & h & ::= & \cdots \\
\text{(Contexts)} & \Gamma & ::= & \cdots
\end{array}
$$

Negation: $\neg A =^{def} A \to \perp$

The λΔ-Calculus

Typing:

Rehof:1994

$$\cdots \quad \frac{\Gamma, x : \neg A \vdash t : \bot}{\Gamma \vdash \Delta x : \neg A . t : A} \;\textsc{Delta}$$

Reduction:

$$\cdots \quad \frac{\begin{array}{c} y \text{ fresh in } t \text{ and } t' \\ z \text{ fresh in } t \text{ and } t' \end{array}}{(\Delta x : \neg(T_1 \to T_2).t)\, t' \rightsquigarrow \Delta y : \neg T_2.[\lambda z : T_1 \to T_2.(y\,(z\,t'))/x]t} \;\textsc{StructRed}$$

Problems with a Naive Extension

- The naive extension is a simple extension to the hereditary substitution function for STLC:

$\ldots$

$$[t/x]^A(\Delta y : A'.t') = \Delta y : A'.([t/x]^A t')$$

$$[t/x]^A(t_1\ t_2) = \Delta z : \neg A'.[\lambda y : A'' \to A'.(z\ (y\ s_2))/y]^{\neg(A'' \to A')} s$$

Where $([t/x]^A t_1) = \Delta y : \neg(A'' \to A').s$ for some, y s, and $A'' \to A'$, $([t/x]^A t_2) = s_2$ for some s_2, $\mathsf{ctype}_A(x, t_1) = A'' \to A'$, and z is completely fresh.

Problems with a Naive Extension

- The naive extension is a simple extension to the hereditary substitution function for STLC:

$\ldots$

$$[t/x]^A(\Delta y : A'.t') = \Delta y : A'.([t/x]^A t')$$

$$[t/x]^{\boxed{A}}(t_1\ t_2) = \Delta z : \neg A'.[\lambda y : A'' \to A'.(z\ (y\ s_2))/y]^{\boxed{\neg(A'' \to A')}} s$$

Where $([t/x]^A t_1) = \Delta y : \neg(A'' \to A').s$ for some, y s, and $A'' \to A'$, $([t/x]^A t_2) = s_2$ for some s_2, $\mathsf{ctype}_A(x, t_1) = \boxed{A'' \to A'}$, and z is completely fresh.

- ctype tells us: $A \geq A'' \to A' < \neg(A'' \to A')$

Problems with a Naive Extension

- The naive extension is a simple extension to the hereditary substitution function for STLC:

$\ldots$

$[t/x]^A(\Delta y : A'.t') = \Delta y : A'.([t/x]^A t')$

$[t/x]^{A}(t_1\ t_2) = \Delta z : \neg A'.[\lambda y : A'' \to A'.(z\ (y\ s_2))/y]^{\neg(A'' \to A')} s$
Where $([t/x]^A t_1) = \Delta y : \neg(A'' \to A').s$ for some, y s, and $A'' \to A'$, $([t/x]^A t_2) = s_2$ for some s_2, $\mathsf{ctype}_A(x, t_1) = A'' \to A'$, and z is completely fresh.

- ctype tells us: $A \geq A'' \to A' < \neg(A'' \to A')$

How do we fix this?

- Consider the following example:

$$\begin{aligned}(\Delta q : \neg(\mathrm{b} \to \mathrm{b}).(q\ y))\ r \quad &\rightsquigarrow \quad \Delta z : \neg\mathrm{b}.[\lambda x : \mathrm{b} \to \mathrm{b}.(z\,(x\ r))/q](q\ y) \\ &= \quad \Delta z : \neg\mathrm{b}.((\lambda x : \mathrm{b} \to \mathrm{b}.(z\,(x\ r)))\ y) \\ &\rightsquigarrow \quad \Delta z : \neg\mathrm{b}.(z\,(y\ r))\end{aligned}$$

How do we fix this?

- Consider the following example:

$$
\begin{array}{rcr}
(\Delta q : \neg(\mathrm{b} \to \mathrm{b}).(q\ y))\ r & \rightsquigarrow & \Delta z : \neg \mathrm{b}.[\lambda x : \mathrm{b} \to \mathrm{b}.(z\ (x\ r))/q](q\ y) \\
 & = & \Delta z : \neg \mathrm{b}.((\lambda x : \mathrm{b} \to \mathrm{b}.(z\ (x\ r)))\ y) \\
 & \rightsquigarrow & \Delta z : \neg \mathrm{b}.(z\ (y\ r))
\end{array}
$$

How do we fix this?

- Consider the following example:

$$\begin{array}{rcl}(\Delta q : \neg(\mathrm{b} \to \mathrm{b}).(q\ y))\ r & \leadsto & \Delta z : \neg\mathrm{b}.[\lambda x : \mathrm{b} \to \mathrm{b}.(z\ (x\ r))/q](q\ y) \\ & = & \Delta z : \neg\mathrm{b}.((\lambda x : \mathrm{b} \to \mathrm{b}.(z\ (x\ r)))\ y) \\ & \leadsto & \Delta z : \neg\mathrm{b}.(z\ (y\ r))\end{array}$$

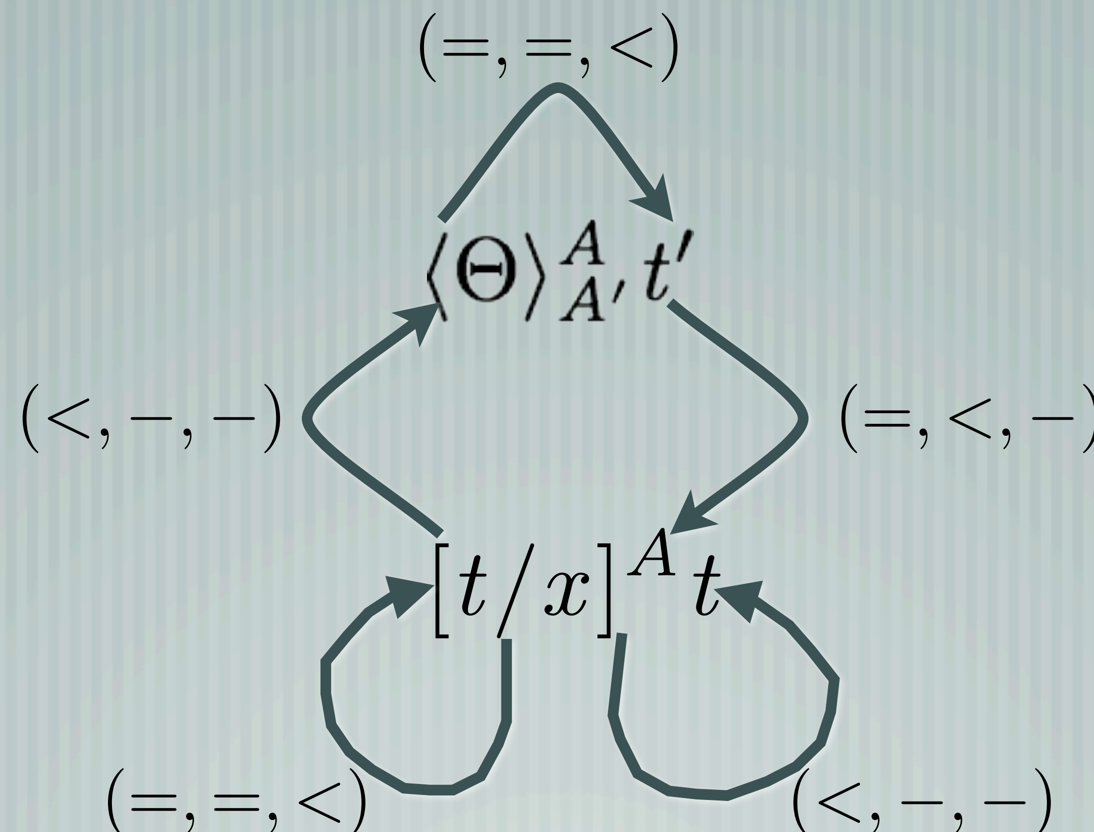
$$\begin{array}{rcl}(\Delta q : \neg(\mathrm{b} \to \mathrm{b}).(q\ y))\ r & \leadsto & \Delta z : \neg\mathrm{b}.\langle(q, z, r)\rangle(q\ y) \\ & = & \Delta z : \neg\mathrm{b}.(z\ (y\ r))\end{array}$$

A Correct Extension of Hereditary Substitution

- Hereditary structural substitution:
  - Is a multi-substitution defined by induction mutually with the hereditary substitution function.
  - Syntax: $\langle \Theta \rangle^{A}_{A'} t'$, where $\Theta ::= \cdot \mid \Theta, (y, z, t)$
  - New termination metric: (A, f, t')

A Correct Extension of Hereditary Substitution

- New termination metric: (A, f, t')



A Correct Extension of Hereditary Substitution

Variables:

$\langle\Theta\rangle^{A_1}_{A_2} x = \lambda y : A_1 \to A_2.(z\,(y\,t))$

Where $(x, z, t) \in \Theta$, for some z and t, and y is fresh in x, z, and t.

$\langle\Theta\rangle^{A_1}_{A_2} x = x$

Where $(x, z, t) \notin \Theta$ for any z or t.

A Correct Extension of Hereditary Substitution

Abstractions:

$$\langle\Theta\rangle^{A_1}_{A_2}(\lambda y : A.t) = \lambda y : A.\langle\Theta\rangle^{A_1}_{A_2} t$$

$$\langle\Theta\rangle^{A_1}_{A_2}(\Delta y : A.t) = \Delta y : A.\langle\Theta\rangle^{A_1}_{A_2} t$$

A Correct Extension of Hereditary Substitution

Applications:

$\langle\Theta\rangle^{A_1}_{A_2}(x\ t') = z\,[t/y]^{A_1}\,s$

Where $(x,z,t) \in \Theta$, $t' \equiv \lambda y : A_1.t''$, for some y and t'', and $\langle\Theta\rangle^{A_1}_{A_2} t'' = s$.

$\langle\Theta\rangle^{A_1}_{A_2}(x\ t') = z\,(\Delta z_2 : \neg A_2.s)$

Where $(x,z,t) \in \Theta$, $t' \equiv \Delta y : \neg(A_1 \to A_2).t''$, for some y and t'', and $\langle\Theta,(y,z_2,t)\rangle^{A_1}_{A_2} t'' = s$, for some fresh z_2.

$\langle\Theta\rangle^{A_1}_{A_2}(x\ t') = z\ s'$

Where $(x,z,t) \in \Theta$, t' is not an abstraction, and $\langle\Theta\rangle^{A_1}_{A_2} t' = s'$.

A Correct Extension of Hereditary Substitution

Applications:

$\langle\Theta\rangle^{A_1}_{A_2}(t_1\ t_2) = s_1\ s_2$

Where t_1 is either not a variable, or it is both a variable and $(t_1, z', t') \notin \Theta$ for any t' and z', $\langle\Theta\rangle^{A_1}_{A_2} t_1 = s_1$, and $\langle\Theta\rangle^{A_1}_{A_2} t_2 = s_2$.

A Correct Extension of Hereditary Substitution

The hereditary substitution function:

$\ldots$

$$[t/x]^A(\Delta y : A'.t') = \Delta y : A'.([t/x]^A t')$$

$$[t/x]^A(t_1\ t_2) = \Delta z : \neg A'.\langle (y, z, s_2)\rangle^{A''}_{A'}\ s$$

Where $([t/x]^A t_1) = \Delta y : \neg(A'' \to A').s$ for some $y\ s$, and $A'' \to A'$, $([t/x]^A t_2) = s_2$ for some s_2, $\mathsf{ctype}_A(x, t_1) = A'' \to A'$, and z is fresh.

A Correct Extension of Hereditary Substitution

Lemma (Totality and Type Preservation).

i. *If* $\Gamma \vdash \Theta^3 : A$ *and* $\Gamma, \Theta^1 : \neg(A \to A') \vdash t' : B$*, then there exists a term* s *such that* $\langle\Theta\rangle^A_{A'} t' = s$ *and* $\Gamma, \Theta^2 : \neg A' \vdash s : B$*.*

ii. *If* $\Gamma \vdash t : A$ *and* $\Gamma, x : A, \Gamma' \vdash t' : B$*, then there exists a term* s *such that* $[t/x]^A t' = s$ *and* $\Gamma, \Gamma' \vdash s : B$*.*

A Correct Extension of Hereditary Substitution

Lemma (Normality Preservation).

i. *If* $\mathsf{norm}(\Theta^3), \Gamma \vdash \Theta^3 : A$ *and* $\Gamma, \Theta^1 : \neg(A \to A') \vdash n' : B$*, then there exists a normal form* m *such that* $\langle\Theta\rangle^{A}_{A'} n' = m$.

ii. *If* $\Gamma \vdash n : A$ *and* $\Gamma, x : A, \Gamma' \vdash n' : B$ *then there exists a term* m *such that* $[n/x]^A n' = m$.

A Correct Extension of Hereditary Substitution

Lemma (Soundness with Respect to Reduction).

i. *If* $\Gamma \vdash \Theta^3 : A$ *and* $\Gamma, \Theta^1 : \neg(A \to A') \vdash t' : B$, *then* $\langle\Theta\rangle^{\uparrow A}_{A'}\ t' \rightsquigarrow^* \langle\Theta\rangle^{A}_{A'} t'$.

ii. *If* $\Gamma \vdash t : A$ *and* $\Gamma, x : A, \Gamma' \vdash t' : B$ *then* $[t/x]t' \rightsquigarrow^* [t/x]^A t'$.

Concluding Normalization

Definition. *The interpretation of types* $[\![T]\!]_\Gamma$ *is defined by:*

$$n \in [\![T]\!]_\Gamma \quad \Longleftrightarrow \quad \Gamma \vdash n : T$$

We extend this definition to non-normal terms t *in the following way:*

$$t \in [\![T]\!]_\Gamma \quad \Longleftrightarrow \quad \exists n. t \leadsto^* n \in [\![T]\!]_\Gamma$$

Concluding Normalization

Lemma (Hereditary Substitution for the Interpretation of Types)**.** *If* $n \in [\![T]\!]_\Gamma$ *and* $n' \in [\![T']\!]_{\Gamma,x:T,\Gamma'}$*, then* $[n/x]^T n' \in [\![T']\!]_{\Gamma,\Gamma'}$*.*

Proof. We know by totality and type preservation that there exists a term s such that $[n/x]^T n' = s$ and $\Gamma, \Gamma' \vdash s : T'$, and by normality preservation s is normal. Therefore, $s \in [\![T']\!]_{\Gamma,\Gamma'}$. □

Concluding Normalization

Theorem (Type Soundness). *If* $\Gamma \vdash t : T$ *then* $t \in [\![T]\!]_\Gamma$.

Corollary (Normalization). *If* $\Gamma \vdash t : T$ *then there exists a term* n *such that* $t \rightsquigarrow^* n$.

Related Work

- The key notion of using a lexicographic ordering on an ordering on types and the strict subexpression ordering on proofs dates all the way to Prawitz 1965.
  - STLC: Lévy:1967, Girard:1989, and Amadio:1998.
- Hereditary substitution was first made explicit by Watkins: 2004 and Adams:2004.

Related Work

- Abel:2006 implemented a normalizer using sized heterogeneous types.
- Abel:2008 uses hereditary substitution as a normalization function at the kind level in the metatheory of higher order subtyping.
- Keller:2010 formalized the hereditary substitution for STLC in Agda.

Related Work

- David:2003 show strong normalization of the simply typed $\lambda\Delta$-calculus using a lexicographic ordering.

Conclusion

- Hereditary substitution is a proof method which shows promise as an effective tool to prove normalization of typed λ-calculi.
- We showed how to adapt this proof method to a type theory with control.
- The key notion was to eliminate auxiliary redexes during reduction.

Thank you!