Nancy-Université – LORIA (INRIA)

Superdeduction as a Logical Framework

Réunion des groupes de travail GEOCAL et LAC du GDR IM

Guillaume Burel

Friday March 7th, 2008

Definition

[Pfenning, 1996]:
"A logical framework is a meta-language for the specification of deductive systems."

Definition

[Pfenning, 1996]:
"A logical framework is a meta-language for the specification of deductive systems."

The canonical logical framework is the one from Edinburgh [Harper et al., 1993].
Based on the λ-calculus with dependent types: λP or $\lambda\Pi$

Example

First-order natural deduction in ELF:

$$\iota : *$$
$$o : *$$
$$\dot{\neg} : o \Rightarrow o$$
$$\dot{\Rightarrow} : o \Rightarrow o \Rightarrow o$$
$$\dot{\forall} : (\iota \Rightarrow o) \Rightarrow o$$

Example

First-order natural deduction in ELF:

$$
\begin{aligned}
&\iota : * \\
&o : * \\
&\dot{\neg} : o \Rightarrow o \\
&\dot{\Rightarrow} : o \Rightarrow o \Rightarrow o \\
&\dot{\forall} : (\iota \Rightarrow o) \Rightarrow o
\end{aligned}
$$

$$|\forall x.\ \phi|_\ell = (\dot{\forall}\ \lambda x : \iota\ |\phi|_{x::\ell})$$

Example

First-order natural deduction in ELF:

$$\begin{aligned} \iota &: * \\ o &: * \\ \dot{\neg} &: o \Rightarrow o \\ \dot{\Rightarrow} &: o \Rightarrow o \Rightarrow o \\ \dot{\forall} &: (\iota \Rightarrow o) \Rightarrow o \end{aligned}$$

$$|\forall x.\ \phi|_\ell = (\dot{\forall}\ \lambda x : \iota\ |\phi|_{x::\ell})$$

$$true : o \Rightarrow *$$
$$\forall_I : \Pi f : \iota \Rightarrow o\ (\Pi x : \iota\ (true\ (f\ x))) \Rightarrow (true\ (\dot{\forall}\ \lambda x : \iota\ (f\ x)))$$

Superdeduction

Opposite approach to LF

From (the presentation of) a theory is computed a deductive system

Example:

$\forall a\ b.\ a \subseteq b \Leftrightarrow (\forall x.\ x \in a \Rightarrow x \in b)$

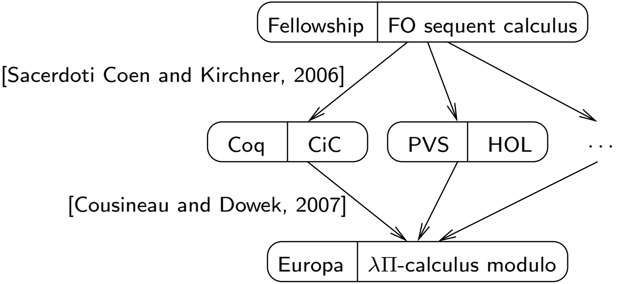
$$\subseteq_I^{\mathsf{df}} \frac{\Gamma, x \in a \vdash x \in b}{\Gamma \vdash a \subseteq b}\ x \text{ not free in } \Gamma \qquad \subseteq_E^{\mathsf{df}} \frac{\Gamma \vdash a \subseteq b \qquad \Gamma \vdash t \in a}{\Gamma \vdash t \in b}$$

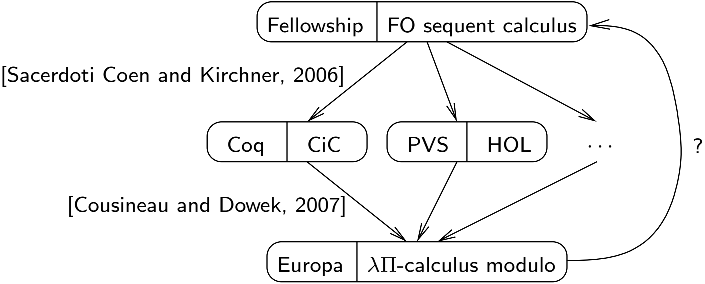
Superdeduction as a Logical Framework

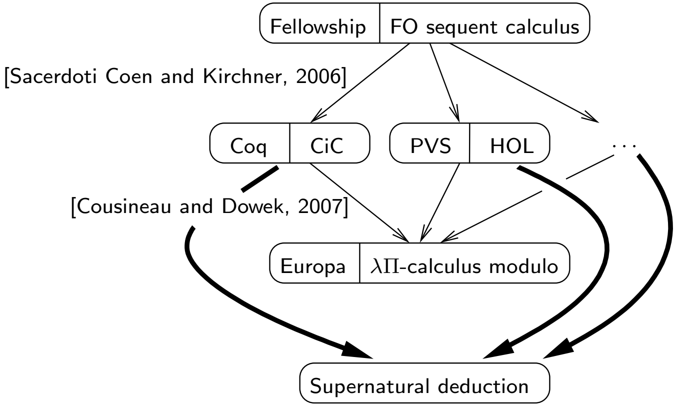
But superdeduction can be seen as a LF:
From a deductive system, find a first-order theory such that the superdeductive system corresponds to it
Already done for HOL [Dowek et al., 2001]
Here: for functional Pure Type Systems

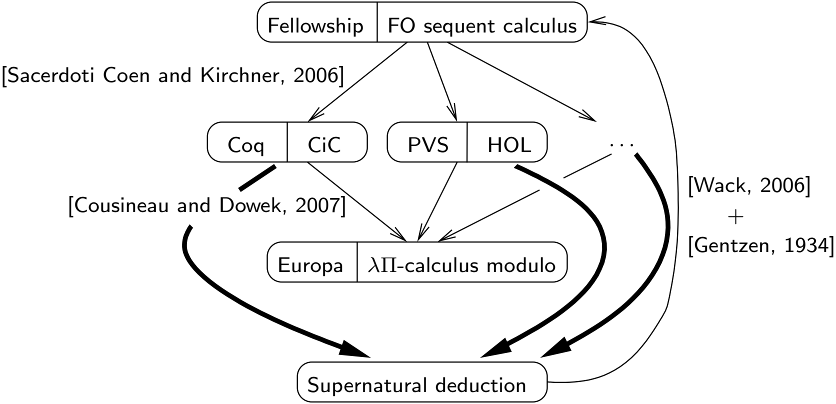
PTS in superdeduction

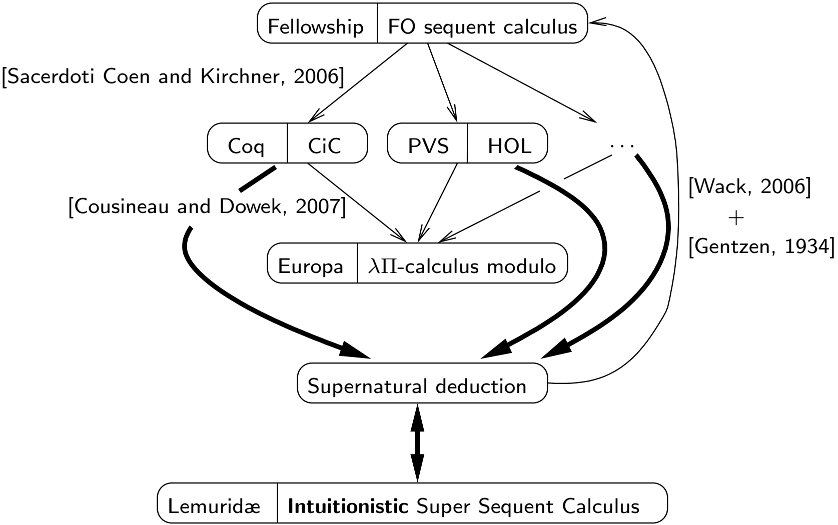
- Show that superdeduction is expressive as a logical framework
- Apply well-studied first-order methods to PTS (see also [Stehr and Meseguer, 2004])
- Cooperation between proof assistants
- New approach to normalization in PTS ?


Fellowship
FO sequent calculus
[Sacerdoti Coen and Kirchner, 2006]
Coq
CiC
PVS
HOL
. . .
[Cousineau and Dowek, 2007]
Europa
λΠ-calculus modulo


Fellowship
FO sequent calculus
[Sacerdoti Coen and Kirchner, 2006]
Coq
CiC
PVS
HOL
· · ·
?
[Cousineau and Dowek, 2007]
Europa
λΠ-calculus modulo


Fellowship
FO sequent calculus
[Sacerdoti Coen and Kirchner, 2006]
Coq
CiC
PVS
HOL
...
[Cousineau and Dowek, 2007]
Europa
λΠ-calculus modulo
Supernatural deduction


Fellowship
FO sequent calculus
[Sacerdoti Coen and Kirchner, 2006]
Coq
CiC
PVS
HOL
. . .
[Cousineau and Dowek, 2007]
[Wack, 2006]
+
[Gentzen, 1934]
Europa
λΠ-calculus modulo
Supernatural deduction


Fellowship
FO sequent calculus
[Sacerdoti Coen and Kirchner, 2006]
Coq
CiC
PVS
HOL
. . .
[Cousineau and Dowek, 2007]
[Wack, 2006]
+
[Gentzen, 1934]
Europa
λΠ-calculus modulo
Supernatural deduction
Lemuridæ
Intuitionistic Super Sequent Calculus

Outline

Input theory

A first-order theory, presented as a rewrite system whose rules

- rewrite terms to terms
- or rewrite atomic propositions to first-order formulæ

Examples:

$$x + s(y) \longrightarrow s(x + y)$$

$$Singleton(p) \longrightarrow \forall x,\ \forall y,\ x \in p \Rightarrow y \in p \Rightarrow x = y$$

Computing introduction rules

Proposition rewrite rule $A \rightarrow P$:
Apply all possible introduction rules to P
Create a new inference rule with A as conclusion and the open leaves as premises
Keep the side-conditions (freshness)

Example

$$Singleton(p) \rightarrow \forall x,\ \forall y,\ x \in p \Rightarrow y \in p \Rightarrow x = y$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\Gamma, x \in p, y \in p \vdash x = y}{\Gamma, x \in p \vdash y \in p \Rightarrow x = y}\Rightarrow_I}{\Gamma \vdash x \in p \Rightarrow y \in p \Rightarrow x = y}\Rightarrow_I}{\Gamma \vdash \forall y,\ x \in p \Rightarrow y \in p \Rightarrow x = y}\forall_I\ y \text{ not free in } \Gamma}{\Gamma \vdash \forall x,\ \forall y,\ x \in p \Rightarrow y \in p \Rightarrow x = y}\forall_I\ x \text{ not free in } \Gamma$$

Example

$$Singleton(p) \rightarrow \forall x,\ \forall y,\ x \in p \Rightarrow y \in p \Rightarrow x = y$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\Gamma, x \in p, y \in p \vdash x = y}{\Gamma, x \in p \vdash y \in p \Rightarrow x = y}\Rightarrow_I}{\Gamma \vdash x \in p \Rightarrow y \in p \Rightarrow x = y}\Rightarrow_I}{\Gamma \vdash \forall y,\ x \in p \Rightarrow y \in p \Rightarrow x = y}\forall_I\ y \text{ not free in } \Gamma}{\Gamma \vdash \forall x,\ \forall y,\ x \in p \Rightarrow y \in p \Rightarrow x = y}\forall_I\ x \text{ not free in } \Gamma$$

$$Singl_I^{\mathsf{def}}\ \dfrac{\Gamma, x \in p, y \in p \overset{+}{\vdash} x = y}{\Gamma \overset{+}{\vdash} Singleton(p)}\ \begin{array}{l} x \text{ not free in } \Gamma \\ y \text{ not free in } \Gamma \end{array}$$

Computing elimination rules

Proposition rewrite rule $A \rightarrow P$:
Apply all possible elimination rules to P, keeping new assumptions
Create a new inference rule with A as first premise, the other open leaves as premises, and the same conclusion

Example

$$Singleton(p) \rightarrow \forall x,\ \forall y,\ x \in p \Rightarrow y \in p \Rightarrow x = y$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\Gamma \vdash \forall x,\ \forall y,\ x \in p \Rightarrow y \in p \Rightarrow x = y}{\Gamma \vdash \forall y,\ t \in p \Rightarrow y \in p \Rightarrow t = y}\ \forall_E}{\Gamma \vdash t \in p \Rightarrow u \in p \Rightarrow t = u}\ \forall_E \qquad \Gamma \vdash t \in p}{\Gamma \vdash u \in p \Rightarrow t = u}\ \Rightarrow_E \qquad \Gamma \vdash u \in p}{\Gamma \vdash t = u}\ \Rightarrow_E$$

Example

$$Singleton(p) \rightarrow \forall x,\ \forall y,\ x \in p \Rightarrow y \in p \Rightarrow x = y$$

$$\Rightarrow_E \dfrac{\Rightarrow_E \dfrac{\forall_E \dfrac{\forall_E \dfrac{\Gamma \vdash \forall x,\ \forall y,\ x \in p \Rightarrow y \in p \Rightarrow x = y}{\Gamma \vdash \forall y,\ t \in p \Rightarrow y \in p \Rightarrow t = y}}{\Gamma \vdash t \in p \Rightarrow u \in p \Rightarrow t = u} \qquad \Gamma \vdash t \in p}{\Gamma \vdash u \in p \Rightarrow t = u} \qquad \Gamma \vdash u \in p}{\Gamma \vdash t = u}$$

$$Singl_E^{\mathsf{def}} \dfrac{\Gamma \overset{+}{\vdash} Singleton(p) \qquad \Gamma \overset{+}{\vdash} t \in p \qquad \Gamma \overset{+}{\vdash} u \in p}{\Gamma \overset{+}{\vdash} t = u}$$

Equivalence

All inference rules (new and logical) are applied modulo the term rewrite system

Proposition 1 ([Wack, 2006]).

For a theory $\mathcal{T}$ corresponding to the considered rewrite system,

$$\mathcal{T} \vdash P \textit{ iff } \overset{+}{\vdash} P$$

For $P(x) \to Q(x)$, theory: $\forall x,\ P(x) \Leftrightarrow Q(x)$

Outline

Pure type systems

- A set of sorts $\mathcal{S}$
- A binary relation $\mathcal{A} \subseteq \mathcal{S} \times \mathcal{S}$ (axioms)
- A trinary relation $\mathcal{R} \subseteq \mathcal{S} \times \mathcal{S} \times \mathcal{S}$ (rules)

Functional PTS: $\mathcal{A}$ and $\mathcal{R}$ defines (partial) functions from $\mathcal{S}$ and $\mathcal{S} \times \mathcal{S}$ to $\mathcal{S}$

Typing system

$$\text{Empty}\ \frac{}{[]\ \text{well-formed}}$$

$$\text{Declaration}\ \frac{\Gamma \vdash_{\mathsf{PTS}} A : s}{\Gamma[x : A]\ \text{well-formed}}\ s \in \mathcal{S} \text{ and } x \text{ not in } \Gamma$$

$$\text{Sort}\ \frac{\Gamma\ \text{well-formed}}{\Gamma \vdash_{\mathsf{PTS}} s_1 : s_2}\ \langle s_1, s_2 \rangle \in \mathcal{A}$$

$$\text{Variable}\ \frac{\Gamma\ \text{well-formed} \qquad x : A \in \Gamma}{\Gamma \vdash_{\mathsf{PTS}} x : A}$$

Typing system (cont.)

$$\text{Product}\ \frac{\Gamma \vdash_{\mathsf{PTS}} A : s_1 \qquad \Gamma[x:A] \vdash_{\mathsf{PTS}} B : s_2}{\Gamma \vdash_{\mathsf{PTS}} \Pi x : A\ B : s_3}\ \langle s_1, s_2, s_3 \rangle \in \mathcal{R}$$

$$\text{Application}\ \frac{\Gamma \vdash_{\mathsf{PTS}} T : \Pi x : A\ B \qquad \Gamma \vdash_{\mathsf{PTS}} U : A}{\Gamma \vdash_{\mathsf{PTS}} (T\ U) : \{U/x\}B}$$

$$\text{Abstraction}\ \frac{\Gamma \vdash_{\mathsf{PTS}} \Pi x : A\ B : s \qquad \Gamma[x:A] \vdash_{\mathsf{PTS}} T : B}{\Gamma \vdash_{\mathsf{PTS}} \lambda x : A\ T : \Pi x : A\ B}$$

$$\text{Conversion}\ \frac{\Gamma \vdash_{\mathsf{PTS}} T : A \qquad \Gamma \vdash_{\mathsf{PTS}} B : s}{\Gamma \vdash_{\mathsf{PTS}} T : B}\ s \in \mathcal{S} \text{ and } A \overset{*}{\longleftrightarrow}_{\beta} B$$

Outline

- Introduction
  - Logical Frameworks
  - Superdeduction
- Natural Superdeduction
- Pure Type Systems
- PTS in Supernatural Deduction
  - Encoding
  - Resulting System
  - Soundness and conservativeness
- Normalization
  - Proof terms
  - Equivalence of reductions ?
- Conclusion

First-order syntax

Constant s for each sort $s \in \mathcal{S}$

Function symbol $\dot{\pi}_{\langle s_1, s_2, s_3 \rangle}$ for each rule $\langle s_1, s_2, s_3 \rangle \in \mathcal{R}$

λ-calculus with explicit substitutions λ_W following a scheme [Kesner, 2000] ensuring its confluence

Translations of terms

- $|x|_{\ell_1::x::\ell_2} \stackrel{\text{def}}{=} \underline{\overline{\ell_1} + 1}$ if $x \notin \ell_1$
- $|x|_{\ell} \stackrel{\text{def}}{=} x\,[shift]^{\overline{\ell}}$ if $x \notin \ell$
- $|s|_{\ell} \stackrel{\text{def}}{=} \mathsf{s}$
- $|\lambda x : A\ t|_{\ell} \stackrel{\text{def}}{=} \lambda |t|_{x::\ell}$
- $|\Pi x : A\ B|^{\Gamma}_{\ell} \stackrel{\text{def}}{=} \dot{\pi}_{\langle s_1, s_2, s_3 \rangle}\left(|A|^{\Gamma}_{\ell}, |B|^{\Gamma[x:A]}_{x::\ell}\right)$ where
  - s_1 corresponds the type of A in Γ
  - s_2 the type of B in $\Gamma[x : A]$
  - $\langle s_1, s_2, s_3 \rangle \in \mathcal{R}$

First-order theory

Rules on terms: λ_W +

$$\mathsf{s}\,[t] \longrightarrow \mathsf{s}$$
$$\dot\pi_{\langle s_1,s_2,s_3\rangle}(a,b)\,[s] \longrightarrow \dot\pi_{\langle s_1,s_2,s_3\rangle}(a\,[s]\,,b\,[\mathsf{lift}(s)])$$

Rules on propositions:

$$\epsilon\,(\mathsf{s}_1,\mathsf{s}_2) \longrightarrow \top \qquad (\langle s_1,s_2\rangle \in \mathcal{A}) \qquad (1)$$
$$\epsilon\left(\dot\pi_{\langle s_1,s_2,s_3\rangle}(a,b)\,,\mathsf{s}_3\right) \longrightarrow \epsilon\,(a,\mathsf{s}_1) \wedge \forall z.\ \epsilon\,(z,a) \Rightarrow \epsilon\,(b\,[\mathsf{cons}(z)]\,,\mathsf{s}_2) \qquad (2)$$
$$\epsilon\left(t,\dot\pi_{\langle s_1,s_2,s_3\rangle}(a,b)\right) \longrightarrow \epsilon\left(\dot\pi_{\langle s_1,s_2,s_3\rangle}(a,b)\,,\mathsf{s}_3\right) \wedge \forall z.\ \epsilon\,(z,a) \Rightarrow \epsilon\,(t\ z,b\,[\mathsf{cons}(z)]) \qquad (3)$$

$$\epsilon\left(\mathsf{s}_1,\mathsf{s}_2\right) \quad \rightarrow \quad \top \qquad (1)$$

$$(1)_I \; \frac{}{\Gamma \vdash^{+} \epsilon\left(\mathsf{s}_1,\mathsf{s}_2\right)}$$

$$\text{Sort}\; \frac{\Gamma \text{ well-formed}}{\Gamma \vdash_{\mathsf{PTS}} s_1 : s_2}$$

$$\epsilon\left(\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b),\mathsf{s}_3\right) \quad\rightarrow\quad \begin{array}{l}\epsilon(a,\mathsf{s}_1)\wedge\\ \forall z.\ \epsilon(z,a)\Rightarrow\epsilon(b[\mathsf{cons}(z)],\mathsf{s}_2)\end{array} \qquad (2)$$

$$(2)_I\ \frac{\Gamma\overset{+}{\vdash}\epsilon(a,\mathsf{s}_1) \qquad \Gamma,\epsilon(z,a)\overset{+}{\vdash}\epsilon(b[\mathsf{cons}(z)],\mathsf{s}_2)}{\Gamma\overset{+}{\vdash}\epsilon\left(\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b),\mathsf{s}_3\right)}\ z\ \text{not free in}\ \Gamma$$

$$\text{Product}\ \frac{\Gamma\vdash_{\mathsf{PTS}}A:s_1 \qquad \Gamma[x:A]\vdash_{\mathsf{PTS}}B:s_2}{\Gamma\vdash_{\mathsf{PTS}}\Pi x:A\ B:s_3}\ \langle s_1,s_2,s_3\rangle\in\mathcal{R}$$

Loria Nancy-Université

$$\epsilon\left(\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b),\mathsf{s}_3\right) \quad\longrightarrow\quad \epsilon(a,\mathsf{s}_1)\wedge \qquad (2)$$
$$\forall z.\ \epsilon(z,a)\Rightarrow\epsilon\left(b\left[\mathsf{cons}(z)\right],\mathsf{s}_2\right)$$

$$(2)_{E1}\ \frac{\Gamma \vdash^{+} \epsilon\left(\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b),\mathsf{s}_3\right)}{\Gamma \vdash^{+} \epsilon(a,\mathsf{s}_1)}$$

No corresponding inference rule in the PTS

Conservative because if $\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b)$ is the translation of some PTS term $\Pi x : A\ B$, then A has type s_1

$$\epsilon\left(\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b),\mathsf{s}_3\right) \quad\longrightarrow\quad \epsilon(a,\mathsf{s}_1)\wedge \forall z.\ \epsilon(z,a)\Rightarrow\epsilon\left(b\left[\mathsf{cons}(z)\right],\mathsf{s}_2\right) \qquad (2)$$

$$(2)_{E2}\ \frac{\Gamma \overset{+}{\vdash} \epsilon\left(\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b),\mathsf{s}_3\right) \qquad \Gamma \overset{+}{\vdash} \epsilon(u,a)}{\Gamma \overset{+}{\vdash} \epsilon\left(b\left[\mathsf{cons}(u)\right],\mathsf{s}_2\right)}$$

No corresponding inference rule in the PTS

Conservative because $\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b)$ is the translation of some PTS term $\Pi x : A\ B$, then B has type s_2 in $\Gamma[x : A]$ and by substitution $\Gamma \vdash_{\mathsf{PTS}} \{U/x\}B : s_2$

$$\epsilon\left(t, \dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b)\right) \quad\rightarrow\quad \epsilon\left(\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b), \mathsf{s_3}\right) \wedge \forall z.\ \epsilon(z,a) \Rightarrow \epsilon\left(t\ z, b\left[\mathsf{cons}(z)\right]\right) \qquad (3)$$

$$(3)_I \frac{\Gamma \overset{+}{\vdash} \epsilon\left(\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b), \mathsf{s_3}\right) \qquad \Gamma, \epsilon(z,a) \overset{+}{\vdash} \epsilon\left(t\ z, b\left[\mathsf{cons}(z)\right]\right)}{\Gamma \overset{+}{\vdash} \epsilon\left(t, \dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b)\right)}$$

$$\text{Abstraction}\ \frac{\Gamma \vdash_{\mathsf{PTS}} \Pi x : A\ B : s_3 \qquad \Gamma[x:A] \vdash_{\mathsf{PTS}} T : B}{\Gamma \vdash_{\mathsf{PTS}} \lambda x : A\ T : \Pi x : A\ B}$$

Need for subject reduction for η

$$\epsilon\left(t, \dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b)\right) \quad \longrightarrow \quad \epsilon\left(\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b), \mathsf{s_3}\right) \wedge \forall z.\ \epsilon(z,a) \Rightarrow \epsilon\left(t\ z, b\left[\mathsf{cons}(z)\right]\right) \qquad (3)$$

$$(3)_{E1}\ \frac{\Gamma \overset{+}{\vdash} \epsilon\left(t, \dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b)\right)}{\Gamma \overset{+}{\vdash} \epsilon\left(\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b), \mathsf{s_3}\right)}$$

No corresponding inference rule in the PTS

Conservative because if $\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b)$ is the translation of some PTS term $\Pi x : A\ B$, then A has type s_1 in Γ, B has type s_2 in $\Gamma[x : A]$, hence $\Pi x : A\ B$ has type s_3

$$\epsilon\left(t, \dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b)\right) \quad\longrightarrow\quad \epsilon\left(\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b), \mathsf{s_3}\right) \wedge \forall z.\ \epsilon(z,a) \Rightarrow \epsilon\left(t\ z, b\left[\mathsf{cons}(z)\right]\right) \qquad (3)$$

$$(3)_{E2}\ \frac{\Gamma \vdash^{+} \epsilon\left(t, \dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b)\right) \qquad \Gamma \vdash^{+} \epsilon(u,a)}{\Gamma \vdash^{+} \epsilon\left(t\ u, b\left[\mathsf{cons}(u)\right]\right)}$$

$$\text{Application}\ \frac{\Gamma \vdash_{\mathsf{PTS}} T : \Pi x : A\ B \qquad \Gamma \vdash_{\mathsf{PTS}} U : A}{\Gamma \vdash_{\mathsf{PTS}} (T\ U) : \{U/x\}B}$$

Soundness

Theorem 2.
For all contexts Γ*, for all PTS terms* A *and* B*, if* $\Gamma \vdash_{\mathsf{PTS}} A : B$ *then* $|\Gamma| \vdash^{+} \epsilon\,(|A|, |B|)$.

Remaining inference rules:

- Variable: through Axiom
- Conversion: modulo the term rewrite system

Conservativeness

Theorem 3.
*For all well-formed contexts Γ, for all terms a, b,
if $|\Gamma| \vdash^{+} \epsilon\,(a, b)$ then there exists A and B such that*

- $a \xrightarrow{*} |A|^{\Gamma}$
- $b \xrightarrow{*} |B|^{\Gamma}$
- $\Gamma \vdash_{\mathsf{PTS}} A : B$

Conservativeness

Theorem 3.
For all well-formed contexts Γ, for all terms a, b,
if $|\Gamma| \vdash^{+} \epsilon\,(a, b)$ *then there exists A and B such that*

- $a \xrightarrow{*} |A|^{\Gamma}$
- $b \xrightarrow{*} |B|^{\Gamma}$
- $\Gamma \vdash_{\mathsf{PTS}} A : B$

Outline

Proof terms

Proof terms for superdeduction are particular ρ-terms
In this case, we need function symbols R_1, R_2 and R_3

$$(1)_I \frac{}{\Gamma \stackrel{+}{\vdash} R_1 : \epsilon(\mathsf{s}_1, \mathsf{s}_2)}$$

Proof terms for (2)

$$(2)_I \; \frac{\Gamma \vdash^{+} M : \epsilon\,(a, \mathsf{s}_1) \qquad \Gamma, \alpha : \epsilon\,(z, a) \vdash^{+} N : \epsilon\,(b\,[\mathsf{cons}(z)]\,, \mathsf{s}_2)}{\Gamma \vdash^{+} \lambda R_2(z, \alpha, y).\ y\langle M, N\rangle : \epsilon\left(\dot{\pi}_{\langle s_1, s_2, s_3\rangle}\,(a, b)\,, \mathsf{s}_3\right)}$$

$$(2)_{E1} \; \frac{\Gamma \vdash^{+} M : \epsilon\left(\dot{\pi}_{\langle s_1, s_2, s_3\rangle}\,(a, b)\,, \mathsf{s}_3\right)}{\Gamma \vdash^{+} M\ R_2(z, \alpha, \lambda\langle x, y\rangle.\ x) : \epsilon\,(a, \mathsf{s}_1)}$$

$$(2)_{E2} \; \frac{\Gamma \vdash^{+} M : \epsilon\left(\dot{\pi}_{\langle s_1, s_2, s_3\rangle}\,(a, b)\,, \mathsf{s}_3\right) \qquad \Gamma \vdash^{+} N : \epsilon\,(u, a)}{\Gamma \vdash^{+} M\ R_2(u, N, \lambda\langle x, y\rangle.\ y) : \epsilon\,(b\,[\mathsf{cons}(u)]\,, \mathsf{s}_2)}$$

Proof terms for (3)

$$(3)_I \; \frac{\Gamma \vdash^{+} M : \epsilon\left(\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b), \mathsf{s}_3\right) \qquad \Gamma, \alpha : \epsilon(z,a) \vdash^{+} N : \epsilon(t\ z, b[\mathsf{cons}(z)])}{\Gamma \vdash^{+} \lambda R_3(z,\alpha,y).\ y\langle M, N\rangle : \epsilon\left(t, \dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b)\right)}$$

$$(3)_{E1} \; \frac{\Gamma \vdash^{+} M : \epsilon\left(t, \dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b)\right)}{\Gamma \vdash^{+} M\ R_3(z,\alpha,\lambda\langle x,y\rangle.\ x) : \epsilon\left(\dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b), \mathsf{s}_3\right)}$$

$$(3)_{E2} \; \frac{\Gamma \vdash^{+} M : \epsilon\left(t, \dot{\pi}_{\langle s_1,s_2,s_3\rangle}(a,b)\right) \qquad \Gamma \vdash^{+} N : \epsilon(u,a)}{\Gamma \vdash^{+} M\ R_3(u,N,\lambda\langle x,y\rangle.\ y) : \epsilon(t\ u, b[\mathsf{cons}(u)])}$$

Reductions

$$(\lambda x.\ T\ U) \xrightarrow[\beta]{} \{U/x\}T$$

$$(\lambda R_3(z,\alpha,y).\ y\langle T_1,T_2\rangle)\ R_3(u,N,\lambda\langle x,y\rangle.\ y) \xrightarrow[\rho]{2} \{u/z\}\{N/\alpha\}T_2$$

What about other ρ-reductions ?

Conjecture

Conjecture 1.

A PTS is strongly normalizing iff the superdeductive system associated with it is.

Conjecture

Conjecture 1.

A PTS is strongly normalizing iff the superdeductive system associated with it is.

Proposition 4.

A PTS is strongly normalizing if the superdeductive system associated with it is.

Allow to use normalization methods for deduction modulo (premodel, superconsistency)

Outline

- Introduction
  - Logical Frameworks
  - Superdeduction
- Natural Superdeduction
- Pure Type Systems
- PTS in Supernatural Deduction
  - Encoding
  - Resulting System
  - Soundness and conservativeness
- Normalization
  - Proof terms
  - Equivalence of reductions ?
- Conclusion

Conclusion

Quite natural encoding of functional PTS into supernatural deduction

Valid typing judgments = (reducts of) provable sequents

Conclusion

Quite natural encoding of functional PTS into supernatural deduction

Valid typing judgments = (reducts of) provable sequents

Type checking = proof search

Perspectives

- Add new rules to check wellformedness of contexts (or pass the context into the term and type)

Perspectives

- Add new rules to check wellformedness of contexts (or pass the context into the term and type)
- Normalization:

Perspectives

- Add new rules to check wellformedness of contexts (or pass the context into the term and type)
- Normalization:
    - Prove conjecture

Perspectives

- Add new rules to check wellformedness of contexts (or pass the context into the term and type)
- Normalization:
  - Prove conjecture
  - Look at superconsistency

Perspectives

- Add new rules to check wellformedness of contexts (or pass the context into the term and type)
- Normalization:
  - Prove conjecture
  - Look at superconsistency
- Add inductive types: as in CAC ?

Perspectives

- Add new rules to check wellformedness of contexts (or pass the context into the term and type)
- Normalization:
  - Prove conjecture
  - Look at superconsistency
- Add inductive types: as in CAC ?
- Add subtyping as in PVS

Perspectives

- Add new rules to check wellformedness of contexts (or pass the context into the term and type)
- Normalization:
  - Prove conjecture
  - Look at superconsistency
- Add inductive types: as in CAC ?
- Add subtyping as in PVS
- Modify Lemuridæ to deal with intuitionistic logic

Perspectives

- Add new rules to check wellformedness of contexts (or pass the context into the term and type)
- Normalization:
  - Prove conjecture
  - Look at superconsistency
- Add inductive types: as in CAC ?
- Add subtyping as in PVS
- Modify Lemuridæ to deal with intuitionistic logic
- Encode more deductive systems using superdeduction: superdeduction as a logical framework

Cousineau, D. and Dowek, G. (2007).
Embedding pure type systems in the lambda-pi-calculus modulo.
In Ronchi Della Rocca, S., editor, *TLCA*, volume 4583 of *Lecture Notes in Computer Science*, pages 102–117. Springer.

Dowek, G., Hardin, T., and Kirchner, C. (2001).
HOL-$\lambda\sigma$ an intentional first-order expression of higher-order logic.
Mathematical Structures in Computer Science, 11(1):1–25.

Gentzen, G. (1934).
Untersuchungen über das logische Schliessen.
Mathematische Zeitschrift, 39:176–210, 405–431.

Translated in Szabo, editor., *The Collected Papers of Gerhard Gentzen* as "Investigations into Logical Deduction".

Harper, R., Honsell, F., and Plotkin, G. (1993).
A framework for defining logics.
Journal of the ACM, 40(1):143–184.

Kesner, D. (2000).
Confluence of extensional and non-extensional λ-calculi with explicit substitutions.
Theoretical Computer Science, 238(1–2):183–220.

Pfenning, F. (1996).
The practice of logical frameworks.
In *CAAP*, volume 1059 of *Lecture Notes in Computer Science*, pages 119–134. Springer.

Sacerdoti Coen, C. and Kirchner, F. (2006).
Fellowship.
http://www.lix.polytechnique.fr/Labo/Florent.Kirchner/fellowship/.

Stehr, M.-O. and Meseguer, J. (2004).
Pure type systems in rewriting logic: Specifying typed higher-order languages in a first-order logical framework.
In Owe, O., Krogdahl, S., and Lyche, T., editors, *Essays in Memory of Ole-Johan Dahl*, volume 2635 of *Lecture Notes in Computer Science*, pages 334–375. Springer.

Wack, B. (2006).
Supernatural deduction.
Manuscript, available at http://www.loria.fr/~wack/papers/supernatural.ps.gz.