Untyped Algorithmic Equality for Martin-Löf's Logical Framework with Surjective Pairs

Andreas Abel

joint work with Thierry Coquand

Slide 1

Arbeitstreffen Bern-München 2005
Munich, Germany
December 08, 2005

Work supported by: TYPES & APPSEM-II (EU), CoVer (SSF)

Background: $\beta\eta$-equality

Slide 2

- Checking dependent types requires equality test
- One approach: reduce to normal form and compare syntactically
- Works fine for β-equality
- Problem with η-reduction: surjective pairing destroys confluence (Klop 1980)
- Even subject reduction fails:

$$z : \mathsf{Pair}\, A\, (\lambda x.\, F\, x) \;\vdash\; (z\,\mathsf{L},\, z\,\mathsf{R}) \;:\; \mathsf{Pair}\, A\, (\lambda_.\, F\, (z\,\mathsf{L}))$$

[I write $\mathsf{Pair}\, A\, (\lambda x B)$ for $\Sigma x{:}A.\, B$]

Slide 3

Coquand's Equality Algorithm

- Incremental check for $\beta\eta$-equality in dependently-typed λ-calculus (Coquand 1991)
- Alternates weak head normalization and comparison of head symbols
- We extend this algorithm to Σ-types with surjective pairing
- Challenge: termination and completeness
- Two major technical difficulties to overcome

Slide 4

Martin-Löf's Logical Framework (MLF)

- Expressions = Curry-style λ-terms

c	$::=$	$\mathsf{Fun} \mid \mathsf{El} \mid \mathsf{Set}$	constants
r, s, t	$::=$	$c \mid x \mid \lambda x t \mid r\, s$	expressions
A, B, C	$::=$	$\mathsf{Set} \mid \mathsf{El}\, t \mid \mathsf{Fun}\, A\, (\lambda x B)$	types

- Examples

$\mathsf{Fun}\, A\, (\lambda x B)$	dependent function space $\Pi x : A.\, B$
$\mathsf{Fun}\, \mathsf{Set}\, (\lambda a.\, \mathsf{Fun}\, (\mathsf{El}\, a)\, (\lambda_.\, \mathsf{El}\, a))$	type of identity: $\forall a : *.\, a \to a$

Martin-Löf's logical framework (Typing)

Slide 5

- Judgements for typing and equality, e.g.,

$$\begin{array}{ll} \Gamma \vdash A : \mathsf{Type} & A \text{ is a well-formed type} \\ \Gamma \vdash t : A & t \text{ has type } A \end{array}$$

- Example: application rule

$$\frac{\Gamma \vdash r : \mathsf{Fun}\, A\, (\lambda x B) \qquad \Gamma \vdash s : A}{\Gamma \vdash r\, s : B[s/x]}$$

Untyped conversion?

Slide 6

- Untyped conversion rule problematic

$$\frac{\Gamma \vdash t : A \qquad \Gamma \vdash B : \mathsf{Type}}{\Gamma \vdash t : B}\ A =_{\beta\eta} B$$

- Injectivity of type constructors, e.g., $\mathsf{El}\, t =_{\beta\eta} \mathsf{El}\, t'$ implies $t =_{\beta\eta} t'$, needed for soundness proof.
- Not provable since equality cannot a priori be defined as $\searrow_* \swarrow$.

Judgemental equality

- Judgements for typing and equality, e.g.,

$\Gamma \vdash A = A' : \mathsf{Type}$	A and A' are equal types
$\Gamma \vdash t = t' : A$	t and t' are equal terms of type A

Slide 7

- Example: β- and η-rules

$$\frac{\Gamma, x{:}A \vdash t = t' : B \qquad \Gamma \vdash s = s' : A}{\Gamma \vdash (\lambda x t)\, s = t'[s'/x] : B[s/x]}$$

$$\frac{\Gamma \vdash t = t' : \mathsf{Fun}\, A\, (\lambda x B)}{\Gamma \vdash (\lambda x.\, t\, x) = t' : \mathsf{Fun}\, A\, (\lambda x B)}\ x \notin \mathsf{FV}(t)$$

Lambda Algebra

- Entities

v, f, V, F	$\in$	D	elements of the algebra
ρ	$\in$	$\mathsf{Var} \to \mathsf{D}$	environments

Slide 8

- Operations

$f \cdot v$	$\in$	D	application in the algebra
$t\rho$	$\in$	D	denotation of expression t in environment ρ

Lambda Algebra Axiomatization

Slide 9

Congruences

$$\begin{aligned} c\rho &= c \\ x\rho &= \rho(x) \\ (r\,s)\rho &= r\rho \cdot (s\rho) \end{aligned}$$

Computation (β)

$$(\lambda x t)\rho \cdot v = t(\rho, x{=}v)$$

Injectivity

$$\begin{aligned} \mathsf{El} \cdot v &= \mathsf{El} \cdot v' && \text{implies } v = v' \\ \mathsf{Fun} \cdot V \cdot F &= \mathsf{Fun} \cdot V' \cdot F' && \text{implies } V = V' \text{ and } F = F' \end{aligned}$$

PER Model

Slide 10

- Assume a basic partial equivalence relation (PER) $\mathcal{S}$ on D
- Interpretation of *types* in D as sub-PERs of $\mathcal{S}$

$$\begin{aligned} [\mathsf{Set}] &= \mathcal{S} \\ [\mathsf{El} \cdot v] &= \mathcal{S} \\ [\mathsf{Fun} \cdot V \cdot F] &= \{(f, f') \mid (f \cdot v, f' \cdot v') \in [F \cdot v] \text{ for all } (v, v') \in [V]\} \end{aligned}$$

- Soundness of typing and equality rules

$$\begin{aligned} &\text{If } \Gamma \vdash t \quad\;\; : A \text{ then } (t\,\rho,\ t\,\rho) \in [A\rho] \text{ for all } \rho \in [\Gamma]. \\ &\text{If } \Gamma \vdash t = t' : A \text{ then } (t\,\rho,\ t'\rho) \in [A\rho] \text{ for all } \rho \in [\Gamma]. \end{aligned}$$

- Implication: $(t\,\rho,\ t'\rho) \in \mathcal{S}$

Substitution and Extensionality

Slide 11

- Difficulty 1: Soundness proof of application rule

$$\frac{\Gamma \vdash r : \mathsf{Fun}\, A\, (\lambda x B) \qquad \Gamma \vdash s : A}{\Gamma \vdash r\, s : B[s/x]}$$

- requires substitution property $(B[s/x])\rho = B(\rho, x{=}s\rho)$.
- Hence, need *λ-model* instead of λ-algebra.
- Additional axiom: weak extensionality

$$(\xi) \quad (\lambda x t)\rho = (\lambda x t')\rho' \quad \text{if } t(\rho, x{=}v) = t'(\rho', x{=}v) \text{ for all } v \in \mathsf{D}$$

- Irrelevance $t(\rho, x{=}v) = t\rho$ if $x \notin \mathsf{FV}(t)$, needed for η, now admissible.

Alternative λ-Model Axiomatization

Slide 12

- Benzmüller, Brown, Kohlhase (JSL 2004):

Congruences

$$\begin{aligned} x\rho &= \rho(x) \\ (r\, s)\rho &= r\rho \cdot (s\rho) \end{aligned}$$

Computation (β)

$$t\rho = t'\rho \qquad \text{if } t =_\beta t'$$

Irrelevance

$$t\rho = t\rho' \qquad \text{if } \rho(x) = \rho'(x) \text{ for all } x \in \mathsf{FV}(t)$$

- Also has substitution property $(t[s/x])\rho = t(\rho, x{=}s\rho)$.

Comparing Notions of λ-Model

Slide 13

- Every λ-model is a BBK-model.
- Not every BBK-model is a λ-model.
- Instance: closed terms modulo $\beta\eta$.
- Plotkin 1974: ω-rule fails in $\lambda\beta\eta$-calculus.

$$(\omega) \quad \frac{r\,t =_{\beta\eta} s\,t \text{ for all closed } t}{r =_{\beta\eta} s}$$

- Hence, ξ not valid.
- What about closed terms modulo β?

Weak head evaluation

Slide 14

- Weak head values

$$\begin{array}{rcll} n & ::= & c\,\vec{t} \mid x\,\vec{t} & \text{neutral expressions} \\ w & ::= & n \mid \lambda x t & \text{weak head values} \end{array}$$

- Weak head evaluation (call-by-name)

$$\begin{array}{rcll} (r\,s)\!\downarrow & := & r\!\downarrow @ s & \\ t\!\downarrow & := & t & t \text{ not application} \\ & & & \\ n @ s & := & n\,s & \\ (\lambda x t) @ s & := & (t[s/x])\!\downarrow & \end{array}$$

Untyped Algorithmic $\beta\eta$-Equality

Slide 15

- $\beta\eta$-conversion test for normalizable weak head values $w \sim w'$
- Two neutral expressions

$$\frac{}{c \sim c} \qquad \frac{}{x \sim x} \qquad \frac{n \sim n' \qquad s\downarrow \sim s'\downarrow}{n\,s \sim n'\,s'}$$

- At least one λ

$$\frac{t\downarrow \sim t'\downarrow}{\lambda x t \sim \lambda x t'} \qquad \frac{t\downarrow \sim n\,x}{\lambda x t \sim n} \qquad \frac{n\,x \sim t'\downarrow}{n \sim \lambda x t'}$$

- Relation $\sim$ is a PER

Transitivity of Algorithmic Equality

Slide 16

- Lemma:
  1. If $\mathcal{D}_1 :: w \sim n\,\vec{x}$ and $\mathcal{D}_2 :: n \sim n'$ then $w \sim n'\,\vec{x}$ (plus symmetrical proposition).
  2. If $\mathcal{D}_1 :: w_1 \sim w_2$ and $\mathcal{D}_2 :: w_2 \sim w_3$ then $w_1 \sim w_3$.
- Proof by simultaneous induction on $\mathcal{D}_1$ and $\mathcal{D}_2$.
- 1. is needed for the following case of 2.

$$\mathcal{D}_1 = \frac{\begin{array}{c}\mathcal{D}_1' \\ t\downarrow \sim n\,x\end{array}}{\lambda x t \sim n} \qquad \begin{array}{c}\mathcal{D}_2 \\ n \sim n'\end{array}$$

Completeness of Algorithmic Equality

Slide 17

- Recall: $\vdash t = t' : A$ implies $(t, t') \in \mathcal{S}$
- Take model instance

$$\begin{array}{lcl} \mathsf{D} & = & \beta\text{-equivalence classes} \\ f \cdot v & = & \overline{f\,v} \\ t\rho & = & \overline{t[\rho]} \\ \mathcal{S} & = & \text{lifted algorithmic equality } \sim \end{array}$$

- algorithmic equality on β-equivalence classes

$$\overline{t} \sim \overline{t'} \;:\!\Longleftrightarrow\; t =_\beta v \text{ and } t' =_\beta v' \text{ for some } v, v' \text{ with } v \sim v'$$

Standardization

Slide 18

- Using standardization, $\overline{t} \sim \overline{t'}$ implies $t{\downarrow} \sim t'{\downarrow}$.
- Summary (ρ_0 is identity valuation):

$$\begin{array}{c} \Gamma \vdash t = t' : A \\ \Big\Downarrow \text{ Soundness of judgement} \\ (t\rho_0, t'\rho_0) \in [A\rho_0] \\ \Big\Downarrow\; [A\rho_0] \subseteq \mathcal{S} \\ \overline{t} \sim \overline{t'} \\ \Big\Downarrow \text{ Standardization} \\ t{\downarrow} \sim t'{\downarrow} \end{array}$$

Extension to Σ-types

Slide 19

- Expressions

$$
\begin{array}{lcll}
c & ::= & \cdots \mid \mathsf{Pair} & \text{constants} \\
r, s, t & ::= & \cdots \mid (r, s) \mid t\,\mathsf{L} \mid t\,\mathsf{R} & \text{expressions} \\
A, B, C & ::= & \cdots \mid \mathsf{Pair}\,A\,(\lambda x B) & \text{types}
\end{array}
$$

- Example: $\mathsf{Pair}\,A\,(\lambda x B)$ dependent type of pairs $(\Sigma x{:}A.\,B)$
- Surjective pairing rule

$$
\frac{\Gamma \vdash r = r' : \mathsf{Pair}\,A\,(\lambda x B)}{\Gamma \vdash (r\,\mathsf{L},\ r\,\mathsf{R}) = r' : \mathsf{Pair}\,A\,(\lambda x B)}
$$

η-Reduction Destroys Subject Reduction

Slide 20

- Pair intro: types of s and t *do not* determine type of (s, t)

$$
\frac{\Gamma \vdash s : A \qquad \Gamma \vdash t : B[s/x]}{\Gamma \vdash (s, t) : \mathsf{Pair}\,A\,(\lambda x B)}
$$

- E.g., if $B[s/x] = \mathsf{Eq}\,A\,s\,s$, then $B \in \{\mathsf{Eq}\,A\,x\,x,\ \mathsf{Eq}\,A\,x\,s,\ \ldots\}$
- Change typing through η-expansion

$$
\frac{\dfrac{z : \mathsf{Pair}\,A\,(\lambda x B)}{z\,\mathsf{L} : A} \qquad \dfrac{z : \mathsf{Pair}\,A\,(\lambda x B)}{z\,\mathsf{R} : B[z\,\mathsf{L}/x]}}{(z\,\mathsf{L},\ z\,\mathsf{R}) \ :\ \mathsf{Pair}\,A\,(\lambda_.\,B[z\,\mathsf{L}/x])}
$$

- Subtyping does not solve this problem

Extended Algorithmic Equality

Slide 21

- Neutral expressions

$$\frac{n \sim n'}{n\,\mathsf{L} \sim n'\,\mathsf{L}} \qquad \frac{n \sim n'}{n\,\mathsf{R} \sim n'\,\mathsf{R}}$$

- At least one pair

$$\frac{r{\downarrow} \sim r'{\downarrow} \qquad s{\downarrow} \sim s'{\downarrow}}{(r,s) \sim (r',s')}$$

$$\frac{r{\downarrow} \sim n\,\mathsf{L} \qquad s{\downarrow} \sim n\,\mathsf{R}}{(r,s) \sim n} \qquad \frac{n\,\mathsf{L} \sim r'{\downarrow} \qquad n\,\mathsf{R} \sim s'{\downarrow}}{n \sim (r',s')}$$

Transitivity

Slide 22

- Problem 2: Alg. Eq. not transitive
- $\lambda x.\, z\, x \sim z$ and $z \sim (z\,\mathsf{L}, z\,\mathsf{R})$, but *not* $\lambda x.\, z\, x \sim (z\,\mathsf{L}, z\,\mathsf{R})$
- Solution: "Transitivization" $\overset{+}{\sim}$ through additional rules

$$\frac{t{\downarrow} \overset{+}{\sim} n\,x \qquad n\,\mathsf{L} \overset{+}{\sim} r \qquad n\,\mathsf{R} \overset{+}{\sim} s}{\lambda x t \overset{+}{\sim} (r,s)}$$

+ symmetrical rule

- If t, t' are of the same type, $t \overset{+}{\sim} t'$ does not use extra rules
- Equality $\sim$ *is* transitive for expressions of the same type

Proof of Transitivity

Slide 23

- Alternative 1: Direct. Technical, needs complicated measure.
- Alternative 2: Show that $\overset{+}{\sim}$ is equivalent to $=_\eta$ on β-normal forms.
  - Soundness: $v \overset{+}{\sim} v'$ implies $v =_\eta v'$.
  - Completeness 1: $v \overset{+}{\sim} v$ for β-normal form v.
  - Completeness 2: $v_1 \longrightarrow_\eta v_1'$ and $v_1' \overset{+}{\sim} v_2$ implies $v_1 \overset{+}{\sim} v_2$.

Summary of Completeness Proof

Slide 24

$$\Gamma \vdash t = t' : A$$

$$\Big\Downarrow \text{ Soundness of judgement}$$

$$(t\rho_0, t'\rho_0) \in [A\rho_0]$$

$$\Big\Downarrow [A\rho_0] \subseteq \mathcal{S}$$

$$\overline{t} \overset{+}{\sim} \overline{t'}$$

$$\Big\Downarrow \text{ Standardization}$$

$$t{\downarrow} \overset{+}{\sim} t'{\downarrow}$$

$$\Big\Downarrow \text{ Transitivity (with } \Gamma \vdash t, t' : A)$$

$$t{\downarrow} \sim t'{\downarrow}$$

Slide 25

Proof Economics

Injectivity	required
Inversion of typing	required
Standardization	required
Subject reduction	not required
Confluence (Church-Rosser)	not required
Normalization	not required
Certificate	good economics!

Slide 26

Related Work

- Vaux (2004): PER model for MLF with intersection
- Aspinall/Hofmann (TAPL II), Goguen (2005): completeness of algorithmic equality using standard meta theory
- Coquand, Pollack, and Takeyama (2003): extension of MLF by records with manifest fields
- Harper and Pfenning (2005): algorithmic equality for ELF directed by simple types (obtained by erasure)
- Schürmann and Sarnat (2004): extension to Σ-types
- Adams (2001): Luo's LF with Σ-kinds and type-directed equality

Future Work

- Logical framework with proof-irrelevant propositions
- Type-directed equality *without* erasure

Slide 27

Thanks to Frank Pfenning, Carsten Schürmann, and Lionel Vaux