On Equal μ-Terms

Jörg Endrulllis* Clemens Grabmayer[△]
Jan Willem Klop* Vincent van Oostrom[△]

[△]) Universiteit Utrecht
*) Vrije Universiteit Amsterdam

TeReSe (autumn),
VU Amsterdam
November 18, 2010

Overview

1. Weak μ-equality
2. Avoiding α-conversion in μ-reductions
3. Decidability of $=_{\mu/\alpha}$ by a first-order proof
4. Decidability of $=_{\mu/\alpha}$ by a higher-order proof
5. Decidability of $=_{\mu/\alpha}$ using regular languages
6. Summary

Finite representation of infinite pattern

finite representation?

Finite representation of infinite pattern

$$\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap \cdots$$

finite representation?

$$\mu x . \cap \cup x$$

Finite representation of infinite pattern

$$\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cdots$$

finite representation?

$$\mu x.\cap\cup x$$

with μ-rule

$$\mu x.s \to s[\![x := \mu x.s]\!]$$

Finite representation of infinite pattern

$$\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cdots$$

finite representation?

$$\mu x.\cap\cup x$$

with μ-rule

$$\mu x.s \to s[\![x := \mu x.s]\!]$$

$$\mu x.\cap\cup x \to \cap\cup\mu x.\cap\cup x \to \cap\cup\cap\cup\mu x.\cap\cup x \to \cap\cup\cap\cup\cap\cup\mu x.\cap\cup x \to \ldots$$

Finite representation of infinite pattern

$$\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cdots$$

finite representation?

$$\mu x.\cap\cup x$$

with μ-rule

$$\mu x.s \to s[\![x := \mu x.s]\!]$$

$$\mu x.\cap\cup x \to \cap\cup\mu x.\cap\cup x \to \cap\cup\cap\cup\mu x.\cap\cup x \to \cap\cup\cap\cup\cap\cup\mu x.\cap\cup x \to \dots$$

hieroglyph 𓈖 $\Rightarrow$ phoenician 𐤌 $\Rightarrow$ greek μ

Finite representations of infinite pattern

$$\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cdots$$

represented by

$$\mu x.\cap\cup x$$

other representations of same pattern?

Finite representations of infinite pattern

$$\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cdots$$

represented by

$$\mu x.\cap\cup x$$

other representations of same pattern?

$$\mu x'.\cap\cup x'$$

Finite representations of infinite pattern

$$\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap \cdots$$

represented by

$$\mu x.\cap\cup x$$

other representations of same pattern?

$$\mu x'.\cap\cup x'$$

$$\cap\mu y.\cup\cap y$$

Finite representations of infinite pattern

$$\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cdots$$

represented by

$$\mu x.\cap\cup x$$

other representations of same pattern?

$$\mu x'.\cap\cup x'$$

$$\cap\mu y.\cup\cap y$$

$$\cap\cup\mu z.\cap\cup z$$

Finite representations of infinite pattern

$$\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cdots$$

represented by

$$\mu x.\cap\cup x$$

other representations of same pattern?

$$\mu x'.\cap\cup x'$$

$$\cap\mu y.\cup\cap y$$

$$\cap\cup\mu z.\cap\cup z$$

$$\mu w.\cap\cup\cap\cup w$$

Finite representations of infinite pattern

$$\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cup\cap\cdots$$

represented by

$$\mu x.\cap\cup x$$

other representations of same pattern?

$$\mu x'.\cap\cup x'$$

$$\cap\mu y.\cup\cap y$$

$$\cap\cup\mu z.\cap\cup z$$

$$\mu w.\cap\cup\cap\cup w$$

when are two representations the same (finitely)?

Weak μ-equality

- Weak μ-equality on μ-terms:

$$=_{\mu} \;:=\; (\leftarrow_{\mu} \cup \rightarrow_{\mu})^{*}$$

(convertibility with respect to $\rightarrow_{\mu}$).

- Weak μ-equality on μ-pseudoterms:

$$=_{\mu/\alpha} \;:=\; (\leftarrow_{\mu/\alpha} \cup \rightarrow_{\mu/\alpha})^{*} \cup =_{\alpha}$$

(convertibility with respect to $\rightarrow_{\mu/\alpha} \;:=\; =_{\alpha} \cdot \rightarrow_{\mu} \cdot =_{\alpha}$).

Weak μ-equality

- Weak μ-equality on μ-terms:

$$=_\mu \ := \ (\leftarrow_\mu \cup \rightarrow_\mu)^*$$

(convertibility with respect to $\rightarrow_\mu$).

- Weak μ-equality on μ-pseudoterms:

$$=_{\mu/\alpha} \ := \ (\leftarrow_{\mu/\alpha} \cup \rightarrow_{\mu/\alpha})^* \cup =_\alpha$$

(convertibility with respect to $\rightarrow_{\mu/\alpha} \ := \ =_\alpha \cdot \rightarrow_\mu \cdot =_\alpha$).

Proposition

For all $M, N \in Ter(\mu)$ and $s, t \in PTer(\mu)$:

$$s =_{\mu/\alpha} t \iff [s] =_\mu [t]$$

μ-pseudoterms, μ-terms

Inductive definition of the set $PTer(\mu)$ of μ-pseudoterms:

(i) $x, y, z, \ldots \in PTer(\mu)$ (variables);

(ii) $\mathsf{c}, \mathsf{d}, \mathsf{e}, \ldots \in PTer(\mu)$ (constants);

(iii) $s, t \in PTer(\mu) \implies \mathsf{F}(s, t) \in PTer(\mu)$;

(iv) $s \in PTer(\mu)$ and x a variable $\implies \mu x.s \in PTer(\mu)$.

Notation:

- $s \to_\alpha t$ for α-*renaming*, and $s =_\alpha t$ for α-*equivalence* induced by α-conversion $=_\alpha := (\leftarrow_\alpha \cup \to_\alpha)^*$.
- $s[\![x := t]\!]$ for α-*converting substitution* à la Curry.

The set $Ter(\mu)$ of μ-terms consists of α-equivalence classes of μ-pseudoterms.

Deciding weak μ-equality by rewriting?

Deciding weak μ-equality by rewriting?

- μ-reduction $\mu x.s \to s[\![x := \mu x.s]\!]$ confluent but not terminating

$$\mu x.\mathsf{F}(c, x) \to \mathsf{F}(c, \mu x.\mathsf{F}(c, x)) \to \mathsf{F}(c, \mathsf{F}(c, \mu x.\mathsf{F}(c, x))) \to \ldots$$

Deciding weak μ-equality by rewriting?

- μ-reduction $\mu x.s \to s[\![x := \mu x.s]\!]$ confluent but not terminating

$$\mu x.\mathsf{F}(c, x) \to \mathsf{F}(c, \mu x.\mathsf{F}(c, x)) \to \mathsf{F}(c, \mathsf{F}(c, \mu x.\mathsf{F}(c, x))) \to \ldots$$

- μ-expansion $s[\![x := \mu x.s]\!] \to \mu x.s$ terminating but not confluent

$$\not\to \mathsf{F}(M, M) \to N \leftarrow \mu x.\mathsf{F}(M, \mathsf{F}(c, x)) \not\leftarrow$$

for

$$M = \mu y.\mathsf{F}(c, \mu x.\mathsf{F}(y, \mathsf{F}(c, x)))$$

$$N = \mathsf{F}(M, \mathsf{F}(c, \mu x.\mathsf{F}(M, \mathsf{F}(c, x))))$$

How to overcome?

Deciding weak μ-equality by rewriting!

μ-reduction non-terminating but active part repeats

$$\mu x.\mathsf{F}(c,x) \to \mathsf{F}(c,\mu x.\mathsf{F}(c,x)) \to \mathsf{F}(c,\mathsf{F}(c,\mu x.\mathsf{F}(c,x))) \to \ldots$$

Deciding weak μ-equality by rewriting!

μ-reduction non-terminating but active part repeats

$$\mu x.\mathsf{F}(c, x) \to \mathsf{F}(c, \mu x.\mathsf{F}(c, x)) \to \mathsf{F}(c, \mathsf{F}(c, \mu x.\mathsf{F}(c, x))) \to \ldots$$

Active part and repetition intuitions formalised in rest of talk

- Clemens: proof system
- Jörg: automata

Allows to bound the search space (loop checking).

Deciding weak μ-equality by rewriting!

μ-reduction non-terminating but active part repeats

$$\mu x.\mathsf{F}(c, x) \to \mathsf{F}(c, \mu x.\mathsf{F}(c, x)) \to \mathsf{F}(c, \mathsf{F}(c, \mu x.\mathsf{F}(c, x))) \to \ldots$$

Active part and repetition intuitions formalised in rest of talk

- Clemens: proof system
- Jörg: automata

Allows to bound the search space (loop checking).
Problem dealt with now: dealing with α-equivalence

$$\mu x.\mathsf{F}(c, x) \to \mathsf{F}(c, \mu y.\mathsf{F}(c, y)) \to \mathsf{F}(c, \mathsf{F}(c, \mu z.\mathsf{F}(c, z))) \to \ldots$$

Repetiton?

Overview

α-conversion unavoidable in λ-calculus

$$(\lambda w.ww)\lambda xy.xy$$

α-conversion unavoidable in λ-calculus

$$(\lambda w.ww)\lambda xy.xy$$

$$\to (\lambda xy.xy)\lambda xy.xy$$

α-conversion unavoidable in λ-calculus

$$(\lambda w.ww)\lambda xy.xy$$

$$\to (\lambda xy.xy)\lambda xy.xy$$

$$\to \lambda y.(\lambda xy.xy)y$$

α-conversion unavoidable in λ-calculus

$$(\lambda w.ww)\lambda xy.xy$$

$$\to (\lambda xy.xy)\lambda xy.xy$$

$$\to \lambda y.(\lambda xy.xy)y$$

$$\to \lambda y.(\lambda y.yy) \quad \text{wrong!}$$

α-conversion unavoidable in λ-calculus

$$(\lambda w.ww)\lambda xy.xy$$

$$\to (\lambda xy.xy)\lambda xy.xy$$

$$\to \lambda y.(\lambda xy.xy)y$$

$$\to \lambda y.(\lambda y.yy) \quad \text{wrong!}$$

- first step: non-linear (duplicating)
- second step: non-development (redex was created by first)
- third step: non-weak (redex below λ)

α-conversion can be avoided if one of these does hold.

Safe reduction

term is safe if α-free substitution $s[x := t]$ correct during reduction

Definition (α-free substitution)

- $x[x := t] = t$
- $y[x := t] = y$
- $(\mathsf{F}(s, s'))[x := t] = \mathsf{F}(s[x := t], s'[x := t])$
- $(\mu x.s)[x := t] = \mu x.s$
- $(\mu y.s)[x := t] = \mu y.s[x := t]$

Unsafe μ-terms

Is the following term safe?

$$\mu x.\mathsf{F}(y, \mu y.x)$$

Unsafe μ-terms

Is the following term safe?

$$\mu x.\mathsf{F}(y, \mu y.x)$$

No:

$$\to \mathsf{F}(y, \mu y.\mu x.\mathsf{F}(y, \mu y.x))$$

Unsafe μ-terms

Is the following term safe?

$$\mu x.\mathsf{F}(y, \mu y.x)$$

No:

$$\to \mathsf{F}(y, \mu y.\mu x.\mathsf{F}(y, \mu y.x))$$

but can be α-converted to safe μ-term

$$\mu x.\mathsf{F}(y, \mu z.x)$$

$$\to \mathsf{F}(y, \mu z.\mu x.\mathsf{F}(y, \mu z.x))$$

$$\to \dots$$

Unsafe μ-terms

Is the following term safe?

$$\mu x.\mathsf{F}(y, \mu y.x)$$

No:

$$\to \mathsf{F}(y, \mu y.\mu x.\mathsf{F}(y, \mu y.x))$$

but can be α-converted to safe μ-term

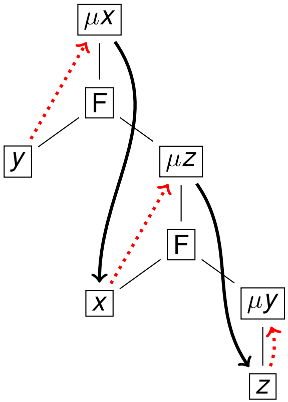
$$\mu x.\mathsf{F}(y, \mu z.x)$$

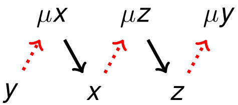
$$\to \mathsf{F}(y, \mu z.\mu x.\mathsf{F}(y, \mu z.x))$$

$$\to \dots$$

can this always be done?

Analysis of problem: self-capturing chains





A self-capturing chain of length 5 for the term $\mu x.\mathsf{F}(y, \mu z.\mathsf{F}(x, \mu y.z))$.

Self-capture-freeness guarantees safety

Definition

Term is self-capture-free if no self-capturing chains

Self-capture-freeness guarantees safety

Definition

Term is self-capture-free if no self-capturing chains

Theorem (Preservation of Self-capture-freeness)

If $s \to t$ and s self-capture-free then t self-capture-free.

Self-capture-freeness guarantees safety

Definition

Term is self-capture-free if no self-capturing chains

Theorem (Preservation of Self-capture-freeness)

If $s \to t$ and s self-capture-free then t self-capture-free.

Theorem (Self-catpure-free α-conversion)

Every term can be α-converted to a self-capture-free term.

Proof.

Choose all bound-variables distinct and distinct from free ones. □

Overview

Decision problem for weak μ-equality

We adress:

WEAK μ-EQUALITY PROBLEM *Instance:* μ-terms *M*, *N* *Question:* Does $M =_\mu N$ hold?

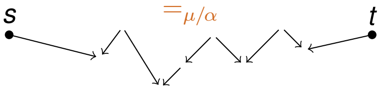
and its 'first-order' version:

WEAK μ-EQUALITY PROBLEM on μ-pseudoterms *Instance:* μ-pseudoterms *s*, *t* *Question:* Does $s =_{\mu/\alpha} t$ hold?

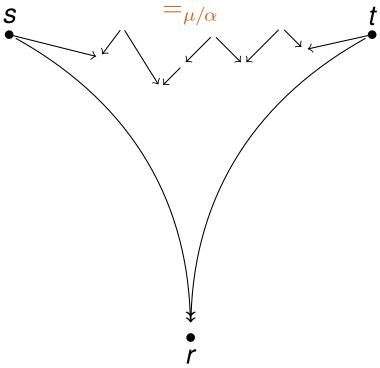
Structure of the first-order proof

$s \quad =_{\mu/\alpha} \quad t$

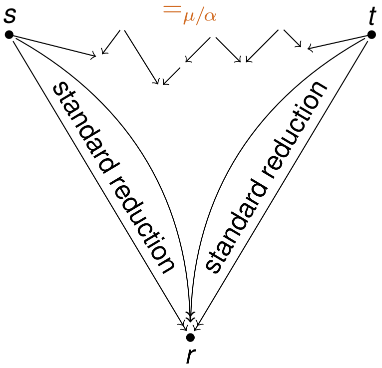
Structure of the first-order proof



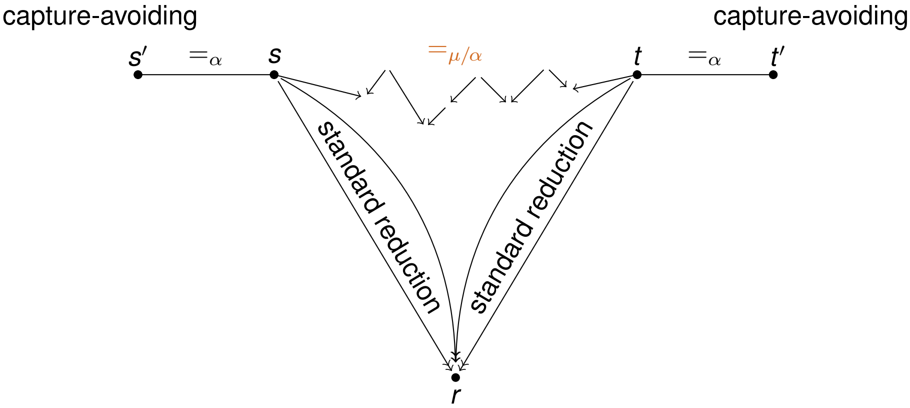
Structure of the first-order proof



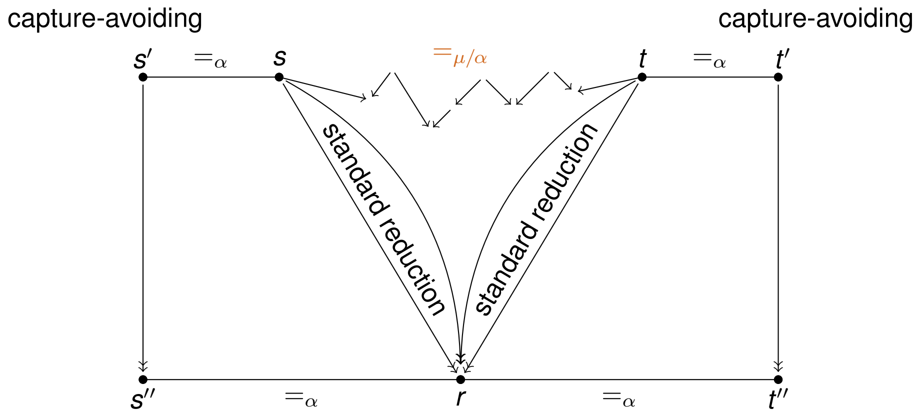
Structure of the first-order proof



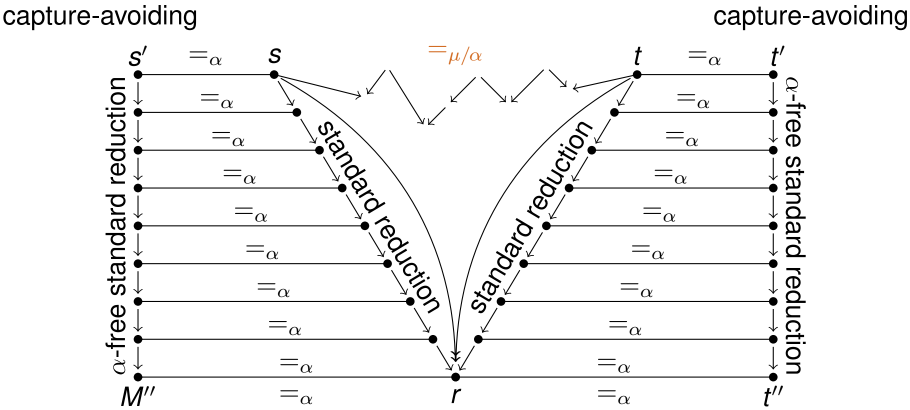
Structure of the first-order proof



Structure of the first-order proof



Structure of the first-order proof



Structure of the first-order proof

Thus:

- the weak μ-equality problem for μ-terms

can be reduced to:

- JOINABILITY PROBLEM UP TO $=_\alpha$ FOR $\to_\mu$ on capture-avoiding μ-pseudoterms

  Instance: capture avoiding μ-pseudoterms s, t
  Question: Are there s', t' with $s \twoheadrightarrow_{\text{std}} s' =_\alpha t' \twoheadleftarrow_{\text{std}} t$?

Structure of the first-order proof

Thus:

- the weak μ-equality problem for μ-terms

can be reduced to:

- JOINABILITY PROBLEM UP TO $=_\alpha$ FOR $\to_\mu$ on capture-avoiding μ-pseudoterms

  Instance: capture avoiding μ-pseudoterms s, t
  Question: Are there s', t' with $s \twoheadrightarrow_{\text{std}} s' =_\alpha t' \twoheadleftarrow_{\text{std}} t$?

Further proof strategy. Obtain a proof system $\mathcal{S}$ such that:

(1) $\mathcal{S}$ is complete for $\to_\mu$-joinability up to $=_\alpha$ on capture-avoiding μ-pseudoterms.

(2) the search-space for irredundant derivations in $\mathcal{S}$ is always finite.

Complete proof system (I) for $=_{\mu/\alpha}$ on μ-pseudoterms

$$\frac{(\mu\text{-unfolding})}{\mu x.s = s[x := \mu x.s]} \qquad \frac{(\alpha\text{-renaming})}{\mu x.s = \mu y.s[x := y]}$$

$$\frac{(\text{REFL})}{s = s} \qquad \frac{s = t}{t = s}\ \text{SYMM} \qquad \frac{s = r \quad r = t}{s = t}\ \text{TRANS}$$

$$\frac{s = t}{\mu x.s = \mu x.t}\ \mu\text{-COMPAT} \qquad \frac{s_1 = t_1 \quad s_2 = t_2}{\mathsf{F}(s_1, s_2) = \mathsf{F}(t_1, t_2)}\ \text{F-COMPAT}$$

- extension of a complete proof system for $=_\alpha$ (i.e. $\to_\alpha$-conversion)
- derivations correspond to $\to_{\mu/\alpha}$-conversions

Complete proof system (I) for $=_{\mu/\alpha}$ on μ-pseudoterms

$$\frac{(\mu\text{-unfolding})}{\mu x.s = s[x := \mu x.s]} \qquad \frac{(\alpha\text{-renaming})}{\mu x.s = \mu y.s[x := y]}$$

$$\frac{(\text{REFL})}{s = s} \qquad \frac{s = t}{t = s}\ \text{SYMM} \qquad \frac{s = r \quad r = t}{s = t}\ \text{TRANS}$$

$$\frac{s = t}{\mu x.s = \mu x.t}\ \mu\text{-COMPAT} \qquad \frac{s_1 = t_1 \quad s_2 = t_2}{\mathsf{F}(s_1, s_2) = \mathsf{F}(t_1, t_2)}\ \text{F-COMPAT}$$

- extension of a complete proof system for $=_\alpha$ (i.e. $\to_\alpha$-conversion)
- derivations correspond to $\to_{\mu/\alpha}$-conversions
- *Disadvantages:*
  - complex search space for proofs (**no** subformula property)
  - does not directly give rise to a decision method

Example

$$\mu x_3 x_2 x_1 . x_2 \ =_{\mu/\alpha} \ \mu yz.y$$

holds because of:

$$\mu x_3 x_2 x_1 . x_2 \to_\mu \mu x_2 x_1 . x_2 \to_\mu \mu x_2 . x_2 =_\alpha \mu y.y \leftarrow_\mu \mu yz.y$$

which gives rise to the derivation:

$$\text{TRANS}\ \dfrac{\text{TRANS}\ \dfrac{\dfrac{(\mu\text{-unfolding})}{\mu x_3 x_2 x_1 . x_2 = \mu x_2 x_1 . x_2} \quad \dfrac{\dfrac{(\mu\text{-unfolding})}{\mu x_1 . x_2 = x_2}}{\mu x_2 x_1 . x_2 = \mu x_2 . x_2}\ \mu}{\mu x_3 x_2 x_1 . x_2 = \mu x_2 . x_2} \quad \dfrac{(\alpha\text{-renaming})}{\mu x_2 . x_2 = \mu y.y}}{\mu x_3 x_2 x_1 . x_2 = \mu y.y} \quad \dfrac{\dfrac{\dfrac{(\mu\text{-unfolding})}{\mu z.y = y}}{\mu yz.y = \mu y.y}}{\mu y.y = \mu yz.y}}{\mu x_3 x_2 x_1 . x_2 = \mu yz.y}$$

Complete proof system (II) for $=_{\mu/\alpha}$ on μ-pseudoterms

$$\frac{}{s = s}\ \text{(if } s \text{ a variable or a constant)}$$

$$\frac{s[x := z] = t[y := z]}{\mu x.s = \mu y.t}\ \mu\ (z \text{ fresh}) \qquad \frac{s_1 = t_1 \quad s_2 = t_2}{\mathsf{F}(s_1, s_2) = \mathsf{F}(t_1, t_2)}\ \text{F-COMPAT}$$

$$\frac{s[\![x := \mu x.s]\!] = t}{\mu x.s = t}\ \text{FOLD}_l \qquad \frac{s = t[\![y := \mu y.t]\!]}{s = \mu y.t}\ \text{FOLD}_r$$

- extension of Schroer's characterisation of $\to_\alpha$-conversion
- derivations can be obtained by transitivity/symmetry-elimination in derivations of the previous system.
- derivations correspond to $\to_{\mu/\alpha}$-standard reductions

Complete proof system (II) for $=_{\mu/\alpha}$ on μ-pseudoterms

$$\frac{}{s = s}\ \text{(if } s \text{ a variable or a constant)}$$

$$\frac{s[x := z] = t[y := z]}{\mu x.s = \mu y.t}\ \mu\ (z \text{ fresh}) \qquad \frac{s_1 = t_1 \qquad s_2 = t_2}{\mathrm{F}(s_1, s_2) = \mathrm{F}(t_1, t_2)}\ \text{F-COMPAT}$$

$$\frac{s[\![x := \mu x.s]\!] = t}{\mu x.s = t}\ \text{FOLD}_l \qquad \frac{s = t[\![y := \mu y.t]\!]}{s = \mu y.t}\ \text{FOLD}_r$$

- extension of Schroer's characterisation of $\to_\alpha$-conversion
- derivations can be obtained by transitivity/symmetry-elimination in derivations of the previous system.
- derivations correspond to $\to_{\mu/\alpha}$-standard reductions
- *advantage:* (much more) restricted search space for derivations
- certain *disadvantage:* capture of free variables in μ-applications

Example

Proof System (II)

$$\cfrac{\cfrac{\cfrac{\cfrac{\overline{u = u}}{u = \mu z.u}\ \text{FOLD}_r}{\mu x_1.u = \mu z.u}\ \text{FOLD}_l}{\mu x_2 x_1.x_2 = \mu y z.y}\ \mu}{\mu x_3 x_2 x_1.x_2 = \mu y z.y}\ \text{FOLD}_l$$

Example

Proof System (II)

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{u = u}}{u = \mu z.u}\ \text{FOLD}_r}{\mu x_1.u = \mu z.u}\ \text{FOLD}_l}{\mu x_2 x_1.x_2 = \mu y z.y}\ \mu}{\mu x_3 x_2 x_1.x_2 = \mu y z.y}\ \text{FOLD}_l$$

Proof System (III)

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{x_2 = y \vdash x_2 = y}}{x_2 = y \vdash x_2 = \mu z.y}\ \text{FOLD}_r}{x_2 = y \vdash \mu x_1.x_2 = \mu z.y}\ \text{FOLD}_l}{\vdash \mu x_2 x_1.x_2 = \mu y z.y}\ \mu}{\vdash \mu x_3 x_2 x_1.x_2 = \mu y z.y}\ \text{FOLD}_l$$

Complete proof system (III) for $=_{\mu/\alpha}$ on μ-pseudoterms

$$\frac{}{x = y \vdash x = y}$$

$$\frac{\text{(restr-REFL)}}{\vdash s = s} \quad \text{(if } s \text{ a variable or a constant)}$$

$$\frac{\Gamma\,,\,\vec{z} = \vec{u} \vdash s = t}{\Gamma\,,\,x = y\,,\,\vec{z} = \vec{u} \vdash s = t}\ \text{COMPR} \quad (\text{if } x \notin \mathrm{FV}(\mu\vec{z}.s) \text{ and } y \notin \mathrm{FV}(\mu\vec{u}.t))$$

$$\frac{\Gamma\,,\,x = y \vdash s = t}{\Gamma \vdash \mu x.s = \mu y.t}\ \mu \qquad \frac{\Gamma \vdash s_1 = t_1 \qquad \Gamma \vdash s_2 = t_2}{\Gamma \vdash \mathrm{F}(s_1, s_2) = \mathrm{F}(t_1, t_2)}\ \mathrm{F}$$

$$\frac{\Gamma \vdash s[\![x := \mu x.s]\!] = t}{\Gamma \vdash \mu x.s = t}\ \mathrm{FOLD}_l \qquad \frac{\Gamma \vdash s = t[\![y := \mu y.t]\!]}{\Gamma \vdash s = \mu y.t}\ \mathrm{FOLD}_r$$

- extension of Kahrs' characterisation of α-conversion
- der's obtainable by trans./symm.-elim. from der's in system (I)
- derivations correspond to $\rightarrow_{\mu/\alpha}$-standard reductions

Complete proof system (III) for $=_{\mu/\alpha}$ on μ-pseudoterms

$$\frac{}{x = y \vdash x = y}$$

$$\frac{}{\vdash s = s}\ \text{(restr-REFL)} \quad \text{(if } s \text{ a variable or a constant)}$$

$$\frac{\Gamma\,,\, \vec{z} = \vec{u} \vdash s = t}{\Gamma\,,\, x = y\,,\, \vec{z} = \vec{u} \vdash s = t}\ \text{COMPR} \quad (\text{if } x \notin \text{FV}(\mu\vec{z}.s) \text{ and } y \notin \text{FV}(\mu\vec{u}.t))$$

$$\frac{\Gamma\,,\, x = y \vdash s = t}{\Gamma \vdash \mu x.s = \mu y.t}\ \mu \qquad \frac{\Gamma \vdash s_1 = t_1 \qquad \Gamma \vdash s_2 = t_2}{\Gamma \vdash \text{F}(s_1, s_2) = \text{F}(t_1, t_2)}\ \text{F}$$

$$\frac{\Gamma \vdash s[\![x := \mu x.s]\!] = t}{\Gamma \vdash \mu x.s = t}\ \text{FOLD}_l \qquad \frac{\Gamma \vdash s = t[\![y := \mu y.t]\!]}{\Gamma \vdash s = \mu y.t}\ \text{FOLD}_r$$

- extension of Kahrs' characterisation of α-conversion
- der's obtainable by trans./symm.-elim. from der's in system (I)
- derivations correspond to $\rightarrow_{\mu/\alpha}$-standard reductions
- *advantage:* restricted search space for derivations

Complete proof system (III) for $=_{\mu/\alpha}$ on μ-pseudoterms

$$\frac{x = y \vdash x = y}{}$$

$$\frac{\vdash s = s}{\text{(restr-REFL)}} \quad \text{(if } s \text{ a variable or a constant)}$$

$$\frac{\Gamma\,,\, x = y\,,\, \vec{z} = \vec{u} \vdash s = t}{\Gamma\,,\, \vec{z} = \vec{u} \vdash s = t}\ \text{COMPR} \quad (\text{if } x \notin \mathrm{FV}(\mu\vec{z}.s) \text{ and } y \notin \mathrm{FV}(\mu\vec{u}.t))$$

$$\frac{\Gamma \vdash \mu x.s = \mu y.t}{\Gamma\,,\, x = y \vdash s = t}\ \mu \qquad \frac{\Gamma \vdash \mathrm{F}(s_1, s_2) = \mathrm{F}(t_1, t_2)}{\Gamma \vdash s_1 = t_1 \qquad \Gamma \vdash s_2 = t_2}\ \mathrm{F}$$

$$\frac{\Gamma \vdash \mu x.s = t}{\Gamma \vdash s[\![x := \mu x.s]\!] = t}\ \mathrm{UNFOLD}_l \qquad \frac{\Gamma \vdash s = \mu y.t}{\Gamma \vdash s = t[\![y := \mu y.t]\!]}\ \mathrm{UNFOLD}_r$$

- extension of Kahrs' characterisation of α-conversion
- der's obtainable by trans./symm.-elim. from der's in system (I)
- derivations correspond to $\to_{\mu/\alpha}$-standard reductions
- *advantage:* restricted search space for derivations

Complete proof system (III) for $=_{\mu/\alpha}$ on μ-pseudoterms

$$\frac{}{(\mu x)x = (\mu y)y} \qquad \frac{()s = ()s}{\text{(restr-REFL)}} \text{ (if } s \text{ a variable or a constant)}$$

$$\frac{(\mu \vec{z}_1 x \vec{z}_2)s = (\mu \vec{u}_1 y \vec{u}_2)t}{(\mu \vec{z}_1 \vec{z}_2)s = (\mu \vec{u}_1)\vec{u}_2 t} \text{ COMPR} \quad \text{(if } |\vec{z}_2| = |\vec{u}_2|,\ x \notin \mathrm{FV}(\mu\vec{z}_2.s) \text{ and } y \notin \mathrm{FV}(\mu\vec{u}_2.t))$$

$$\frac{(\mu\vec{z})\mu x.s = (\mu\vec{u})\mu y.t}{(\mu\vec{z}x)s = (\mu\vec{u}y)t}\ \mu \qquad \frac{(\mu\vec{z})\mathrm{F}(s_1, s_2) = (\mu\vec{u})\mathrm{F}(t_1, t_2)}{(\mu\vec{z})s_1 = (\mu\vec{u})t_1 \qquad (\mu\vec{z})s_2 = (\mu\vec{u})t_2}\ \mathrm{F}$$

$$\frac{(\mu\vec{z})\mu x.s = (\mu\vec{u})t}{(\mu\vec{z})s[\![x := \mu x.s]\!] = (\mu\vec{u})t}\ \mathrm{UNF}_l \qquad \frac{(\mu\vec{z})s = (\mu\vec{u})\mu y.t}{(\mu\vec{z})s = (\mu\vec{u})t[\![y := \mu y.t]\!]}\ \mathrm{UNF}_r$$

- extension of Kahrs' characterisation of α-conversion
- der's obtainable by trans./symm.-elim. from der's in system (I)
- derivations correspond to $\to_{\mu/\alpha}$-standard reductions
- *advantage:* restricted search space for derivations

Example

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{x_2 = y \vdash x_2 = y}}{x_2 = y \vdash x_2 = \mu z.y}\ \text{UNFOLD}_r}{x_2 = y \vdash \mu x_1.x_2 = \mu z.y}\ \text{UNFOLD}_l}{\vdash \mu x_2 x_1.x_2 = \mu yz.y}\ \mu}{\vdash \mu x_3 x_2 x_1.x_2 = \mu yz.y}\ \text{UNFOLD}_l
$$

Example

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{(\mu x_2)x_2 = (\mu y)y}}{(\mu x_2)x_2 = (\mu y)\mu z.y}\ \text{UNFOLD}_r}{(\mu x_2)\mu x_1.x_2 = (\mu y)\mu z.y}\ \text{UNFOLD}_l}{()\mu x_2 x_1.x_2 = ()\mu yz.y}\ \mu}{()\mu x_3 x_2 x_1.x_2 = ()\mu yz.y}\ \text{UNFOLD}_l$$

Example

$$\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{(\,)\mu x_3 x_2 x_1.x_2 = (\,)\mu yz.y}{(\,)\mu x_2 x_1.x_2 = (\,)\mu yz.y}\ \text{UNFOLD}_l}{(\mu x_2)\mu x_1.x_2 = (\mu y)\mu z.y}\ \mu}{(\mu x_2)x_2 = (\mu y)\mu z.y}\ \text{UNFOLD}_l}{(\mu x_2)x_2 = (\mu y)y}\ \text{UNFOLD}_r}{}$$

Extraction of reductions:

$$\begin{aligned}\mu x_3 x_2 x_1.x_2 \;\to_\mu\; \mu x_2 x_1.x_2 \;\triangleright_{\text{frz}}\; (\mu x_2)\mu x_1.x_2 \;\to_\mu\; (\mu x_2)x_2 \\ =_\alpha\; (\mu y)y \;\leftarrow_\mu\; (\mu y)\mu z.y \;\triangleleft_{\text{frz}}\; \mu yz.y\end{aligned}$$

Example

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{()\mu x_3 x_2 x_1.x_2 = ()\mu yz.y}{()\mu x_2 x_1.x_2 = ()\mu yz.y}\,\text{UNFOLD}_l}{(\mu x_2)\mu x_1.x_2 = (\mu y)\mu z.y}\,\mu}{(\mu x_2)x_2 = (\mu y)\mu z.y}\,\text{UNFOLD}_l}{(\mu x_2)x_2 = (\mu y)y}\,\text{UNFOLD}_r}{}
$$

Extraction of reductions:

$$
\begin{aligned}
\mu x_3 x_2 x_1.x_2 \;\to_\mu\; \mu x_2 x_1.x_2 \;\rhd_{\text{frz}}\; (\mu x_2)\mu x_1.x_2 \;\to_\mu\; (\mu x_2)x_2 \\
=_\alpha\; (\mu y)y \;\leftarrow_\mu\; (\mu y)\mu z.y \;\lhd_{\text{frz}}\; \mu yz.y
\end{aligned}
$$

gives a joining pair of standard reductions:

$$
\mu x_3 x_2 x_1.x_2 \;\to_\mu\; \mu x_2 x_1.x_2 \;\to_\mu\; \mu x_2.x_2 \;=_\alpha\; \mu y.y \;\leftarrow_\mu\; \mu yz.y
$$

Subterm closure

The $\mu\pi$-calculus on μ-pseudoterms:

$$\begin{aligned} \mathsf{F}(s_1, s_2) &\to s_i \quad \text{for } i \in \{1,2\} && \text{(F-projection)} \\ \mu x.s &\to s && (\mu\text{-projection}) \\ \mu x.s &\to s[x := \mu x.s] && (\mu\text{-reduction}) \end{aligned}$$

By $\to^{\varepsilon}_{\mu\pi}$ we denote $\mu\pi$-root-reduction.

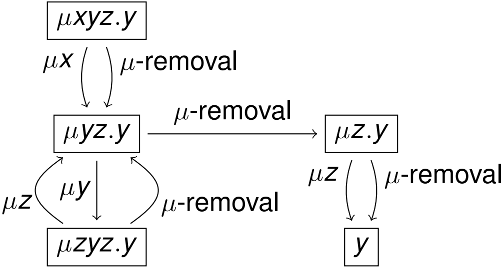
The subterm closure $\mathsf{SC}(s)$ of a capture-avoiding $s \in PTer(\mu)$ is:

$$\mathsf{SC}(s) := \{t \in PTer(\mu) \mid s \twoheadrightarrow^{\varepsilon}_{\mu\pi} t\} \ .$$

Theorem

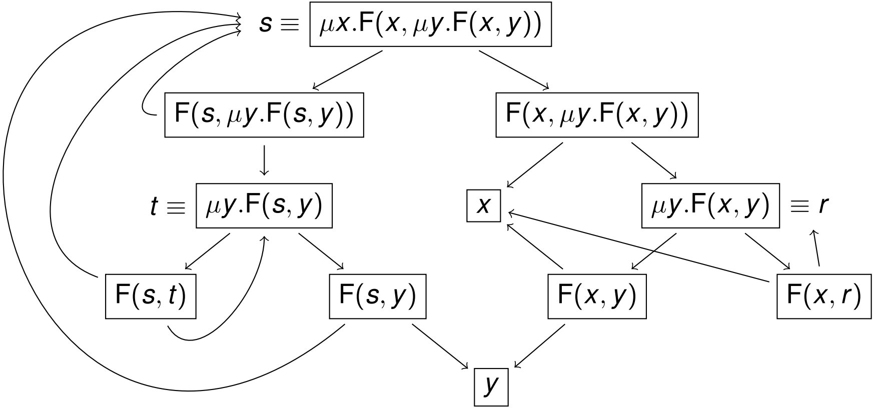
For all capture-avoiding $s \in PTer(\mu)$, $\mathsf{SC}(s)$ is finite.

Subterm closure



The subterm closure of $\mu xyz.y$.

Subterm closure



The subterm closure of $\mu x.\mathsf{F}(x, \mu y.\mathsf{F}(x, y))$.

Decidability of $=_{\mu/\alpha}$ by a first-order proof

Lemma

Provability in system (III) of formulas $\vdash s = t$, where $s, t \in PTer(\mu)$ are capture-avoiding, is decidable.

Proof.

- subformula property: for an equation $(\mu \ldots)s' = (\mu \ldots)t'$ in a derivation $\mathcal{D}$ with conclusion $()s = ()t$ it holds that $s' \in \mathrm{SC}(s)$ and $t' \in \mathrm{SC}(t)$.

Decidability of $=_{\mu/\alpha}$ by a first-order proof

Lemma

Provability in system (III) of formulas $\vdash s = t$, where $s, t \in PTer(\mu)$ are capture-avoiding, is decidable.

Proof.

- subformula property: for an equation $(\mu \ldots)s' = (\mu \ldots)t'$ in a derivation $\mathcal{D}$ with conclusion $()s = ()t$ it holds that $s' \in \mathrm{SC}(s)$ and $t' \in \mathrm{SC}(t)$.
- bound L on annotation lengths: if rule COMPR is applied 'greedily', $L :=$ no. of binder occurr's in ps.terms in conclusion.

Decidability of $=_{\mu/\alpha}$ by a first-order proof

Lemma

Provability in system (III) of formulas $\vdash s = t$, where $s, t \in PTer(\mu)$ are capture-avoiding, is decidable.

Proof.

- subformula property: for an equation $(\mu \ldots)s' = (\mu \ldots)t'$ in a derivation $\mathcal{D}$ with conclusion $()s = ()t$ it holds that $s' \in \mathrm{SC}(s)$ and $t' \in \mathrm{SC}(t)$.
- bound L on annotation lengths: if rule COMPR is applied 'greedily', $L :=$ no. of binder occurr's in ps.terms in conclusion.
- bound on the size of irredundant derivations: as a consequence, the size of an irredundant derivation (no formula repetitions) with conclusion $s = t$ is bounded by $|\pi(L)| \cdot |\mathrm{SC}(s)| \cdot |\mathrm{SC}(t)|$.

Decidability of $=_{\mu/\alpha}$ by a first-order proof

Lemma

Provability in system (III) of formulas $\vdash s = t$, where $s, t \in PTer(\mu)$ are capture-avoiding, is decidable.

Proof.

- subformula property: for an equation $(\mu \ldots)s' = (\mu \ldots)t'$ in a derivation $\mathcal{D}$ with conclusion $()s = ()t$ it holds that $s' \in \mathrm{SC}(s)$ and $t' \in \mathrm{SC}(t)$.
- bound L on annotation lengths: if rule COMPR is applied 'greedily', $L :=$ no. of binder occurr's in ps.terms in conclusion.
- bound on the size of irredundant derivations: as a consequence, the size of an irredundant derivation (no formula repetitions) with conclusion $s = t$ is bounded by $|\pi(L)| \cdot |\mathrm{SC}(s)| \cdot |\mathrm{SC}(t)|$.

Theorem

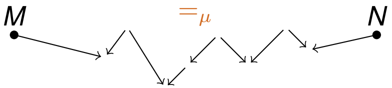
Weak μ-equality is decidable.

Overview

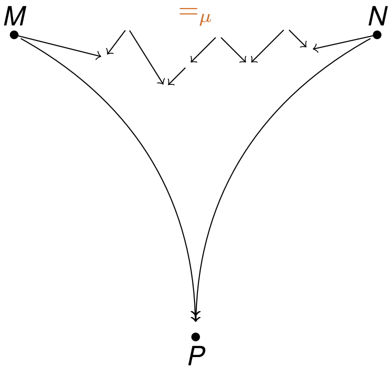
1. Weak μ-equality
2. Avoiding α-conversion in μ-reductions
3. Decidability of $=_{\mu/\alpha}$ by a first-order proof
4. Decidability of $=_{\mu/\alpha}$ by a higher-order proof
5. Decidability of $=_{\mu/\alpha}$ using regular languages
6. Summary

Structure of the higher-order proof

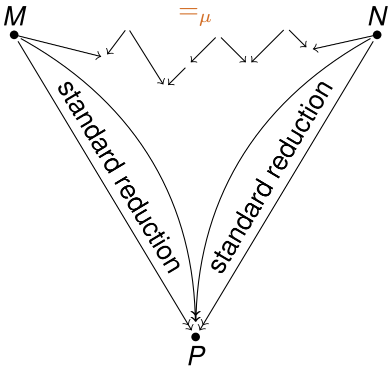
Structure of the higher-order proof



Structure of the higher-order proof



Structure of the higher-order proof



Complete proof system (II) for $=_{\mu/\alpha}$ on μ-pseudoterms

$$\frac{}{s = s}\ \text{(if } s \text{ a variable or a constant)}$$

$$\frac{s[x := z] = t[y := z]}{\mu x.s = \mu y.t}\ \mu\ (z \text{ fresh}) \qquad \frac{s_1 = t_1 \qquad s_2 = t_2}{\mathrm{F}(s_1, s_2) = \mathrm{F}(t_1, t_2)}\ \text{F-COMPAT}$$

$$\frac{s[\![x := \mu x.s]\!] = t}{\mu x.s = t}\ \mathrm{FOLD}_l \qquad \frac{s = t[\![y := \mu y.t]\!]}{s = \mu y.t}\ \mathrm{FOLD}_r$$

- extension of Schroer's characterisation of $\rightarrow_\alpha$-conversion
- derivations correspond to $\rightarrow_{\mu/\alpha}$-standard reductions
- *advantage:* restricted search space for derivations
- *disadvantage:* capture of free variables in μ-applications

Complete proof system for $=_\mu$ on μ-terms

$$\frac{}{s = s} \ \text{(if } s \text{ a variable or a constant)}$$

$$\frac{s[x := \underline{n}] = t[y := \underline{n}]}{\mu x.s = \mu y.t} \ \mu \ (\underline{n} \text{ fresh numeral}) \qquad \frac{s_1 = t_1 \qquad s_2 = t_2}{\mathrm{F}(s_1, s_2) = \mathrm{F}(t_1, t_2)} \ \text{F-COMPAT}$$

$$\frac{s[\![x := \mu x.s]\!] = t}{\mu x.s = t} \ \text{FOLD}_l \qquad \frac{s = t[\![y := \mu y.t]\!]}{s = \mu y.t} \ \text{FOLD}_r$$

- extension of Schroer's characterisation of $\rightarrow_\alpha$-conversion
- derivations correspond to $\rightarrow_{\mu/\alpha}$-standard reductions
- *advantage:* restricted search space for derivations
- a certain subformula property

Overview

1. Weak μ-equality
2. Avoiding α-conversion in μ-reductions
3. Decidability of $=_{\mu/\alpha}$ by a first-order proof
4. Decidability of $=_{\mu/\alpha}$ by a higher-order proof
5. Decidability of $=_{\mu/\alpha}$ using regular languages
6. Summary

Decidability of $=_{\mu/\alpha}$ using regular languages

An alternative approach: using regular languages.

Decidability of $=_{\mu/\alpha}$ using regular languages

An alternative approach: using regular languages.

- For a capture-avoiding *M* we construct a regular grammar $\mathcal{G}_M$ generating the set of reducts of *M* (without α-conversion).

Decidability of $=_{\mu/\alpha}$ using regular languages

An alternative approach: using regular languages.

- For a capture-avoiding *M* we construct a regular grammar $\mathcal{G}_M$ generating the set of reducts of *M* (without α-conversion).
- Given a regular grammar $\mathcal{G}$, we construct a grammar $\mathcal{G}^\alpha$ generating the closure of $\mathcal{G}$ under α-conversion.

Decidability of $=_{\mu/\alpha}$ using regular languages

An alternative approach: using regular languages.

- For a capture-avoiding M we construct a regular grammar $\mathcal{G}_M$ generating the set of reducts of M (without α-conversion).
- Given a regular grammar $\mathcal{G}$, we construct a grammar $\mathcal{G}^\alpha$ generating the closure of $\mathcal{G}$ under α-conversion.
- Then $M =_\alpha N$ boils down to the question:

$$\mathcal{L}(\mathcal{G}_M^\alpha) \cap \mathcal{L}(\mathcal{G}_N^\alpha) \neq \varnothing \text{ ?}$$

Decidability of $=_{\mu/\alpha}$ using regular languages

An alternative approach: using regular languages.

- For a capture-avoiding *M* we construct a regular grammar $\mathcal{G}_M$ generating the set of reducts of *M* (without α-conversion).
- Given a regular grammar $\mathcal{G}$, we construct a grammar $\mathcal{G}^\alpha$ generating the closure of $\mathcal{G}$ under α-conversion.
- Then $M =_\alpha N$ boils down to the question:

$$\mathcal{L}(\mathcal{G}_M^\alpha) \cap \mathcal{L}(\mathcal{G}_N^\alpha) \neq \varnothing \text{ ?}$$

This problem is known to be decidable.

Step 1: a regular grammar for μ-reducts

Let $M \in Ter(\mu)$ be a capture-avoiding μ-pseudoterm.

Step 1: a regular grammar for μ-reducts

Let $M \in \mathit{Ter}(\mu)$ be a capture-avoiding μ-pseudoterm.

We construct a regular grammar $\mathcal{G}_M$ for the μ-reducts of M:

The start symbol of $\mathcal{G}_M$ is V_M, and the rules are:

$$\mathsf{V}_{\mu x.N} \Rightarrow \mathsf{V}_{N[x:=\mu x.N]} \tag{1}$$
$$\mathsf{V}_{\mu x.N} \Rightarrow \mu x.\mathsf{V}_N \tag{2}$$
$$\mathsf{V}_{\mathsf{F}(N,N')} \Rightarrow \mathsf{F}(\mathsf{V}_N, \mathsf{V}_{N'}) \tag{3}$$
$$\mathsf{V}_x \Rightarrow x \tag{4}$$

for every V_s such that $s \in \mathrm{SC}(M)$.

Step 1: a regular grammar for μ-reducts

Let $M \in \mathit{Ter}(\mu)$ be a capture-avoiding μ-pseudoterm.

We construct a regular grammar $\mathcal{G}_M$ for the μ-reducts of M:

The start symbol of $\mathcal{G}_M$ is V_M, and the rules are:

$$\begin{aligned} \mathsf{V}_{\mu x.N} &\Rightarrow \mathsf{V}_{N[x:=\mu x.N]} && (1)\\ \mathsf{V}_{\mu x.N} &\Rightarrow \mu x.\mathsf{V}_N && (2)\\ \mathsf{V}_{\mathsf{F}(N,N')} &\Rightarrow \mathsf{F}(\mathsf{V}_N, \mathsf{V}_{N'}) && (3)\\ \mathsf{V}_x &\Rightarrow x && (4) \end{aligned}$$

for every V_s such that $s \in \mathrm{SC}(M)$.

Lemma

$$\mathcal{L}(\mathcal{G}_M) = \{N \mid M \to^* N\}$$

where $\to$ is α-conversion free μ-reduction.

Step 1: a regular grammar for μ-reducts

Example

Let $M \equiv \mu y.\mathsf{F}(x, y)$, then $\mathcal{G}_M$ consists of:

$$
\begin{aligned}
\mathsf{V}_{\mu y.\mathsf{F}(x,y)} &\Rightarrow_{(1)} \mathsf{V}_{\mathsf{F}(x,\mu y.\mathsf{F}(x,y))} \\
\mathsf{V}_{\mu y.\mathsf{F}(x,y)} &\Rightarrow_{(2)} \mu y.\mathsf{V}_{\mathsf{F}(x,y)} \\
\mathsf{V}_{\mathsf{F}(x,y)} &\Rightarrow_{(3)} \mathsf{F}(\mathsf{V}_x, \mathsf{V}_y) \\
\mathsf{V}_{\mathsf{F}(x,\mu y.\mathsf{F}(x,y))} &\Rightarrow_{(3)} \mathsf{F}(\mathsf{V}_x, \mathsf{V}_{\mu y.\mathsf{F}(x,y)}) \\
\mathsf{V}_x &\Rightarrow_{(4)} x \\
\mathsf{V}_y &\Rightarrow_{(4)} y
\end{aligned}
$$

The start symbol of $\mathcal{G}_M$ is $\mathsf{V}_{\mu y.\mathsf{F}(x,y)}$.

Step 2: α-conversion

Let $\mathcal{G}$ be normalised with start variable V over a finite set of binder $\mathbb{B}$.

We define a grammar accepting all α-equivalent terms over $\mathbb{B}$:

Step 2: α-conversion

Let $\mathcal{G}$ be normalised with start variable V over a finite set of binder $\mathbb{B}$.

We define a grammar accepting all α-equivalent terms over $\mathbb{B}$:

Let $\mathcal{G}^\alpha$ have start variable $\mathsf{V}_{id,\varnothing}$, and for all:

- $\sigma : \mathbb{B} \to \mathbb{B}$ (renaming map),
- $\dagger \subseteq \mathbb{B}$ (forbidden variables),

consist of rules:

- $\mathsf{V}_{\sigma,\dagger} \Rightarrow \sigma(x) \in \mathcal{G}^\alpha$ (renaming) if $\mathsf{V} \Rightarrow x \in G$ and $x \notin \dagger$
- $\mathsf{V}_{\sigma,\dagger} \Rightarrow \bot \in \mathcal{G}^\alpha$ (name clash) if $\mathsf{V} \Rightarrow x \in G$ and $x \in \dagger$
- $\mathsf{V}_{\sigma,\dagger} \Rightarrow \mathsf{F}(\mathsf{V}'_{\sigma,\dagger}, \mathsf{V}''_{\sigma,\dagger}) \in \mathcal{G}^\alpha$ (propagation) if $\mathsf{V} \Rightarrow \mathsf{F}(\mathsf{V}', \mathsf{V}'') \in G$
- $\mathsf{V}_{\sigma,\dagger} \Rightarrow \mu y(\mathsf{V}'_{\sigma',\dagger'}) \in G$ (pick renaming) if $\mathsf{V} \Rightarrow \mu x(\mathsf{V}') \in G$

  where $y \in \mathbb{B}$, $\sigma' = \sigma[x \mapsto y]$, $\dagger' = (\dagger \cup \sigma^{-1}(y)) \setminus \{x\}$.

Step 2: α-conversion

pick renaming: $\mathsf{V}_{\sigma,\dagger} \Rightarrow \mu y(\mathsf{V}'_{\sigma',\dagger'}) \in G$ if $\mathsf{V} \Rightarrow \mu x(\mathsf{V}') \in G$
where $y \in \mathbb{B}$, $\sigma' = \sigma[x \mapsto y]$, $\dagger' = (\dagger \cup \sigma^{-1}(y)) \setminus \{x\}$.

Example

Let G have start variable V_1 and consist of the rules:

$$\mathsf{V}_1 \Rightarrow \mu x.\mathsf{V}_2 \quad \mathsf{V}_2 \Rightarrow \mu y.\mathsf{V}_3 \quad \mathsf{V}_3 \Rightarrow \mathsf{F}(\mathsf{V}_4, \mathsf{V}_5) \quad \mathsf{V}_4 \Rightarrow x \quad \mathsf{V}_5 \Rightarrow y$$

Step 2: α-conversion

pick renaming: $\mathsf{V}_{\sigma,\dagger} \Rightarrow \mu y(\mathsf{V}'_{\sigma',\dagger'}) \in G$ if $\mathsf{V} \Rightarrow \mu x(\mathsf{V}') \in G$
where $y \in \mathbb{B}$, $\sigma' = \sigma[x \mapsto y]$, $\dagger' = (\dagger \cup \sigma^{-1}(y)) \setminus \{x\}$.

Example

Let G have start variable V_1 and consist of the rules:

$$\mathsf{V}_1 \Rightarrow \mu x.\mathsf{V}_2 \quad \mathsf{V}_2 \Rightarrow \mu y.\mathsf{V}_3 \quad \mathsf{V}_3 \Rightarrow \mathsf{F}(\mathsf{V}_4, \mathsf{V}_5) \quad \mathsf{V}_4 \Rightarrow x \quad \mathsf{V}_5 \Rightarrow y$$

Note that G generates the term $\mu x.\mu y.\mathsf{F}(x, y)$.

Step 2: α-conversion

pick renaming: $V_{\sigma,\dagger} \Rightarrow \mu y(V'_{\sigma',\dagger'}) \in G$ if $V \Rightarrow \mu x(V') \in G$
where $y \in \mathbb{B}$, $\sigma' = \sigma[x \mapsto y]$, $\dagger' = (\dagger \cup \sigma^{-1}(y)) \setminus \{x\}$.

Example

Let G have start variable V_1 and consist of the rules:

$$V_1 \Rightarrow \mu x.V_2 \quad V_2 \Rightarrow \mu y.V_3 \quad V_3 \Rightarrow F(V_4, V_5) \quad V_4 \Rightarrow x \quad V_5 \Rightarrow y$$

Note that G generates the term $\mu x.\mu y.F(x, y)$.

Let $\mathcal{G}^\alpha$ has start variable $V_{1,\{x\mapsto x, y\mapsto y\},\varnothing}$, and contains rules:

Step 2: α-conversion

pick renaming: $\mathsf{V}_{\sigma,\dagger} \Rightarrow \mu y(\mathsf{V}'_{\sigma',\dagger'}) \in G$ if $\mathsf{V} \Rightarrow \mu x(\mathsf{V}') \in G$
where $y \in \mathbb{B}$, $\sigma' = \sigma[x \mapsto y]$, $\dagger' = (\dagger \cup \sigma^{-1}(y)) \setminus \{x\}$.

Example

Let G have start variable V_1 and consist of the rules:

$$\mathsf{V}_1 \Rightarrow \mu x.\mathsf{V}_2 \quad \mathsf{V}_2 \Rightarrow \mu y.\mathsf{V}_3 \quad \mathsf{V}_3 \Rightarrow \mathsf{F}(\mathsf{V}_4, \mathsf{V}_5) \quad \mathsf{V}_4 \Rightarrow x \quad \mathsf{V}_5 \Rightarrow y$$

Note that G generates the term $\mu x.\mu y.\mathsf{F}(x, y)$.

Let $\mathcal{G}^\alpha$ has start variable $\mathsf{V}_{1,\{x\mapsto x,y\mapsto y\},\varnothing}$, and contains rules:

$$\mathsf{V}_{1,\{x\mapsto x,y\mapsto y\},\varnothing} \Rightarrow \mu y.\mathsf{V}_{2,\{x\mapsto y,y\mapsto y\},\{y\}}$$

Step 2: α-conversion

pick renaming: $V_{\sigma,\dagger} \Rightarrow \mu y(V'_{\sigma',\dagger'}) \in G$ if $V \Rightarrow \mu x(V') \in G$
where $y \in \mathbb{B}$, $\sigma' = \sigma[x \mapsto y]$, $\dagger' = (\dagger \cup \sigma^{-1}(y)) \setminus \{x\}$.

Example

Let G have start variable V_1 and consist of the rules:

$$V_1 \Rightarrow \mu x.V_2 \quad V_2 \Rightarrow \mu y.V_3 \quad V_3 \Rightarrow F(V_4, V_5) \quad V_4 \Rightarrow x \quad V_5 \Rightarrow y$$

Note that G generates the term $\mu x.\mu y.F(x, y)$.

Let $\mathcal{G}^\alpha$ has start variable $V_{1,\{x\mapsto x,y\mapsto y\},\varnothing}$, and contains rules:

$$V_{1,\{x\mapsto x,y\mapsto y\},\varnothing} \Rightarrow \mu y.V_{2,\{x\mapsto y,y\mapsto y\},\{y\}}$$
$$V_{2,\{x\mapsto y,y\mapsto y\},\{y\}} \Rightarrow \mu y.V_{3,\{x\mapsto y,y\mapsto y\},\{x\}}$$

Step 2: α-conversion

pick renaming: $V_{\sigma,\dagger} \Rightarrow \mu y(V'_{\sigma',\dagger'}) \in G$ if $V \Rightarrow \mu x(V') \in G$
where $y \in \mathbb{B}$, $\sigma' = \sigma[x \mapsto y]$, $\dagger' = (\dagger \cup \sigma^{-1}(y)) \setminus \{x\}$.

Example

Let G have start variable V_1 and consist of the rules:

$$V_1 \Rightarrow \mu x.V_2 \quad V_2 \Rightarrow \mu y.V_3 \quad V_3 \Rightarrow F(V_4, V_5) \quad V_4 \Rightarrow x \quad V_5 \Rightarrow y$$

Note that G generates the term $\mu x.\mu y.F(x, y)$.

Let $\mathcal{G}^\alpha$ has start variable $V_{1,\{x\mapsto x, y\mapsto y\},\varnothing}$, and contains rules:

$$V_{1,\{x\mapsto x,y\mapsto y\},\varnothing} \Rightarrow \mu y.V_{2,\{x\mapsto y,y\mapsto y\},\{y\}}$$
$$V_{2,\{x\mapsto y,y\mapsto y\},\{y\}} \Rightarrow \mu y.V_{3,\{x\mapsto y,y\mapsto y\},\{x\}}$$
$$V_{3,\{x\mapsto y,y\mapsto y\},\{x\}} \Rightarrow F(V_{4,\{x\mapsto y,y\mapsto y\},\{x\}}, V_{5,\{x\mapsto y,y\mapsto y\},\{x\}})$$

Step 2: α-conversion

pick renaming: $V_{\sigma,\dagger} \Rightarrow \mu y(V'_{\sigma',\dagger'}) \in G$ if $V \Rightarrow \mu x(V') \in G$
where $y \in \mathbb{B}$, $\sigma' = \sigma[x \mapsto y]$, $\dagger' = (\dagger \cup \sigma^{-1}(y)) \setminus \{x\}$.

Example

Let G have start variable V_1 and consist of the rules:

$$V_1 \Rightarrow \mu x.V_2 \quad V_2 \Rightarrow \mu y.V_3 \quad V_3 \Rightarrow F(V_4, V_5) \quad V_4 \Rightarrow x \quad V_5 \Rightarrow y$$

Note that G generates the term $\mu x.\mu y.F(x, y)$.

Let $\mathcal{G}^\alpha$ has start variable $V_{1,\{x\mapsto x,y\mapsto y\},\varnothing}$, and contains rules:

$$V_{1,\{x\mapsto x,y\mapsto y\},\varnothing} \Rightarrow \mu y.V_{2,\{x\mapsto y,y\mapsto y\},\{y\}}$$
$$V_{2,\{x\mapsto y,y\mapsto y\},\{y\}} \Rightarrow \mu y.V_{3,\{x\mapsto y,y\mapsto y\},\{x\}}$$
$$V_{3,\{x\mapsto y,y\mapsto y\},\{x\}} \Rightarrow F(V_{4,\{x\mapsto y,y\mapsto y\},\{x\}}, V_{5,\{x\mapsto y,y\mapsto y\},\{x\}})$$
$$V_{3,\{x\mapsto y,y\mapsto y\},\{x\}} \Rightarrow F(V_{4,\{x\mapsto y,y\mapsto y\},\{x\}}, V_{5,\{x\mapsto y,y\mapsto y\},\{x\}})$$

Step 2: α-conversion

pick renaming: $V_{\sigma,\dagger} \Rightarrow \mu y(V'_{\sigma',\dagger'}) \in G$ if $V \Rightarrow \mu x(V') \in G$
where $y \in \mathbb{B}$, $\sigma' = \sigma[x \mapsto y]$, $\dagger' = (\dagger \cup \sigma^{-1}(y)) \setminus \{x\}$.

Example

Let G have start variable V_1 and consist of the rules:

$$V_1 \Rightarrow \mu x.V_2 \quad V_2 \Rightarrow \mu y.V_3 \quad V_3 \Rightarrow F(V_4, V_5) \quad V_4 \Rightarrow x \quad V_5 \Rightarrow y$$

Note that G generates the term $\mu x.\mu y.F(x, y)$.

Let $\mathcal{G}^\alpha$ has start variable $V_{1,\{x\mapsto x,y\mapsto y\},\varnothing}$, and contains rules:

$$V_{1,\{x\mapsto x,y\mapsto y\},\varnothing} \Rightarrow \mu y.V_{2,\{x\mapsto y,y\mapsto y\},\{y\}}$$
$$V_{2,\{x\mapsto y,y\mapsto y\},\{y\}} \Rightarrow \mu y.V_{3,\{x\mapsto y,y\mapsto y\},\{x\}}$$
$$V_{3,\{x\mapsto y,y\mapsto y\},\{x\}} \Rightarrow F(V_{4,\{x\mapsto y,y\mapsto y\},\{x\}}, V_{5,\{x\mapsto y,y\mapsto y\},\{x\}})$$
$$V_{3,\{x\mapsto y,y\mapsto y\},\{x\}} \Rightarrow F(V_{4,\{x\mapsto y,y\mapsto y\},\{x\}}, V_{5,\{x\mapsto y,y\mapsto y\},\{x\}})$$
$$V_{4,\{x\mapsto y,y\mapsto y\},\{x\}} \Rightarrow \bot$$

Step 2: α-conversion

pick renaming: $\mathsf{V}_{\sigma,\dagger} \Rightarrow \mu y(\mathsf{V}'_{\sigma',\dagger'}) \in G$ if $\mathsf{V} \Rightarrow \mu x(\mathsf{V}') \in G$
where $y \in \mathbb{B}$, $\sigma' = \sigma[x \mapsto y]$, $\dagger' = (\dagger \cup \sigma^{-1}(y)) \setminus \{x\}$.

Example

Let G have start variable V_1 and consist of the rules:

$$\mathsf{V}_1 \Rightarrow \mu x.\mathsf{V}_2 \quad \mathsf{V}_2 \Rightarrow \mu y.\mathsf{V}_3 \quad \mathsf{V}_3 \Rightarrow \mathsf{F}(\mathsf{V}_4,\mathsf{V}_5) \quad \mathsf{V}_4 \Rightarrow x \quad \mathsf{V}_5 \Rightarrow y$$

Note that G generates the term $\mu x.\mu y.\mathsf{F}(x,y)$.

Let $\mathcal{G}^\alpha$ has start variable $\mathsf{V}_{1,\{x\mapsto x,y\mapsto y\},\varnothing}$, and contains rules:

$$\begin{aligned}
\mathsf{V}_{1,\{x\mapsto x,y\mapsto y\},\varnothing} &\Rightarrow \mu y.\mathsf{V}_{2,\{x\mapsto y,y\mapsto y\},\{y\}}\\
\mathsf{V}_{2,\{x\mapsto y,y\mapsto y\},\{y\}} &\Rightarrow \mu y.\mathsf{V}_{3,\{x\mapsto y,y\mapsto y\},\{x\}}\\
\mathsf{V}_{3,\{x\mapsto y,y\mapsto y\},\{x\}} &\Rightarrow \mathsf{F}(\mathsf{V}_{4,\{x\mapsto y,y\mapsto y\},\{x\}},\mathsf{V}_{5,\{x\mapsto y,y\mapsto y\},\{x\}})\\
\mathsf{V}_{3,\{x\mapsto y,y\mapsto y\},\{x\}} &\Rightarrow \mathsf{F}(\mathsf{V}_{4,\{x\mapsto y,y\mapsto y\},\{x\}},\mathsf{V}_{5,\{x\mapsto y,y\mapsto y\},\{x\}})\\
\mathsf{V}_{4,\{x\mapsto y,y\mapsto y\},\{x\}} &\Rightarrow \bot\\
\mathsf{V}_{5,\{x\mapsto y,y\mapsto y\},\{x\}} &\Rightarrow y
\end{aligned}$$

Step 3: deciding $=_{\mu/\alpha}$

Theorem

The following problem is decidable:

- *Input: two μ-terms M and N.*
- *Answer: are M and N convertible?*

Proof.

The decision procedure proceeds in the following steps:

Step 3: deciding $=_{\mu/\alpha}$

Theorem

The following problem is decidable:

- *Input: two μ-terms M and N.*
- *Answer: are M and N convertible?*

Proof.

The decision procedure proceeds in the following steps:

1. take capture-avoiding μ-pseudoterms $M' =_\alpha M$, $N' =_\alpha N$,

Step 3: deciding $=_{\mu/\alpha}$

Theorem

The following problem is decidable:

- *Input: two μ-terms M and N.*
- *Answer: are M and N convertible?*

Proof.

The decision procedure proceeds in the following steps:

1. take capture-avoiding μ-pseudoterms $M' =_\alpha M$, $N' =_\alpha N$,
2. construct $\mathcal{G}_{M'}$ and $\mathcal{G}_{N'}$

Step 3: deciding $=_{\mu/\alpha}$

Theorem

The following problem is decidable:

- *Input: two μ-terms M and N.*
- *Answer: are M and N convertible?*

Proof.

The decision procedure proceeds in the following steps:

1. take capture-avoiding μ-pseudoterms $M' =_\alpha M$, $N' =_\alpha N$,
2. construct $\mathcal{G}_{M'}$ and $\mathcal{G}_{N'}$
3. construct $\mathcal{G}^{\alpha}_{M'}$ over the set of binders of M' and N', and

Step 3: deciding $=_{\mu/\alpha}$

Theorem

The following problem is decidable:

- *Input: two μ-terms M and N.*
- *Answer: are M and N convertible?*

Proof.

The decision procedure proceeds in the following steps:

1. take capture-avoiding μ-pseudoterms $M' =_\alpha M$, $N' =_\alpha N$,
2. construct $\mathcal{G}_{M'}$ and $\mathcal{G}_{N'}$
3. construct $\mathcal{G}^{\alpha}_{M'}$ over the set of binders of M' and N', and
4. answer yes if $\mathcal{L}(\mathcal{G}^{\alpha}_{M'}) \cap \mathcal{L}(\mathcal{G}_{N'}) \neq \varnothing$, and no, otherwise.

$\square$

Overview

1. Weak μ-equality
2. Avoiding α-conversion in μ-reductions
3. Decidability of $=_{\mu/\alpha}$ by a first-order proof
4. Decidability of $=_{\mu/\alpha}$ by a higher-order proof
5. Decidability of $=_{\mu/\alpha}$ using regular languages
6. Summary

Summary

We established decidability of the weak μ-equality problem by:

- a proof using ‘first-order’ techniques:
  - characterising μ-pseudoterms that can be reduced without the need for α-renaming:
  - a complete proof system à la Coppo/Cardone for $=_{\mu/\alpha}$ on μ-pseudoterms
  - showing finiteness of proof-search by establishing finiteness of the subterm closure for capture-avoiding μ-terms
- a proof using ‘higher-order’ techniques
- another proof using ‘first-order’ techniques:
  - the set of reducts of μ-pseudoterms form a regular tree language
  - weak μ-equality reduces to the emptiness problem for the intersection of regular tree languages