A Third-Order Representation of the $\lambda\mu$-Calculus

Andreas Abel

Workshop MERLIN 2001, IJCAR, Siena
June 18, 2001

Slide 1

1. Parigot's $\lambda\mu$-Calculus
2. Computational Interpretation
3. A Third-Order Representation of the $\lambda\mu$-Calculus
4. Applications

Acknowledgements: Ralph Matthes, Frank Pfenning, Brigitte Pientka

Parigot's $\lambda\mu$-Calculus

Slide 2

- Proof terms for the implicational fragment of classical logic in natural deduction style
- Raw terms:

$$\begin{array}{lcll} M & ::= & x \mid \lambda x.M \mid M\ M \mid \mu a.N & \textit{(Unnamed) terms} \\ N & ::= & [a]M & \textit{Named terms} \end{array}$$

- Typing rules resp. proof rules:

$$\frac{\begin{array}{c}\overline{x : A} \\ \vdots \\ M : B\end{array}}{\lambda x.M : A \to B}\ {\to}\mathcal{I}^x \qquad \frac{M_1 : A \to B \qquad M_2 : A}{M_1\ M_2 : B}\ {\to}\mathcal{E} \qquad \frac{a : \overline{A} \qquad M : A}{[a]M : -}\ \textsc{contra}$$

$$\frac{\begin{array}{c}\overline{a : \overline{A}} \\ \vdots \\ N : -\end{array}}{\mu a.N : A}\ \textsc{raa}^a$$

Example: Pierce-Formula

Slide 3

- Proof tree for $((A \to B) \to A) \to A$:

$$\dfrac{\dfrac{\overline{\overline{A}}^{a}}{\dfrac{\dfrac{\overline{\overline{B}}^{b}}{\dfrac{(A \to B) \to A}{A}\,x}\to\mathcal{E}}{\dfrac{\overline{A}}{\bot}\,\text{CONTRA}}\,\text{RAA}^{a}}\to\mathcal{I}^{x}}{((A \to B) \to A) \to A}$$

$$\dfrac{\dfrac{\overline{\overline{A}}^{a} \qquad \overline{A}^{y}}{\bot}\,\text{CONTRA}}{\dfrac{\overline{B}}{A \to B}\,\text{RAA}^{b}}\to\mathcal{I}^{y}$$

- Proof term for $((A \to B) \to A) \to A$:

$$\lambda x^{(A\to B)\to A}.\,\mu a^{\overline{A}}.\,[a](x\,\lambda y^{A}.\,\mu b^{\overline{B}}.\,[a]y)$$

Logical Reduction (β)

Slide 4

- β-reduction as in the λ-calculus

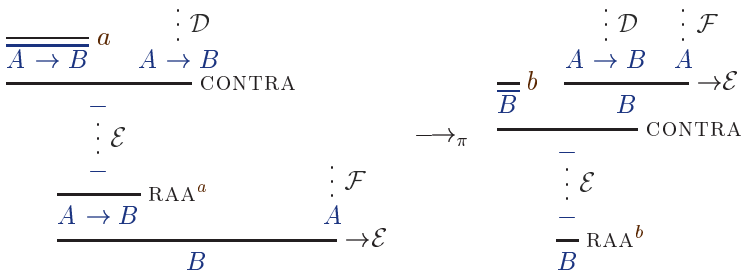
$$\dfrac{\dfrac{\begin{array}{c}\overline{A}\,x\\ \vdots\,\mathcal{D}\\ B\end{array}}{A \to B}\to\mathcal{I}^{x} \qquad \begin{array}{c}\vdots\,\mathcal{E}\\ A\end{array}}{B}\to\mathcal{E} \quad \longrightarrow_{\beta} \quad \begin{array}{c}\vdots\,\mathcal{E}\\ A\\ \vdots\,\mathcal{D}\\ B\end{array}$$

$$(\lambda x.M)N \quad \longrightarrow_{\beta} \quad [N/x]M$$

Structural Reduction (π)

Slide 5

- Uncover hidden β-redices:



$$(\mu a.N)\,M \quad \longrightarrow_\pi \quad \mu b.[\lambda^\circ z.[b](z\,M)/a]N$$

- Structural substitution incorporates silent linear β-reductions

$$[\lambda^\circ z.N]M \longrightarrow_{\beta^\circ} [M/z]N$$

- $\beta\pi$-normal terms fulfill the subformula property.

Typing Rules with Contexts

Slide 6

- λ-calculus (Γ: context of λ-variables; Δ: context of μ-variables)

$$\frac{x\colon A \in \Gamma}{\Gamma;\Delta \vdash_{\lambda\mu} x : A}\,\mathsf{var} \qquad \frac{\Gamma, x\colon A;\Delta \vdash_{\lambda\mu} M : B}{\Gamma;\Delta \vdash_{\lambda\mu} \lambda x.M : A \to B}\,{\to}\mathcal{I}^x$$

$$\frac{\Gamma;\Delta \vdash_{\lambda\mu} M_1 : A \to B \qquad \Gamma;\Delta \vdash_{\lambda\mu} M_2 : A}{\Gamma;\Delta \vdash_{\lambda\mu} M_1\,M_2 : B}\,{\to}\mathcal{E}$$

- μ-abstraction and application

$$\frac{a\colon \overline{A} \in \Delta}{\Gamma;\Delta \vdash_{\lambda\mu} a : \overline{A}}\,\mu\mathsf{var} \qquad \frac{\Gamma;\Delta, a\colon \overline{A} \vdash_{\lambda\mu} N : -}{\Gamma;\Delta \vdash_{\lambda\mu} \mu a.N : A}\,\mathrm{RAA}^a$$

$$\frac{\Gamma;\Delta \vdash_{\lambda\mu} C : \overline{A} \qquad \Gamma;\Delta \vdash_{\lambda\mu} M : A}{\Gamma;\Delta \vdash_{\lambda\mu} [C]M : -}\,\mathrm{CONTRA}$$

- linear contexts (not first class citizens)

$$\frac{\Gamma, z\colon A;\Delta \vdash_{\lambda\mu} N : -}{\Gamma;\Delta \vdash_{\lambda\mu} \lambda^\circ z.N : \overline{A}}\,\lambda^\circ$$

Further Reductions

Slide 7

- Renaming (β-reduction for μ-variables)

$$[a]\mu b.N \longrightarrow_{\mu\beta} [a/b]N$$

- η-reduction for μ-variables

$$\mu a.[a]M \longrightarrow_{\mu\eta} M \qquad (a \notin \mu\mathsf{FV}(M))$$

- Symmetrical reduction to the π-reduction

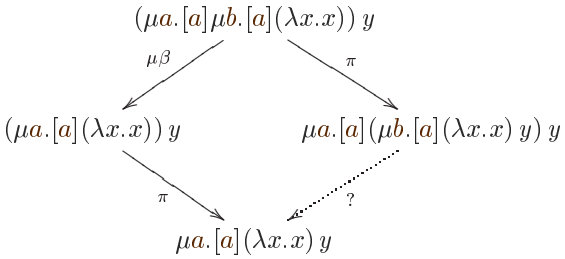
$$v\,(\mu a.N) \longrightarrow_{\pi_v} \mu b.[\lambda^\circ z.[b](v\,z)/b]N$$

- All five reductions appear in the small-step semantics given later.

Why the Diamond Lemma Fails

Slide 8

- A π-reduction can separate a $\mu\beta$-redex:



- A direct application of the Tait & Martin-Löf method fails to prove Church-Rosser.
- Baba, Hirokawa and Fujita recently offered a fix (CATS 2001).

A Small-Step Semantics: Transition System

Slide 9

- Values:

$$v ::= \lambda x.M$$

- Evaluation contexts as evaluation frame stacks:

$$\begin{array}{rcll} S & ::= & \mathsf{eval} & \ulcorner[\mathsf{eval}]M\urcorner = [\mathsf{eval}]M \\ & | & (\bullet\, M) :: S & \ulcorner[(\bullet\, M_2) :: S]M_1\urcorner = \ulcorner[S]M_1\, M_2\urcorner \\ & | & (v\, \bullet) :: S & \ulcorner[(v\, \bullet) :: S]M\urcorner = \ulcorner[S]v\, M\urcorner \end{array}$$

- Call-by-value small-step semantics:

$$\begin{array}{lll} (\lambda) & [(\bullet\, M_2) :: S]\lambda x M_1 \longrightarrow [(\lambda x M_1\, \bullet) :: S]M_2 \\ & [(\lambda x M\, \bullet) :: S]v \longrightarrow [S]([v/x]M) \\ (\mathsf{app}) & [S]M_1\, M_2 \longrightarrow [(\bullet\, M_2) :: S]M_1 \\ (\mu) & [S]\mu a.N \longrightarrow [S/a]N \end{array}$$

- M evaluates to v iff

$$[\mathsf{eval}]M \longrightarrow^* [\mathsf{eval}]v$$

deGroote's Exception Handling Calculus

Slide 10

- An SML-like exception handling mechanism:

$$\frac{\begin{array}{cc}\overline{\overline{e : \overline{A}}} & \overline{x : A} \\ \vdots & \vdots \\ M : C & M_e : C\end{array}}{(\mathsf{exception}\, e\, \mathsf{in}\, M\, \mathsf{handle}\, e(x).M_e) : C}\ \text{Handle} \qquad \frac{e : \overline{A} \qquad M : A}{\mathsf{raise}\, e(M) : C}\ \text{Raise}$$

- Desired computational behaviour:

$$\begin{array}{lll} \mathsf{exception}\, e\, \mathsf{in}\, E[\,\mathsf{raise}\, e(v)]\, \mathsf{handle}\, e(x).M_e & \rightsquigarrow & [v/x]M_e \\ \mathsf{exception}\, e\, \mathsf{in}\, v\, \mathsf{handle}\, e(x).M_e & \rightsquigarrow & v \end{array}$$

- Encoding into the $\lambda\mu$-calculus

$$\begin{array}{lcl} \ulcorner\mathsf{raise}\, e(M)\urcorner & = & (\lambda y.\mu_.[e]y)\ulcorner M\urcorner \\ \ulcorner\mathsf{exception}\, e\, \mathsf{in}\, M\, \mathsf{handle}\, e(x).M_e\urcorner & = & \mu a.[a](\lambda x.\ulcorner M_e\urcorner)\,(\mu e.[a]\ulcorner M\urcorner) \end{array}$$

The Logical Framework LF

Slide 11

- Edinburgh Logical Framework [Harper, Honsell, Plotkin '93]
- Term language: dependently-typed lambda-calculus λ^{Π}
- Admits higher-order abstract syntax
- For free:
  - α-equivalence
  - substitution
- Implementation: Twelf [Pfenning, Schürmann et al.]

Higher-Order Representation of the Typed λ-Calculus

Slide 12

$$
\begin{array}{lcl}
\mathsf{ty} & : & \mathsf{type}. \\
\mathsf{i} & : & \mathsf{ty}. \\
\Rightarrow & : & \mathsf{ty} \to \mathsf{ty} \to \mathsf{ty}. \\
\mathsf{tm} & : & \mathsf{ty} \to \mathsf{type}. \\
 & & \\
\mathsf{lam} & : & (\mathsf{tm}\, A \to \mathsf{tm}\, B) \to \mathsf{tm}(A \Rightarrow B). \\
\mathsf{app} & : & \mathsf{tm}(A \Rightarrow B) \to \mathsf{tm}\, A \to \mathsf{tm}\, B.
\end{array}
\qquad
\begin{array}{lcl}
\ulcorner \iota \urcorner & = & \mathsf{i} \\
\ulcorner A \to B \urcorner & = & \ulcorner A \urcorner \Rightarrow \ulcorner B \urcorner \\
 & & \\
\ulcorner x \urcorner & = & x \\
\ulcorner \lambda x^{A}.M \urcorner & = & \mathsf{lam}\ \lambda x{:}\mathsf{tm}\ \ulcorner A \urcorner.\ulcorner M \urcorner \\
\ulcorner M_1\, M_2 \urcorner & = & \mathsf{app}\ \ulcorner M_1 \urcorner\ \ulcorner M_2 \urcorner
\end{array}
$$

Towards a Representation of the $\lambda\mu$-Calculus

Slide 13

- How to represent μ-abstraction?
- General form of a binder:

$$bind : (\rho \to \sigma) \to \tau \qquad \text{where} \begin{cases} \tau & : & \text{type of result expression} \\ \sigma & : & \text{type of body expression} \\ \rho & : & \text{type of variable} \end{cases}$$

- Extend signature:

$$\begin{array}{lcll} \mathsf{nam} & : & \mathsf{type}. & \textit{Named terms.} \\ \mathsf{mu} & : & (\rho \to \mathsf{nam}) \to \mathsf{tm}\ A. & \mu\textit{-abstraction} \end{array}$$

- Key question: What do μ-variables stand for?
- Semantics suggests: contexts.

$$\rho = \mathsf{tm}\ A \to \mathsf{nam}$$

- μ is a third-order constant.

Third-Order Representation of the $\lambda\mu$-Calculus

Slide 14

- Representation:

$$\mathsf{mu} : ((\mathsf{tm}\ A \to \mathsf{nam}) \to \mathsf{nam}) \to A$$

$$\begin{array}{lcrcl} \ulcorner \mu a^{\overline{A}}.N \urcorner & = & \mathsf{mu}\ \lambda a{:}\mathsf{tm}\ A \to \mathsf{nam}.\ \ulcorner N \urcorner & : & \mathsf{tm} \\ \ulcorner [C]M \urcorner & = & \ulcorner C \urcorner\ \ulcorner M \urcorner & : & \mathsf{nam} \\ \ulcorner a \urcorner & = & a & : & \mathsf{tm}\ A \to \mathsf{nam} \\ \ulcorner \lambda^{\circ} z^{A}.N \urcorner & = & \lambda z{:}\mathsf{tm}\ A.\ \ulcorner N \urcorner & : & \mathsf{tm}\ A \to \mathsf{nam} \end{array}$$

- No constant for μ-application required.
- Instead LF-application is used.
- Since meta theory comes with $\beta\eta$-equality, linear β-reductions are silent indeed.

Theorem: Adequacy of the Representation

1. Forall $\Gamma = x_1 : A_1, \ldots, x_n : A_n$ and $\Delta = a_1 : \overline{B_1}, \ldots, a_m : \overline{B_m}$:

$$\text{If } \Gamma; \Delta \vdash_{\lambda\mu} \begin{cases} M : A \\ N : - \\ C : \overline{A} \end{cases} \quad \text{then } \ulcorner\Gamma\urcorner, \ulcorner\Delta\urcorner \vdash \begin{cases} \ulcorner M \urcorner : \mathsf{tm}\, \ulcorner A \urcorner \\ \ulcorner N \urcorner : \mathsf{nam} \\ \ulcorner C \urcorner : \mathsf{tm}\, \ulcorner A \urcorner \to \mathsf{nam} \end{cases}$$

2. Forall $\Phi = x_1 : \mathsf{tm}\, p_1, \ldots, x_n : \mathsf{tm}\, p_n, a_1 : \mathsf{tm}\, q_1 \to \mathsf{nam}, \ldots, a_m : \mathsf{tm}\, q_m \to \mathsf{nam}$:

Slide 15

$$\text{If } \Phi \vdash \begin{cases} t : \mathsf{tm}\, p \\ s : \mathsf{nam} \\ r : \mathsf{tm}\, p \to \mathsf{nam} \end{cases} \quad \text{then } \Gamma; \Delta \vdash_{\lambda\mu} \begin{cases} M : A \\ N : - \\ C : \overline{A} \end{cases} \quad \text{where } \begin{array}{l} \ulcorner M \urcorner = t, \ulcorner A \urcorner = p \\ \ulcorner N \urcorner = s \\ \ulcorner C \urcorner = r, \ulcorner A \urcorner = p \end{array}$$

for some $M, N, C, A, \Gamma, \Delta$ with $\ulcorner\Gamma\urcorner, \ulcorner\Delta\urcorner = \Phi$

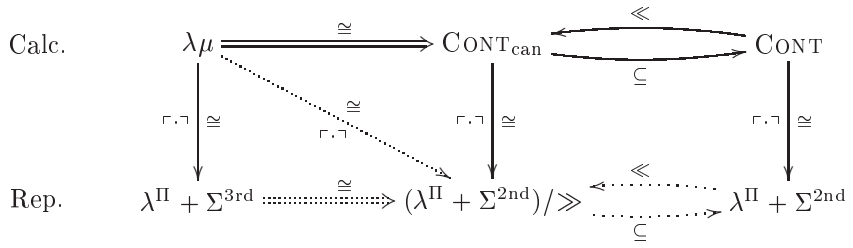
3. The function $\ulcorner\cdot\urcorner$ is a bijection and compositional in the following sense:
   (a) $\ulcorner [M/x] H \urcorner = [\ulcorner M \urcorner / x] \ulcorner H \urcorner$. $\quad (H = M, N)$
   (b) $\ulcorner [C/a] H \urcorner = [\ulcorner C \urcorner / a] \ulcorner H \urcorner$.

Application 1: Second-Order Representation

- Second-order representation: $\mathsf{cont}\, A$ for $\mathsf{tm}\, A \to \mathsf{nam}$.
- Calculus Cont: Continuations are first-class citizens.
- Define canonical fragments Cont_{can}: represents $\lambda\mu$-calculus.
- $\gg : \text{Cont} \to \text{Cont}_{\text{can}}$: eliminate β°-redices.

Slide 16

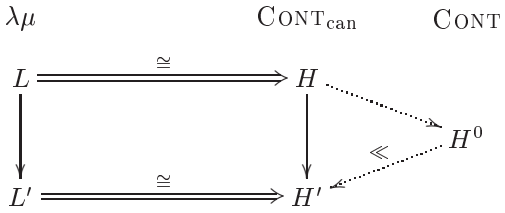
- Goal: second-order representation isomorphic to third-order representation.



Second-Order Representation: Bisimulation

Slide 17

- Show: each reduction step in $\lambda\mu$ corresponds to one reduction step in $\textsc{Cont}_{\text{can}}$.



- π-reductions create β°-redices, lead out of canonical fragment.
- $\gg$-function has to be applied
- Substitution lemma required.
- Proof formalized in Twelf.

Application 2: Big-step Semantics

Slide 18

- Ong and Stewart gave a big-step semantics $M \Downarrow v$.
- Soundness: If $M \Downarrow v$ then $[\mathsf{eval}]M \longrightarrow^* [\mathsf{eval}]v$.
- Completeness: If $[S]M \longrightarrow^* [\mathsf{eval}]v$ then $M \Downarrow v'$ and $[S]v' \longrightarrow^* [\mathsf{eval}]v$.
- Proofs are (almost completely) formalized in Twelf.

Application 3: Confluence

- Parallel-reduction method a là Martin-Löf and Tait.
- $\mu\beta$-rule has to be generalized [BHF01]

Slide 19

$$\frac{C \Longrightarrow C' \qquad M \Longrightarrow M'}{[C]M \Longrightarrow [C]M}\,\mu\beta$$

- Need to reason about evaluation contexts C.
- Third-order representation not directly usable.
- Work in progress.

Conclusion

- Elegant representation of the $\lambda\mu$-calculus.
- New insights:
  - Structural substitution reduced to ordinary substitution. (Independently by de Groote [1998])
  - Concise formulation of small-step semantics.

Slide 20

- Case study for order > 2 representations.
- Reached limits of logical framework.

References

[BHF01] Kensuke Baba, Sachio Hirokawa, and Ken-etsu Fujita. Parallel reduction in type free $\lambda\mu$-calculus. In *Proceedings of CATS 2001 (Computing: the Australasian Theory Symposium)*, volume 42 of *Electronic Notes in Theoretical Computer Science*. Elsevier Science B. V., 2001. Also appeared as Technical Report DOI-TR-177, Kyushu University, Fukuoka, Japan.

[Bie98] Gavin M. Bierman. A computational interpretation of the $\lambda\mu$-calculus. In L. Brim, J. Gruska, and J. Zlatuska, editors, *Proceedings of Symposium on Mathematical Foundations of Computer Science*, volume 1450 of *Lecture Notes in Computer Science*, pages 336–345, Brno, Czech Republic, August 1998.

Slide 21

[dG98] Philippe de Groote. An environment machine for the $\lambda\mu$-calculus. To appear in MSCS, 1998.

[OS97] C.-H. L. Ong and C. A. Stewart. A Curry-Howard foundation for functional computation with control. In *Proceedings of ACM SIGPLAN-SIGACT Symposium on Principle of Programming Languages*, pages 215–227, Paris, Januar 1997. ACM Press.

[Par92] M. Parigot. $\lambda\mu$-calculus: An algorithmic interpretation of classical natural deduction. In A. Voronkov, editor, *Logic Programming and Automated Reasoning: Proc. of the International Conference LPAR'92*, pages 190–201. Springer, Berlin, Heidelberg, 1992.

[Pfe01] Frank Pfenning. Logical frameworks. In Alan Robinson and Andrei Voronkov, editors, *Handbook of Automated Reasoning*. Elsevier Science Publishers B.V. and MIT Press, 2001. In preparation. Draft available from http://www.cs.cmu.edu/~fp.

Slide 22