A Simpler Proof Theory for Nominal Logic

James Cheney

University of Edinburgh

FOSSACS 2005

April 6, 2005

Motivation

- Nominal logic [Pitts 2003]: an extension of sorted first-order logic that formalizes
  - *names*, *name-binding*, and *quantification over fresh names*.
  - via primitive concepts of *swapping* and *freshness* [Gabbay-Pitts 1999]
- Problem: Existing proof systems/axiomatizations are "overly complex" (a subjective judgment)
- One difficulty: complex axiom schemes/rules for И-quantifier

Motivation

- Original approach [Pitts 2003]: an axiom scheme

  $$\text{И}a.\phi \iff \exists a.a \# \vec{x} \wedge \phi \qquad (FV(\text{И}a.\phi) \subseteq \{\vec{x}\})$$

  defining И in terms of $\exists$, $\wedge$, and freshness $\#$.

- Gives little insight into self-duality and symmetry properties of И

- Syntactic side-condition makes checking uses painful

- Gentzen-style rule systems often preferable to axiomatic definitions

Motivation

- [Gabbay,Pitts 1999], [Pitts 2003] proposed sequent rules

$$\frac{\Gamma, a \# \vec{x}, \phi \Rightarrow \psi}{\Gamma, \text{И}a.\phi \Rightarrow \psi}\,\text{И}L \qquad \frac{\Gamma, a \# \vec{x} \Rightarrow \phi}{\Gamma \Rightarrow \text{И}a.\phi}\,\text{И}R$$

  where $a \notin FV(\Gamma, \psi)$ and $FV(\Gamma, \psi, \text{И}a.\phi) \subseteq \{\vec{x}\}$.

- Not much simpler than axiom scheme

- Not closed under substitution, so cut-elimination hard to prove

Motivation

- Most recent idea [Gabbay, Cheney 2004]:

$$\frac{\Gamma, a \# \vec{t}, \phi \Rightarrow \psi}{\Gamma, \text{И}a.\phi \Rightarrow \psi}\,\text{И}L \qquad \frac{\Gamma, a \# \vec{t} \Rightarrow \phi}{\Gamma \Rightarrow \text{И}a.\phi}\,\text{И}R$$

  where $a \notin FV(\Gamma, \psi)$ and ϕ can be decomposed as $\phi'(a, \vec{t})$ where $a \notin FV(\vec{t})$ and $\phi'(\cdots)$ mentions only quantifiers/connectives.

- Closed under substitution, so cut-elimination straightforward

- but seems nondeterministic & side-conditions even more painful

Motivation

- Miller and Tiu's $FO\lambda^{\nabla}$ logic includes *local name contexts* and a self-dual quantifier ∇:

$$\frac{\Sigma : \Gamma, (\sigma, x) \triangleright \phi \Rightarrow \mathcal{A}}{\Sigma : \Gamma, \sigma \triangleright \nabla x.\phi \Rightarrow \mathcal{A}}\,\nabla L \qquad \frac{\Sigma : \Gamma \Rightarrow (\sigma, x) \triangleright \phi}{\Sigma : \Gamma \Rightarrow \sigma \triangleright \nabla x.\phi}\,\nabla R$$

  where $x \notin \Sigma$.

- These rules are not much more complicated that $\forall R, \exists L$.

- Can we obtain similarly simple rules for И?

Motivation

- In αProlog [Cheney, Urban 2004] clauses can mention explicit name symbols $\mathsf{a}, \mathsf{b}, \ldots$:

  $$p(\vec{\mathsf{a}}, \vec{X}) :- G(\vec{\mathsf{a}}, \vec{X})$$

  Clauses are interpreted as *implicitly* И∀*-quantified*:

  $$\text{И}\vec{\mathsf{a}}.\forall \vec{X}.G(\vec{\mathsf{a}}, \vec{X}) \supset p(\vec{\mathsf{a}}, \vec{X})$$

  The И-quantifier is interpreted in proof search as "generate a fresh name a, then proceed"

- Can we justify this interpretation using similar proof rules for И?

Motivation

- My approach: use special name symbols a and "freshness contexts" Σ that store needed freshness information

$$\frac{\Sigma\#\mathsf{a} : \Gamma, \phi \Rightarrow \psi}{\Sigma : \Gamma, \text{И}\mathsf{a}.\phi \Rightarrow \psi}\,\text{И}L \qquad \frac{\Sigma\#\mathsf{a} : \Gamma \Rightarrow \phi}{\Sigma : \Gamma \Rightarrow \text{И}\mathsf{a}.\phi}\,\text{И}R$$

  where $\mathsf{a} \notin \Sigma$.

- Closed under substitution, side conditions simpler (like $\forall R, \exists L, \nabla L/R$)

- Management of freshness information "compartmentalized" into Σ-context and an additional rule.

Outline

- Quick overview of nominal logic
- The sequent calculus $NL^{\Rightarrow}$
- Relating $FO\lambda^{\nabla}$ and nominal logic
- Conclusion

Nominal Logic: Syntax

- Names a, b inhabiting name-sorts A, A'
- Swapping $(\mathsf{a}\ \mathsf{b}) \cdot x$ exchanges two names
- Abstraction $\langle \mathsf{a} \rangle x$ constructs "objects with one bound name"
- Freshness relation $\mathsf{a} \# x$ means "x does not depend on a"
- И-quantifier quantifies over fresh names: $\text{И}\mathsf{a}.\phi$ means "for fresh names a, ϕ holds"

Names: What are they?

- In this approach, names are a new syntactic class, distinct from variables and from function or constant symbols

- Syntactically different name symbols always denote semantically distinct names

- Names can be "semantically bound" in abstractions $\langle \mathsf{a} \rangle x$, but also "syntactially bound" by $\reflectbox{N}$: $\reflectbox{N}\mathsf{a}.\phi$

- $\langle \mathsf{a} \rangle f(\mathsf{a}, x)$ and $\langle \mathsf{b} \rangle (\mathsf{b}, x)$ are different nominal terms (and can denote different values), while $\reflectbox{N}\mathsf{a}.p(\mathsf{a}, x)$ and $\reflectbox{N}\mathsf{b}.p(\mathsf{b}, x)$ are α-equivalent formulas

Theory of Swapping and Freshness

- Swapping

$$(a\ b) \cdot a \approx b \quad (a\ a) \cdot x \approx x \quad (a\ b) \cdot (a\ b) \cdot x \approx x$$

$$(a\ b) \cdot c \approx c \quad (a\ b) \cdot f(\vec{x}) = f((a\ b) \cdot \vec{x})$$

- Freshness

$$a \# a' \iff a \not\approx a' \quad a \# x \land b \# x \supset (a\ b) \cdot x \approx x$$

- Examples

$$\mathsf{a} \# \mathsf{b} \approx (\mathsf{a}\ \mathsf{b}) \cdot \mathsf{a} \quad (\mathsf{a}\ \mathsf{b}) \cdot f(\mathsf{a}, \langle \mathsf{b} \rangle \mathsf{a}, g(\mathsf{a})) \approx f(\mathsf{b}, \langle \mathsf{a} \rangle \mathsf{b}, g(\mathsf{b}))$$

Theory of Name-Abstraction

- Intuitively, $\langle \mathsf{a} \rangle x$ is "the value x with a distinguished bound name a".

- Considered equal up to "safe" renaming (α-equivalence)

$$\langle a \rangle x \approx \langle b \rangle x \quad \Longleftrightarrow \quad (a \approx b \wedge x \approx y) \vee (a \# y \wedge x \approx (a\ b) \cdot y)$$

- For example,

$$\vDash \langle \mathsf{a} \rangle \mathsf{a} \approx \langle \mathsf{b} \rangle \mathsf{b} \qquad \nvDash \langle \mathsf{a} \rangle f(\mathsf{a}, \mathsf{b}) \approx \langle \mathsf{b} \rangle f(\mathsf{b}, \mathsf{a})$$

Sequent Calculus

- Judgments use context Σ expressing both typing and freshness information

$$\Sigma ::= \cdot \mid \Sigma, x{:}S \mid \Sigma\#\mathsf{a}{:}\mathsf{A}$$

- Associate contexts with freshness constraint sets $|\Sigma|$:

$$|\cdot| = \varnothing \quad |\Sigma, x{:}S| = |\Sigma| \quad |\Sigma\#\mathsf{a}{:}\mathsf{A}| = |\Sigma| \cup \{\mathsf{a} \# t \mid \Sigma \vdash t : S\}$$

- Auxiliary rule for extracting freshness information:

$$\frac{\mathsf{a} \# t \in |\Sigma| \quad \Sigma : \Gamma, \mathsf{a} \# t \Rightarrow \psi}{\Sigma : \Gamma \Rightarrow \psi}\ \Sigma\#$$

Freshness Principle

- Fresh names can always be chosen.

$$\frac{\Sigma\#\mathsf{a} : \Gamma \Rightarrow \psi}{\Sigma : \Gamma \Rightarrow \psi}\, F \qquad (\mathsf{a} \notin \Sigma)$$

- An example derivation using (F) and $(\Sigma\#)$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\mathsf{a} \;\#\; x \in |\Sigma, x\#\mathsf{a}| \qquad \overline{\Sigma, x\#\mathsf{a} : \mathsf{a} \;\#\; x \Rightarrow \mathsf{a} \;\#\; x}}{\Sigma, x\#\mathsf{a} : \cdot \Rightarrow \mathsf{a} \;\#\; x}\,\Sigma\#}{\Sigma, x\#\mathsf{a} : \cdot \Rightarrow \exists a.a \;\#\; x}\,\exists R}{\Sigma, x : \cdot \Rightarrow \exists a.a \;\#\; x}\,F}{\Sigma : \cdot \Rightarrow \forall x.\exists a.a \;\#\; x}\,\forall R$$

Equivariance Principle

- Constants fixed by name-swapping

$$(a\ b) \cdot c \approx c$$

- Functions commute with name-swapping

$$(a\ b) \cdot f(\vec{t}) \approx f((a\ b) \cdot \vec{t})$$

- Truth preserved by name-swapping

$$\frac{\Sigma : \Gamma, p((a\ b) \cdot \vec{t}) \Rightarrow \psi}{\Sigma : \Gamma, p(\vec{t}) \Rightarrow C}\, EV$$

$\reflectbox{N}$-Quantifier Rules

- Our rules:

$$\frac{\Sigma\#\mathsf{a} : \Gamma, \phi \Rightarrow \psi}{\Sigma : \Gamma, \reflectbox{N}\mathsf{a}.\phi \Rightarrow \psi}\,\reflectbox{N}L \qquad \frac{\Sigma\#\mathsf{a} : \Gamma \Rightarrow \phi}{\Sigma : \Gamma \Rightarrow \reflectbox{N}\mathsf{a}.\phi}\,\reflectbox{N}R \qquad (\mathsf{a} \notin \Sigma)$$

- Intuitively, to either prove or use a $\reflectbox{N}$-quantified formula, instantiate it to a completely fresh name and proceed.

- Previous systems have used complex syntactic side-conditions to do this.

Denotational Semantics?

- That's another talk. Sorry!

- An incomplete semantics can be inherited from Pitts' nominal logic semantics

- A complete semantics is known [Cheney 2004], working on publication

Examples

- A simple theorem: $\text{И}\mathsf{a}.\text{И}\mathsf{b}.\mathsf{a} \# \mathsf{b}$

$$\dfrac{\dfrac{\dfrac{}{\Sigma\#\mathsf{a}\#\mathsf{b} : \mathsf{a} \# \mathsf{b} \Rightarrow \mathsf{a} \# \mathsf{b}}}{\Sigma\#\mathsf{a}\#\mathsf{b} : \cdot \Rightarrow \mathsf{a} \# \mathsf{b}}\,\Sigma\#}{\Sigma : \cdot \Rightarrow \text{И}\mathsf{a},\mathsf{b}.\mathsf{a} \# \mathsf{b}}\,\text{И}R^2$$

- Another theorem: $\text{И}\mathsf{a},\mathsf{b}.p(\mathsf{a}) \supset p(\mathsf{b})$

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\Sigma\#\mathsf{a}\#\mathsf{b} : p(\mathsf{b}) \Rightarrow p(\mathsf{b})}}{\Sigma\#\mathsf{a}\#\mathsf{b} : (\mathsf{a}\ \mathsf{b}) \cdot p(\mathsf{a}) \Rightarrow p(\mathsf{b})}\,axioms}{\Sigma\#\mathsf{a}\#\mathsf{b} : p(\mathsf{a}) \Rightarrow p(\mathsf{b})}\,EV}{\Sigma : \cdot \Rightarrow \text{И}\mathsf{a},\mathsf{b}.p(\mathsf{a}) \supset p(\mathsf{b})}\,\text{И}R^2, \supset R$$

Examples

- A non-theorem: $\text{И}\mathsf{a}.p(\mathsf{a},\mathsf{a}) \Rightarrow \text{И}\mathsf{a},\mathsf{b}.p(\mathsf{a},\mathsf{b})$

$$\frac{\Sigma\#\mathsf{a}\#\mathsf{b}\#\mathsf{a}' : p(\mathsf{a}',\mathsf{a}') \Rightarrow p(\mathsf{a},\mathsf{b})}{\Sigma : \text{И}\mathsf{a}.p(\mathsf{a},\mathsf{a}) \Rightarrow \text{И}\mathsf{a},\mathsf{b}.p(\mathsf{a},\mathsf{b})}\ \text{И}R^2, \text{И}L$$

- Another non-theorem: $\text{И}\mathsf{a}.p(\mathsf{a},y) \Rightarrow \forall x.p(x,y)$.

$$\frac{\Sigma, x\#\mathsf{a} : p(\mathsf{a},y) \Rightarrow p(x,y)}{\Sigma : \text{И}\mathsf{a}.p(\mathsf{a},y) \Rightarrow \forall x.p(x,y)}\ \text{И}L, \forall R$$

Failure?

- Observe that *failure can be difficult to detect because of equivariance...*

$$\dfrac{\dfrac{\vdots}{\Sigma : (a\ b) \cdot (a\ b) \cdot P \Rightarrow Q}}{\dfrac{\Sigma : (a\ b) \cdot P \Rightarrow Q}{\Sigma : P \Rightarrow Q}}$$

- This problem was already present in other formalizations.

- Future work: deciding $\bigwedge P \supset \bigvee Q$, where P, Q are freshness, equality, or atomic formulas.

Formal properties

- Weakening, invertibility, contraction properties

  Lemma 1 (Weakening). *If* $\Sigma : \Gamma \Rightarrow \phi$ *then* $\Sigma : \Gamma, \psi \Rightarrow \phi$*.*

  Lemma 2 (Invertibility). *The* $\text{И}L$ *and* $\text{И}R$ *rules are invertible:*

  – *If* $\Sigma : \Gamma, \text{Иa}.\psi \Rightarrow \phi$ *then* $\Sigma\#\mathsf{a} : \Gamma, \psi \Rightarrow \phi$ *(for* $\mathsf{a} \notin \Sigma$*)*

  – *If* $\Sigma : \Gamma, \psi \Rightarrow \text{Иa}.\phi$ *then* $\Sigma\#\mathsf{a} : \Gamma, \psi \Rightarrow \phi$ *(for* $\mathsf{a} \notin \Sigma$*)*

  Lemma 3 (Contraction). *If* $\Sigma : \Gamma, \psi, \psi \Rightarrow \phi$ *then* $\Sigma : \Gamma, \psi \Rightarrow \phi$*.*

Formal properties

- Equivariance was only assumed for atomic formulas, but more general rules are admissible.

  Lemma 4 (Admissibility of EVL). *If* $\Sigma : \Gamma, (a\ b) \cdot \psi \Rightarrow \phi$ *then* $\Sigma : \Gamma, \psi \Rightarrow \phi$*.*

  Lemma 5 (Admissibility of EVR). *If* $\Sigma : \Gamma, \psi \Rightarrow (a\ b) \cdot \phi$ *then* $\Sigma : \Gamma, \psi \Rightarrow \phi$*.*

  Subtle point in proof: left and right equivariance are mutually recursive (because of implication)

$$\frac{\Sigma : \Gamma, (a\ b) \cdot \phi_1 \Rightarrow (a\ b) \cdot \phi_2}{\Sigma : \Gamma \Rightarrow (a\ b) \cdot (\phi_1 \supset \phi_2)} \supset R$$

Formal properties

- hyp rule only assumed for atomic formulas, but generalized form admissible.

  Lemma 6 (Admissibility of hyp^*). *The rule*

  $$\frac{}{\Sigma : \Gamma, \phi \Rightarrow \phi}\, hyp^*$$

  is admissible.

  Proof relies on EVL for И-case:

  $$\dfrac{\dfrac{\dfrac{}{\Sigma\#\mathsf{a}\#\mathsf{b} : \phi(\mathsf{b}) \Rightarrow \phi(\mathsf{b})}\, hyp^*}{\Sigma\#\mathsf{a}\#\mathsf{b} : \Gamma, (\mathsf{a}\ \mathsf{b}) \cdot \phi(\mathsf{a}) \Rightarrow \phi(\mathsf{b})}\, axioms}{\dfrac{\Sigma\#\mathsf{a}\#\mathsf{b} : \Gamma, \phi(\mathsf{a}) \Rightarrow \phi(\mathsf{b})}{\Sigma : \Gamma, \text{И}\mathsf{a}.\phi \Rightarrow \text{И}\mathsf{a}.\phi}\, \text{И}L, \text{И}R}\, EVL$$

Formal properties

- Cut-elimination

  Theorem 7. *If* $\Sigma : \Gamma, \phi \Rightarrow \psi$ *and* $\Sigma : \Gamma' \Rightarrow \phi$ *then* $\Gamma, \Gamma' \Rightarrow \psi$

  Proof follows standard techniques of permuting cuts upward.

- *The proof is straightforward, but relies on the previous properties*

Cut-elimination: interesting case

- Given a principal И-cut,

$$\dfrac{\dfrac{\Sigma\#\mathsf{a} : \Gamma \Rightarrow \phi}{\Sigma : \Gamma \Rightarrow \text{И}\mathsf{a}.\phi}\,\text{И}R \qquad \dfrac{\Sigma\#\mathsf{a} : \Gamma, \phi \Rightarrow \psi}{\Sigma : \Gamma, \text{И}\mathsf{a}.\phi \Rightarrow \psi}\,\text{И}L}{\Sigma : \Gamma \Rightarrow \psi}\,cut$$

  permute the cut upward using the freshness principle:

$$\dfrac{\dfrac{\Sigma\#\mathsf{a} : \Gamma \Rightarrow \phi \qquad \Sigma\#\mathsf{a} : \Gamma, \phi \Rightarrow \psi}{\Sigma\#\mathsf{a} : \Gamma \Rightarrow \psi}\,cut}{\Sigma : \Gamma \Rightarrow \psi}\,F$$

Applications

- Syntactic proof of **consistency**
- Proof of **conservativity** relative to Pitts' system
- **Sound and complete** translation from $FO\lambda^{\nabla}$ to $NL^{\Rightarrow}$

Translation from $FO\lambda^{\nabla}$ to nominal logic

- $FO\lambda^{\nabla}$ [Miller, Tiu 2003]: a logic with *local name contexts* σ and a self-dual local name quantifier $\nabla x.\phi$:

$$\frac{\Sigma : \Gamma, (\sigma, x) \triangleright \phi \Rightarrow \mathcal{A}}{\Sigma : \Gamma, \sigma \triangleright \nabla x.\phi \Rightarrow \mathcal{A}} \nabla L \quad \frac{\Sigma : \Gamma \Rightarrow (\sigma, x) \triangleright \phi}{\Sigma : \Gamma \Rightarrow \sigma \triangleright \nabla x.\phi} \nabla R \qquad (x \notin \Sigma, \sigma)$$

- [Gabbay, Cheney 2004] gave a sound but not complete translation to a nominal logic variant

- Incomplete because И admits "weakening", "exchange", but ∇ does not.

Examples of old translation

- translation of "weakening principle"

$$\nabla x.p \iff p \qquad \text{(underivable)}$$

is

$$\text{И}\mathsf{a}.p \iff p \qquad \text{(derivable!)}$$

- translation of "exchange principle"

$$\nabla x, y.p(x, y) \iff \nabla y, x.p(x, y) \qquad \text{(underivable)}$$

is

$$\text{И}\mathsf{a}, \mathsf{b}.p(n(\mathsf{a}), n(\mathsf{b})) \iff \text{И}\mathsf{b}, \mathsf{a}.p(n(\mathsf{a}), n(\mathsf{b})) \qquad \text{(derivable!)}$$

Examples of new translation

- translation of "weakening principle"

$$\nabla x.p \iff p \qquad \text{(underivable)}$$

is

$$\text{И}\mathsf{a}.p[\mathsf{a}] \iff p[] \qquad \text{(underivable)}$$

- translation of "exchange principle"

$$\nabla x, y.p(x, y) \iff \nabla y, x.p(x, y) \qquad \text{(underivable)}$$

is

$$\text{И}\mathsf{a}, \mathsf{b}.p[\mathsf{a}, \mathsf{b}](n(\mathsf{a}), n(\mathsf{b})) \iff \text{И}\mathsf{b}, \mathsf{a}.p[\mathsf{b}, \mathsf{a}](n(\mathsf{a}), n(\mathsf{b})) \qquad \text{(underivable)}$$

Details of translation

$$
\begin{array}{rcll}
[\![\sigma \triangleright C]\!] & = & C & (C \in \{\top, \bot\}) \\
[\![\sigma \triangleright \neg\phi]\!] & = & \neg[\![\sigma \triangleright \phi]\!] & \\
[\![\sigma \triangleright \phi \otimes \psi]\!] & = & [\![\sigma \triangleright \phi]\!] \otimes [\![\sigma \triangleright \psi]\!] & (\otimes \in \{\wedge, \vee, \supset\}) \\
[\![\sigma \triangleright \forall x.\phi]\!] & = & \forall h.ev(h) \supset [\![\sigma \triangleright \phi[h\sigma/x]]\!] & \\
[\![\sigma \triangleright \exists x.\phi]\!] & = & \exists h.ev(h) \wedge [\![\sigma \triangleright \phi[h\sigma/x]]\!] & \\
[\![\sigma \triangleright \nabla x.\phi]\!] & = & [\![\sigma, x \triangleright \phi]\!] & \\
[\![\sigma \triangleright p\vec{t}]\!] & = & \text{И}\vec{\sigma}.p[\sigma]\vec{t} &
\end{array}
$$

Details of translation

$$
\begin{array}{rcll}
[\![\sigma \triangleright C]\!] & = & C & (C \in \{\top, \bot\} \\
[\![\sigma \triangleright \neg\phi]\!] & = & \neg[\![\sigma \triangleright \phi]\!] & \\
[\![\sigma \triangleright \phi \otimes \psi]\!] & = & [\![\sigma \triangleright \phi]\!] \otimes [\![\sigma \triangleright \psi]\!] & (\otimes \in \{\wedge, \vee, \supset\}) \\
[\![\sigma \triangleright \forall x.\phi]\!] & = & \forall h.ev(h) \supset [\![\sigma \triangleright \phi[h\sigma/x]]\!] & \\
[\![\sigma \triangleright \exists x.\phi]\!] & = & \exists h.ev(h) \wedge [\![\sigma \triangleright \phi[h\sigma/x]]\!] & \\
[\![\sigma \triangleright \nabla x.\phi]\!] & = & [\![\sigma, x \triangleright \phi]\!] & \\
[\![\sigma \triangleright p\vec{t}]\!] & = & Ν\vec{\sigma}.p[\sigma]\vec{t} &
\end{array}
$$

Note: Translation is homomorphic on propositional connectives

Details of translation

$$
\begin{array}{rcll}
[\![\sigma \rhd C]\!] & = & C & (C \in \{\top, \bot\}) \\
[\![\sigma \rhd \neg\phi]\!] & = & \neg[\![\sigma \rhd \phi]\!] & \\
[\![\sigma \rhd \phi \otimes \psi]\!] & = & [\![\sigma \rhd \phi]\!] \otimes [\![\sigma \rhd \psi]\!] & (\otimes \in \{\wedge, \vee, \supset\}) \\
[\![\sigma \rhd \forall x.\phi]\!] & = & \forall h.ev(h) \supset [\![\sigma \rhd \phi[h\sigma/x]]\!] & \\
[\![\sigma \rhd \exists x.\phi]\!] & = & \exists h.ev(h) \wedge [\![\sigma \rhd \phi[h\sigma/x]]\!] & \\
[\![\sigma \rhd \nabla x.\phi]\!] & = & [\![\sigma, x \rhd \phi]\!] & \\
[\![\sigma \rhd p\vec{t}]\!] & = & И\vec{\sigma}.p[\sigma]\vec{t} &
\end{array}
$$

Note: We lift $\forall, \exists$ to make local context dependence explicit

(Here $ev(h) = \forall a :$ A.$a \# h$)

Details of translation

$$
\begin{array}{rcll}
[\![\sigma \triangleright C]\!] & = & C & (C \in \{\top, \bot\}) \\
[\![\sigma \triangleright \neg\phi]\!] & = & \neg[\![\sigma \triangleright \phi]\!] & \\
[\![\sigma \triangleright \phi \otimes \psi]\!] & = & [\![\sigma \triangleright \phi]\!] \otimes [\![\sigma \triangleright \psi]\!] & (\otimes \in \{\wedge, \vee, \supset\}) \\
[\![\sigma \triangleright \forall x.\phi]\!] & = & \forall h.ev(h) \supset [\![\sigma \triangleright \phi[h\sigma/x]]\!] & \\
[\![\sigma \triangleright \exists x.\phi]\!] & = & \exists h.ev(h) \wedge [\![\sigma \triangleright \phi[h\sigma/x]]\!] & \\
[\![\sigma \triangleright \nabla x.\phi]\!] & = & [\![\sigma, x \triangleright \phi]\!] & \\
[\![\sigma \triangleright p\vec{t}]\!] & = & \text{И}\vec{\sigma}.p[\sigma]\vec{t} &
\end{array}
$$

Note: We delay using И for ∇ by storing ∇-quantified names in local context.

Details of translation

$$
\begin{array}{rcll}
[\![\sigma \triangleright C]\!] & = & C & (C \in \{\top, \bot\} \\
[\![\sigma \triangleright \neg\phi]\!] & = & \neg[\![\sigma \triangleright \phi]\!] & \\
[\![\sigma \triangleright \phi \otimes \psi]\!] & = & [\![\sigma \triangleright \phi]\!] \otimes [\![\sigma \triangleright \psi]\!] & (\otimes \in \{\wedge, \vee, \supset\}) \\
[\![\sigma \triangleright \forall x.\phi]\!] & = & \forall h.ev(h) \supset [\![\sigma \triangleright \phi[h\sigma/x]]\!] & \\
[\![\sigma \triangleright \exists x.\phi]\!] & = & \exists h.ev(h) \wedge [\![\sigma \triangleright \phi[h\sigma/x]]\!] & \\
[\![\sigma \triangleright \nabla x.\phi]\!] & = & [\![\sigma, x \triangleright \phi]\!] & \\
[\![\sigma \triangleright p\vec{t}]\!] & = & \text{И}\vec{\sigma}.p[\sigma]\vec{t} &
\end{array}
$$

Note: We translate local contexts to *И-quantified names*

Note also: We also parameterize translated atomic formulas by list of local names.

Idea of proof

- Identify a normal form for NL derivations
- Prove that all normal forms represent $FO\lambda^{\nabla}$ proofs
- Prove that all derivations of translated $FO\lambda^{\nabla}$ sequents can be normalized.
- **Many details omitted here.**

Some details

- "First normal form": derivation consists only of И, hyp, or equational,freshness, or equivariance laws.

- Example: $[\![\Sigma : \Gamma, x \triangleright p \Rightarrow x \triangleright p]\!]$ derivable as

$$\frac{}{\Sigma : \Gamma, \text{И}x.p[x] \Rightarrow \text{И}x.p[x]}\ hyp^*$$

  which expands to 1NF.

  Proposition 8. *$[\![\Sigma : \Gamma \Rightarrow \mathcal{A}]\!]$ is in 1NF if and only if $\Sigma : \Gamma \Rightarrow \mathcal{A}$ is an initial sequent (i.e., $\mathcal{A} \in \Gamma$).*

  By induction on derivations (using knowledge of translation).

More details

- "Second normal form": derivation starts with a logical rule.

- If the first rule is $\forall$ (or $\exists$) then it must be followed by corresponding $\supset$ (or $\wedge$) on the same formula.

  Proposition 9. *A translated sequent has a 2NF derivation if and only if there exists a $FO\lambda^{\nabla}$ logical rule instance*

$$\frac{J_1 \quad \cdots \quad J_n}{\Sigma : \Gamma \Rightarrow \mathcal{A}}$$

  such that the translations $[\![J_1]\!], \ldots, [\![J_n]\!]$ are also derivable.

More details

- So far so good. The hard part is proving that that translated derivations have normal forms.

  Proposition 10. *If* $[\![J]\!]$ *has a* $NL^{\Rightarrow}$ *derivation, then it has a 1NF or 2NF derivation.*

  The proof is by complicated induction on a strengthened induction hypothesis.

  Theorem 11. *If* $[\![J]\!]$ *is derivable in* $NL^{\Rightarrow}$*, then* J *is derivable in* $FO\lambda^{\nabla}$*.*

Related work

- Many spatial/tree/graph/concurrency logics now incorporating И (e.g., [Caires, Cardelli 2002])

- [Gabbay, Cheney 2004]: presented an alternative system with И-rules using more complex syntactic side-conditions

- [Schöpp, Stark 2004]: develop a dependent type theory with names & binding using similar (but more general) *bunched contexts*

- [Miculan, Yemane 2005] describe an (incomplete) denotational semantics of $FO\lambda^{\nabla}$.

Future work

- Uniform proof semantics of nominal logic programming

- Semantics of $FO\lambda^{\nabla}$

- A *truly* simple proof theory?

- A *simple* type theory?

Conclusions

- Presented a proof theory for nominal logic that uses explicit name symbols and structured contexts to deal with И
- We argue that this approach is "simpler" / "easier to use"; this is subjective
- Re-proved existing results (cut-elimination, consistency, conservativity)
- In addition, proved a nontrivial new result (embedding of $FO\lambda^{\nabla}$).