Proof Terms for Higher-Order Rewriting and Their Equivalence

February 14, 2023

Pablo Barenbaum
CONICET / Universidad Nacional de Quilmes
ICC, Universidad de Buenos Aires
Argentina

Eduardo Bonelli
Stevens Institute of Technology
USA

Outline

First-order proof terms

Higher-order proof terms

This work

Conclusion

Proof terms for first-order rewriting

First-order terms

$$
\begin{array}{lcll}
s & ::= & x & \text{variables} \\
  & \mid & \mathbf{f}(s_1, \ldots, s_n) & \text{applied function symbols}
\end{array}
$$

Proof terms for first-order rewriting

First-order terms

$$
\begin{array}{rcll}
s & ::= & x & \text{variables} \\
  & \mid & \mathbf{f}(s_1, \ldots, s_n) & \text{applied function symbols}
\end{array}
$$

First-order **proof terms**

$$
\begin{array}{rcll}
\rho & ::= & x & \\
  & \mid & \mathbf{f}(\rho_1, \ldots, \rho_n) & \\
  & \mid & \alpha(\rho_1, \ldots, \rho_n) & \text{applied } \textbf{rule symbols} \\
  & \mid & \rho_1 \,;\, \rho_2 & \text{sequential composition}
\end{array}
$$

Proof terms for first-order rewriting

First-order terms

$$\begin{array}{lcll} s & ::= & x & \text{variables} \\ & | & \mathbf{f}(s_1, \ldots, s_n) & \text{applied function symbols} \end{array}$$

First-order **proof terms**

$$\begin{array}{lcll} \rho & ::= & x & \\ & | & \mathbf{f}(\rho_1, \ldots, \rho_n) & \\ & | & \alpha(\rho_1, \ldots, \rho_n) & \text{applied } \textbf{rule symbols} \\ & | & \rho_1 \,\mathbf{;}\, \rho_2 & \text{sequential composition} \end{array}$$

Proof terms denote rewrites between terms.

Proof terms for first-order rewriting

$$\begin{array}{lcrcl} \alpha(x) & : & \mathbf{f}(x) & \rightharpoonup & \mathbf{g}(x,x) \\ \beta & : & \mathbf{c} & \rightharpoonup & \mathbf{d} \end{array}$$

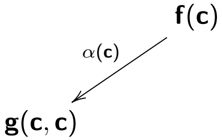
Proof terms for first-order rewriting

$$\begin{array}{lcrcl} \alpha(x) & : & \mathbf{f}(x) & \rightharpoonup & \mathbf{g}(x,x) \\ \beta & : & \mathbf{c} & \rightharpoonup & \mathbf{d} \end{array}$$

$$\mathbf{f}(\mathbf{c})$$

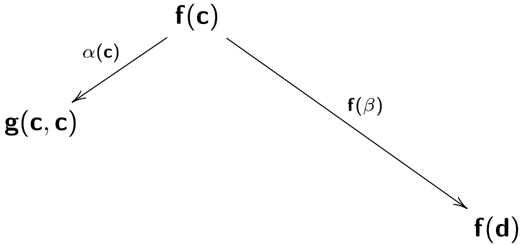
Proof terms for first-order rewriting

$$\begin{array}{lcrcl} \alpha(x) & : & \mathbf{f}(x) & \rightharpoonup & \mathbf{g}(x,x) \\ \beta & : & \mathbf{c} & \rightharpoonup & \mathbf{d} \end{array}$$



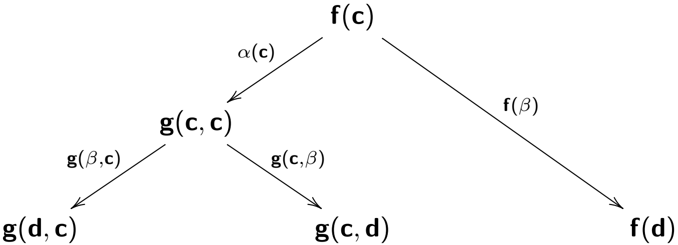
Proof terms for first-order rewriting

$$\begin{array}{lcrcl} \alpha(x) & : & \mathbf{f}(x) & \rightharpoonup & \mathbf{g}(x,x) \\ \beta & : & \mathbf{c} & \rightharpoonup & \mathbf{d} \end{array}$$



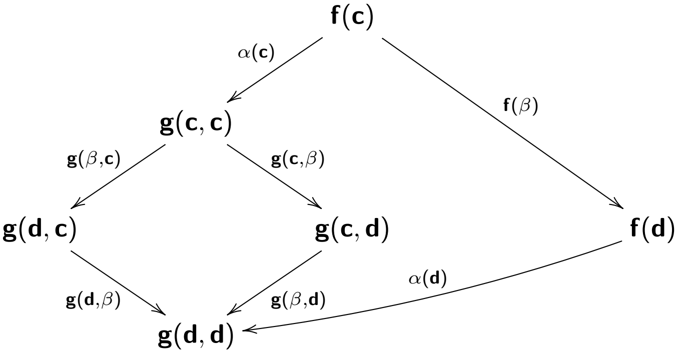
Proof terms for first-order rewriting

$$\begin{array}{lcrcl} \alpha(x) & : & \mathbf{f}(x) & \to & \mathbf{g}(x,x) \\ \beta & : & \mathbf{c} & \to & \mathbf{d} \end{array}$$



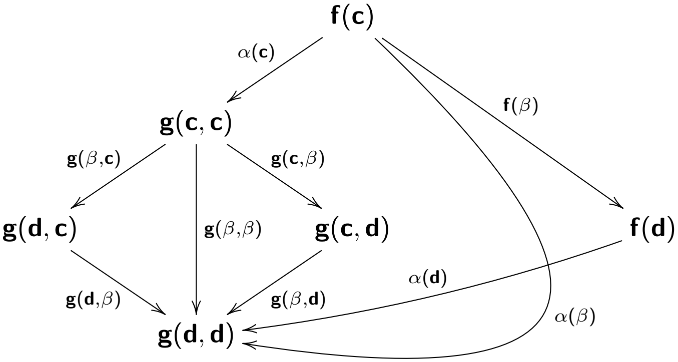
Proof terms for first-order rewriting

$$\begin{array}{lcrcl} \alpha(x) & : & \mathbf{f}(x) & \rightharpoonup & \mathbf{g}(x,x) \\ \beta & : & \mathbf{c} & \rightharpoonup & \mathbf{d} \end{array}$$



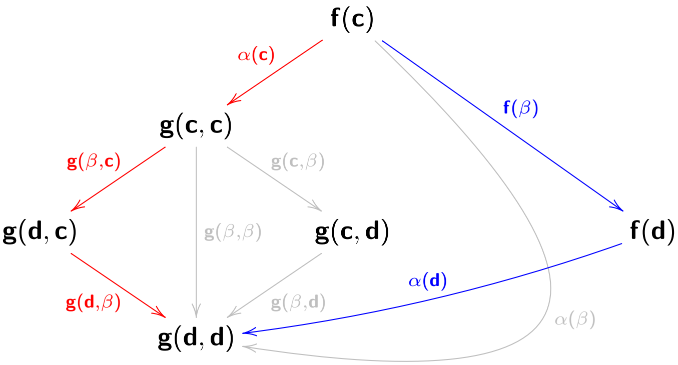
Proof terms for first-order rewriting

$$\begin{array}{lcrcl} \alpha(x) & : & \mathbf{f}(x) & \rightharpoonup & \mathbf{g}(x,x) \\ \beta & : & \mathbf{c} & \rightharpoonup & \mathbf{d} \end{array}$$



Proof terms for first-order rewriting

$$
\begin{array}{lcrcl}
\alpha(x) & : & \mathbf{f}(x) & \rightharpoonup & \mathbf{g}(x,x) \\
\beta & : & \mathbf{c} & \rightharpoonup & \mathbf{d}
\end{array}
$$

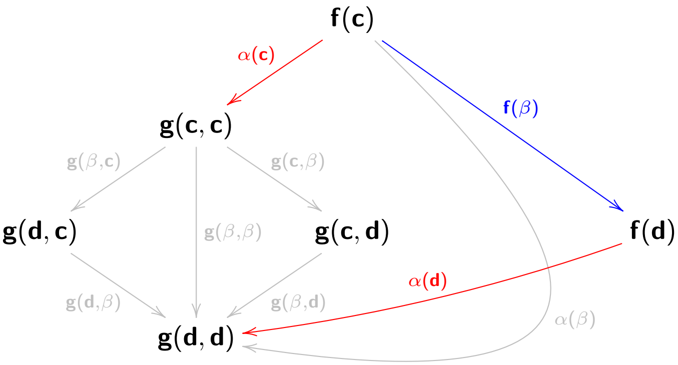


Permutation equivalence

$\alpha(\mathbf{c})\ ;\ \mathbf{g}(\beta,\mathbf{c})\ ;\ \mathbf{g}(\mathbf{d},\beta) \approx \mathbf{f}(\beta)\ ;\ \alpha(\mathbf{d})$

Proof terms for first-order rewriting

$$\begin{array}{lcrcl} \alpha(x) & : & \mathbf{f}(x) & \rightharpoonup & \mathbf{g}(x,x) \\ \beta & : & \mathbf{c} & \rightharpoonup & \mathbf{d} \end{array}$$



Projection
$\alpha(\mathbf{c}) \,/\, \mathbf{f}(\beta) = \alpha(\mathbf{d})$

Equivalent notions of equivalence

Theorem (de Vrijer, van Oostrom, 2002)

Building on previous work by Lévy, Huet, Meseguer, ...

The following are equivalent:

1. **Permutation equivalence:** $\rho \approx \sigma$.
2. **Projection equivalence:** ρ/σ and σ/ρ are empty.

Outline

First-order proof terms

Higher-order proof terms

This work

Conclusion

Proof terms for higher-order rewriting

Higher-order terms

$$
\begin{array}{lcll}
s & ::= & x & \text{variables} \\
 & | & \mathbf{c} & \text{constants} \\
 & | & \lambda x.s & \text{abstractions} \\
 & | & s_1\, s_2 & \text{applications}
\end{array}
$$

Proof terms for higher-order rewriting

Higher-order terms

s	$::=$	x	variables
	$\mid$	$\mathbf{c}$	constants
	$\mid$	$\lambda x.s$	abstractions
	$\mid$	$s_1\, s_2$	applications

Higher-order proof terms

ρ	$::=$	x	
	$\mid$	$\mathbf{c}$	
	$\mid$	$\lambda x.\rho$	
	$\mid$	$\rho_1\, \rho_2$	
	$\mid$	α	rule symbols
	$\mid$	$\rho_1 \mathbin{;} \rho_2$	sequential composition

Well-formed proof terms

$$\frac{}{x : x \rightarrow x} \qquad \frac{}{\mathbf{c} : \mathbf{c} \rightarrow \mathbf{c}} \qquad \frac{(\alpha : s \rightarrow t) \in \mathcal{R}}{\alpha : s \rightarrow t} \qquad \frac{\rho : s \rightarrow t}{\lambda x.\rho : \lambda x.s \rightarrow \lambda x.t}$$

$$\frac{\rho_1 : s_1 \rightarrow t_1 \qquad \rho_2 : s_2 \rightarrow t_2}{\rho_1\,\rho_2 : s_1\,s_2 \rightarrow t_1\,t_2} \qquad \frac{\rho_1 : s_1 \rightarrow s_2 \qquad \rho_2 : s_2 \rightarrow s_3}{\rho_1 \,;\, \rho_2 : s_1 \rightarrow s_3}$$

$$\frac{s =_{\beta\eta} s' \qquad \rho : s' \rightarrow t' \qquad t' =_{\beta\eta} t}{\rho : s \rightarrow t}$$

Well-formed proof terms

$$\frac{}{x : x \rightharpoonup x} \qquad \frac{}{\mathbf{c} : \mathbf{c} \rightharpoonup \mathbf{c}} \qquad \frac{(\alpha : s \rightharpoonup t) \in \mathcal{R}}{\alpha : s \rightharpoonup t} \qquad \frac{\rho : s \rightharpoonup t}{\lambda x.\rho : \lambda x.s \rightharpoonup \lambda x.t}$$

$$\frac{\rho_1 : s_1 \rightharpoonup t_1 \qquad \rho_2 : s_2 \rightharpoonup t_2}{\rho_1\, \rho_2 : s_1\, s_2 \rightharpoonup t_1\, t_2} \qquad \frac{\rho_1 : s_1 \rightharpoonup s_2 \qquad \rho_2 : s_2 \rightharpoonup s_3}{\rho_1 \,;\, \rho_2 : s_1 \rightharpoonup s_3}$$

$$\frac{s =_{\beta\eta} s' \qquad \rho : s' \rightharpoonup t' \qquad t' =_{\beta\eta} t}{\rho : s \rightharpoonup t}$$

If $\rho : s \rightharpoonup t$ then: $\quad \rho^{\text{src}} \stackrel{\text{def}}{=} s \qquad \rho^{\text{tgt}} \stackrel{\text{def}}{=} t.$

Well-formed proof terms

$$\frac{}{x : x \rightarrow x} \qquad \frac{}{\mathbf{c} : \mathbf{c} \rightarrow \mathbf{c}} \qquad \frac{(\alpha : s \rightarrow t) \in \mathcal{R}}{\alpha : s \rightarrow t} \qquad \frac{\rho : s \rightarrow t}{\lambda x.\rho : \lambda x.s \rightarrow \lambda x.t}$$

$$\frac{\rho_1 : s_1 \rightarrow t_1 \quad \rho_2 : s_2 \rightarrow t_2}{\rho_1\,\rho_2 : s_1\,s_2 \rightarrow t_1\,t_2} \qquad \frac{\rho_1 : s_1 \rightarrow s_2 \quad \rho_2 : s_2 \rightarrow s_3}{\rho_1 \mathbin{;} \rho_2 : s_1 \rightarrow s_3}$$

$$\frac{s =_{\beta\eta} s' \quad \rho : s' \rightarrow t' \quad t' =_{\beta\eta} t}{\rho : s \rightarrow t}$$

If $\rho : s \rightarrow t$ then: $\qquad \rho^{\mathsf{src}} \stackrel{\text{def}}{=} s \qquad \rho^{\mathsf{tgt}} \stackrel{\text{def}}{=} t.$

- Setting: orthogonal HRSs (in Nipkow's sense).
- Proof terms are simply-typed. (We omit the details in this talk).

A stumbling block

Proof terms for higher-order rewriting were studied by Bruggink ($\sim$2008).

What does "$(\lambda x.\rho)\,\sigma$" mean?

$$(\lambda x.\rho)\,\sigma \overset{?}{\approx} \rho\{x\backslash\sigma\}$$

A stumbling block

Proof terms for higher-order rewriting were studied by Bruggink ($\sim$2008).

What does "$(\lambda x.\rho)\,\sigma$" mean?

$$(\lambda x.\rho)\,\sigma \overset{?}{\approx} \rho\{x \backslash \sigma\}$$

As noted by Bruggink, this is not sound

Suppose that $\rho : s \rightharpoonup t$ is such that $s \neq t$. Then:

$$x \quad : \quad x \quad \rightharpoonup \quad x$$

A stumbling block

Proof terms for higher-order rewriting were studied by Bruggink ($\sim$2008).

What does "$(\lambda x.\rho)\,\sigma$" mean?

$$(\lambda x.\rho)\,\sigma \overset{?}{\approx} \rho\{x\backslash\sigma\}$$

As noted by Bruggink, this is not sound

Suppose that $\rho : s \rightharpoonup t$ is such that $s \neq t$. Then:

$$\begin{array}{lcll} x & : & x & \rightharpoonup & x \\ x \mathbin{;} x & : & x & \rightharpoonup & x \end{array}$$

A stumbling block

Proof terms for higher-order rewriting were studied by Bruggink ($\sim$2008).

What does "$(\lambda x.\rho)\,\sigma$" mean?

$$(\lambda x.\rho)\,\sigma \overset{?}{\approx} \rho\{x \backslash \sigma\}$$

As noted by Bruggink, this is not sound

Suppose that $\rho : s \rightharpoonup t$ is such that $s \neq t$. Then:

$$\begin{array}{llll} x & : & x & \rightharpoonup & x \\ x \,;\, x & : & x & \rightharpoonup & x \\ \lambda x.(x \,;\, x) & : & \lambda x.x & \rightharpoonup & \lambda x.x \end{array}$$

A stumbling block

Proof terms for higher-order rewriting were studied by Bruggink ($\sim$2008).

What does "$(\lambda x.\rho)\,\sigma$" mean?

$$(\lambda x.\rho)\,\sigma \overset{?}{\approx} \rho\{x\backslash\sigma\}$$

As noted by Bruggink, this is not sound

Suppose that $\rho : s \rightharpoonup t$ is such that $s \neq t$. Then:

$$\begin{array}{lcccc} x & : & x & \rightharpoonup & x \\ x\,;\,x & : & x & \rightharpoonup & x \\ \lambda x.(x\,;\,x) & : & \lambda x.x & \rightharpoonup & \lambda x.x \\ (\lambda x.(x\,;\,x))\,\rho & : & (\lambda x.x)\,s & \rightharpoonup & (\lambda x.x)\,t \end{array}$$

A stumbling block

Proof terms for higher-order rewriting were studied by Bruggink ($\sim$2008).

What does “$(\lambda x.\rho)\,\sigma$” mean?

$$(\lambda x.\rho)\,\sigma \overset{?}{\approx} \rho\{x\backslash\sigma\}$$

As noted by Bruggink, this is not sound

Suppose that $\rho : s \rightharpoonup t$ is such that $s \neq t$. Then:

$$\begin{array}{lcllcl} x & : & x & \rightharpoonup & x \\ x \mathbin{;} x & : & x & \rightharpoonup & x \\ \lambda x.(x \mathbin{;} x) & : & \lambda x.x & \rightharpoonup & \lambda x.x \\ (\lambda x.(x \mathbin{;} x))\,\rho & : & (\lambda x.x)\,s & \rightharpoonup & (\lambda x.x)\,t \\ (\lambda x.(x \mathbin{;} x))\,\rho & : & s & \rightharpoonup & t \end{array}$$

A stumbling block

Proof terms for higher-order rewriting were studied by Bruggink ($\sim$2008).

What does "$(\lambda x.\rho)\,\sigma$" mean?

$$(\lambda x.\rho)\,\sigma \overset{?}{\approx} \rho\{x \backslash \sigma\}$$

As noted by Bruggink, this is not sound

Suppose that $\rho : s \rightharpoonup t$ is such that $s \neq t$. Then:

$$\begin{array}{lcccl}
x & : & x & \rightharpoonup & x \\
x \,;\, x & : & x & \rightharpoonup & x \\
\lambda x.(x \,;\, x) & : & \lambda x.x & \rightharpoonup & \lambda x.x \\
(\lambda x.(x \,;\, x))\,\rho & : & (\lambda x.x)\,s & \rightharpoonup & (\lambda x.x)\,t \\
(\lambda x.(x \,;\, x))\,\rho & : & s & \rightharpoonup & t
\end{array}$$

But $(\rho \,;\, \rho)$ is not well-formed, as ρ cannot be composed with itself.

A stumbling block

Proof terms for higher-order rewriting were studied by Bruggink ($\sim$2008).

What does "$(\lambda x.\rho)\,\sigma$" mean?

$$(\lambda x.\rho)\,\sigma \stackrel{?}{\approx} \rho\{x \backslash \sigma\}$$

As noted by Bruggink, this is not sound

Suppose that $\rho : s \rightharpoonup t$ is such that $s \neq t$. Then:

$$\begin{array}{lcccc}
x & : & x & \rightharpoonup & x \\
x\,;\,x & : & x & \rightharpoonup & x \\
\lambda x.(x\,;\,x) & : & \lambda x.x & \rightharpoonup & \lambda x.x \\
(\lambda x.(x\,;\,x))\,\rho & : & (\lambda x.x)\,s & \rightharpoonup & (\lambda x.x)\,t \\
(\lambda x.(x\,;\,x))\,\rho & : & s & \rightharpoonup & t
\end{array}$$

But $(\rho\,;\,\rho)$ is not well-formed, as ρ cannot be composed with itself.

Bruggink sidesteps the problem by allowing compositions ("$;$") only at the toplevel.

Outline

Permutation equivalence for higher-order proof terms

Definition (Permutation equivalence)

$$
\begin{array}{rcll}
\rho^{\mathsf{src}} \,;\, \rho & \approx & \rho & \\
\rho \,;\, \rho^{\mathsf{tgt}} & \approx & \rho & \\
(\rho \,;\, \sigma) \,;\, \tau & \approx & \rho \,;\, (\sigma \,;\, \tau) & \\
(\lambda x.\rho) \,;\, (\lambda x.\sigma) & \approx & \lambda x.(\rho \,;\, \sigma) & \\
(\rho_1 \rho_2) \,;\, (\sigma_1 \sigma_2) & \approx & (\rho_1 \,;\, \sigma_1)(\rho_2 \,;\, \sigma_2) & \\
(\lambda x.s)\,\rho & \approx & s\{x \sslash \rho\} & \\
(\lambda x.\rho)\,s & \approx & \rho\{x \backslash s\} & \\
\lambda x.\rho\, x & \approx & \rho & \text{if } x \notin \mathsf{fv}(\rho)
\end{array}
$$

Permutation equivalence for higher-order proof terms

Definition (Permutation equivalence)

$$
\begin{array}{rcll}
\rho^{\mathsf{src}} ; \rho & \approx & \rho & \\
\rho ; \rho^{\mathsf{tgt}} & \approx & \rho & \\
(\rho ; \sigma) ; \tau & \approx & \rho ; (\sigma ; \tau) & \\
(\lambda x.\rho) ; (\lambda x.\sigma) & \approx & \lambda x.(\rho ; \sigma) & \\
(\rho_1 \rho_2) ; (\sigma_1 \sigma_2) & \approx & (\rho_1 ; \sigma_1)(\rho_2 ; \sigma_2) & \\
(\lambda x.s)\, \rho & \approx & s\{x \backslash\!\backslash \rho\} & \\
(\lambda x.\rho)\, s & \approx & \rho\{x \backslash s\} & \\
\lambda x.\rho\, x & \approx & \rho & \text{if } x \notin \mathsf{fv}(\rho)
\end{array}
$$

Permutation equivalence for higher-order proof terms

With these axioms, we can **prove**:

$$(\lambda x.\rho)\,\sigma \;\approx\; \rho\{x\backslash\sigma^{\mathsf{src}}\}\,;\,\rho^{\mathsf{tgt}}\{x\backslash\!\!\backslash\sigma\} \;\approx\; \rho^{\mathsf{src}}\{x\backslash\!\!\backslash\sigma\}\,;\,\rho\{x\backslash\sigma^{\mathsf{tgt}}\}$$

Flattening

Definition (Flattening)

$$
\begin{array}{rcll}
\lambda x.(\rho \mathbin{;} \sigma) & \overset{\flat}{\mapsto} & (\lambda x.\rho) \mathbin{;} (\lambda x.\sigma) & \\
(\rho \mathbin{;} \sigma)\,\mu & \overset{\flat}{\mapsto} & (\rho\,\mu^{\mathsf{src}}) \mathbin{;} (\sigma\,\mu) & \\
\mu\,(\rho \mathbin{;} \sigma) & \overset{\flat}{\mapsto} & (\mu\,\rho) \mathbin{;} (\mu^{\mathsf{tgt}}\,\sigma) & \\
(\rho_1 \mathbin{;} \rho_2)\,(\sigma_1 \mathbin{;} \sigma_2) & \overset{\flat}{\mapsto} & ((\rho_1 \mathbin{;} \rho_2)\,\sigma_1^{\mathsf{src}}) \mathbin{;} (\rho_2^{\mathsf{tgt}}\,(\sigma_1 \mathbin{;} \sigma_2)) & \\
(\lambda x.\mu)\,\nu & \overset{\flat}{\mapsto} & \mu\{x \backslash \nu\} & \\
\lambda x.\mu\,x & \overset{\flat}{\mapsto} & \mu & \text{if } x \notin \mathsf{fv}(\mu)
\end{array}
$$

μ, ν, etc. stand for **multisteps**.

Flattening

Definition (Flattening)

$$
\begin{array}{rcll}
\lambda x.(\rho \mathbin{;} \sigma) & \overset{\flat}{\mapsto} & (\lambda x.\rho) \mathbin{;} (\lambda x.\sigma) & \\
(\rho \mathbin{;} \sigma)\,\mu & \overset{\flat}{\mapsto} & (\rho\,\mu^{\mathsf{src}}) \mathbin{;} (\sigma\,\mu) & \\
\mu\,(\rho \mathbin{;} \sigma) & \overset{\flat}{\mapsto} & (\mu\,\rho) \mathbin{;} (\mu^{\mathsf{tgt}}\,\sigma) & \\
(\rho_1 \mathbin{;} \rho_2)\,(\sigma_1 \mathbin{;} \sigma_2) & \overset{\flat}{\mapsto} & ((\rho_1 \mathbin{;} \rho_2)\,\sigma_1^{\mathsf{src}}) \mathbin{;} (\rho_2^{\mathsf{tgt}}\,(\sigma_1 \mathbin{;} \sigma_2)) & \\
(\lambda x.\mu)\,\nu & \overset{\flat}{\mapsto} & \mu\{x \backslash \nu\} & \\
\lambda x.\mu\,x & \overset{\flat}{\mapsto} & \mu & \text{if } x \notin \mathsf{fv}(\mu)
\end{array}
$$

μ, ν, etc. stand for **multisteps**.

Theorem

Flattening is **confluent** and **strongly normalizing**.

Flattening

Definition (Flattening)

$$
\begin{array}{rcll}
\lambda x.(\rho \,;\, \sigma) & \overset{\flat}{\mapsto} & (\lambda x.\rho) \,;\, (\lambda x.\sigma) & \\
(\rho \,;\, \sigma)\,\mu & \overset{\flat}{\mapsto} & (\rho\,\mu^{\mathsf{src}}) \,;\, (\sigma\,\mu) & \\
\mu\,(\rho \,;\, \sigma) & \overset{\flat}{\mapsto} & (\mu\,\rho) \,;\, (\mu^{\mathsf{tgt}}\,\sigma) & \\
(\rho_1 \,;\, \rho_2)\,(\sigma_1 \,;\, \sigma_2) & \overset{\flat}{\mapsto} & ((\rho_1 \,;\, \rho_2)\,\sigma_1^{\mathsf{src}}) \,;\, (\rho_2^{\mathsf{tgt}}\,(\sigma_1 \,;\, \sigma_2)) & \\
(\lambda x.\mu)\,\nu & \overset{\flat}{\mapsto} & \mu\{x \backslash \nu\} & \\
\lambda x.\mu\,x & \overset{\flat}{\mapsto} & \mu & \text{if } x \notin \mathsf{fv}(\mu)
\end{array}
$$

μ, ν, etc. stand for **multisteps**.

Theorem

Flattening is **confluent** and **strongly normalizing**.

- Any proof term can be converted to a proof term in Bruggink's sense (with compositions only at the toplevel).
- Auxiliary tool to relate permutation and projection equivalence.

Flat permutation equivalence

Definition (Flat permutation equivalence)

A notion of permutation equivalence **between flat proof terms** can be defined as follows:

$$\begin{array}{lcll} (\rho \,;\, \sigma) \,;\, \tau & \sim & \rho \,;\, (\sigma \,;\, \tau) & \\ \mu & \sim & \mu_1^\flat \,;\, \mu_2^\flat & \text{if } \mu \Leftrightarrow \mu_1 \,;\, \mu_2 \end{array}$$

where $\mu \Leftrightarrow \mu_1 \,;\, \mu_2$ is a ternary relation meaning that the multistep μ can be "split" as the composition of the multisteps μ_1 and μ_2.

Theorem (Permutation equivalence through flattening)

$\rho \approx \sigma$ if and only if $\rho^\flat \sim \sigma^\flat$

Projection

A notion of **projection** can be defined for **multi-steps** (without composition):

$$\frac{}{x /\!\!/\!\!/ x \Rightarrow x} \qquad \frac{}{\mathbf{c} /\!\!/\!\!/ \mathbf{c} \Rightarrow \mathbf{c}}$$

$$\frac{\mu /\!\!/\!\!/ \nu \Rightarrow \xi}{\lambda x.\mu /\!\!/\!\!/ \lambda x.\nu \Rightarrow \lambda x.\xi} \qquad \frac{\mu_1 /\!\!/\!\!/ \nu_1 \Rightarrow \xi_1 \qquad \mu_2 /\!\!/\!\!/ \nu_2 \Rightarrow \xi_2}{\mu_1\, \mu_2 /\!\!/\!\!/ \nu_1\, \nu_2 \Rightarrow \xi_1\, \xi_2}$$

$$\frac{}{\alpha /\!\!/\!\!/ \alpha \Rightarrow \alpha^{\mathsf{tgt}}} \qquad \frac{}{\alpha /\!\!/\!\!/ \alpha^{\mathsf{src}} \Rightarrow \alpha} \qquad \frac{}{\alpha^{\mathsf{src}} /\!\!/\!\!/ \alpha \Rightarrow \alpha^{\mathsf{tgt}}}$$

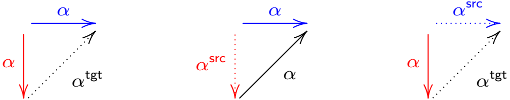
Projection

A notion of **projection** can be defined for **multi-steps** (without composition):

$$\frac{}{x \mathbin{/\!/\!/} x \Rightarrow x} \qquad \frac{}{\mathbf{c} \mathbin{/\!/\!/} \mathbf{c} \Rightarrow \mathbf{c}}$$

$$\frac{\mu \mathbin{/\!/\!/} \nu \Rightarrow \xi}{\lambda x.\mu \mathbin{/\!/\!/} \lambda x.\nu \Rightarrow \lambda x.\xi} \qquad \frac{\mu_1 \mathbin{/\!/\!/} \nu_1 \Rightarrow \xi_1 \qquad \mu_2 \mathbin{/\!/\!/} \nu_2 \Rightarrow \xi_2}{\mu_1\, \mu_2 \mathbin{/\!/\!/} \nu_1\, \nu_2 \Rightarrow \xi_1\, \xi_2}$$

$$\frac{}{\alpha \mathbin{/\!/\!/} \alpha \Rightarrow \alpha^{\mathsf{tgt}}} \qquad \frac{}{\alpha \mathbin{/\!/\!/} \alpha^{\mathsf{src}} \Rightarrow \alpha} \qquad \frac{}{\alpha^{\mathsf{src}} \mathbin{/\!/\!/} \alpha \Rightarrow \alpha^{\mathsf{tgt}}}$$



Projection

A notion of **projection** can be defined for **multi-steps** (without composition):

$$\frac{}{x /\!\!/\!\!/ x \Rightarrow x} \qquad \frac{}{\mathbf{c} /\!\!/\!\!/ \mathbf{c} \Rightarrow \mathbf{c}}$$

$$\frac{\mu /\!\!/\!\!/ \nu \Rightarrow \xi}{\lambda x.\mu /\!\!/\!\!/ \lambda x.\nu \Rightarrow \lambda x.\xi} \qquad \frac{\mu_1 /\!\!/\!\!/ \nu_1 \Rightarrow \xi_1 \qquad \mu_2 /\!\!/\!\!/ \nu_2 \Rightarrow \xi_2}{\mu_1\, \mu_2 /\!\!/\!\!/ \nu_1\, \nu_2 \Rightarrow \xi_1\, \xi_2}$$

$$\frac{}{\alpha /\!\!/\!\!/ \alpha \Rightarrow \alpha^{\mathsf{tgt}}} \qquad \frac{}{\alpha /\!\!/\!\!/ \alpha^{\mathsf{src}} \Rightarrow \alpha} \qquad \frac{}{\alpha^{\mathsf{src}} /\!\!/\!\!/ \alpha \Rightarrow \alpha^{\mathsf{tgt}}}$$

Projection can be extended to **flat** proof terms:

$$\begin{array}{rcll} \mu^\flat /\!\!/ \nu^\flat & \stackrel{\mathrm{def}}{=} & \xi^\flat & \text{if } \mu /\!\!/\!\!/ \nu \Rightarrow \xi \\ \rho /\!\!/ (\sigma\, ;\, \tau) & \stackrel{\mathrm{def}}{=} & (\rho /\!\!/ \sigma) /\!\!/ \tau & \\ (\rho\, ;\, \sigma) /\!\!/ \tau & \stackrel{\mathrm{def}}{=} & (\rho /\!\!/ \tau)\, ;\, (\sigma /\!\!/ (\tau /\!\!/ \rho)) & \end{array}$$

Projection

A notion of **projection** can be defined for **multi-steps** (without composition):

$$\frac{}{x \mathbin{/\!/\!/} x \Rightarrow x} \qquad \frac{}{\mathbf{c} \mathbin{/\!/\!/} \mathbf{c} \Rightarrow \mathbf{c}}$$

$$\frac{\mu \mathbin{/\!/\!/} \nu \Rightarrow \xi}{\lambda x.\mu \mathbin{/\!/\!/} \lambda x.\nu \Rightarrow \lambda x.\xi} \qquad \frac{\mu_1 \mathbin{/\!/\!/} \nu_1 \Rightarrow \xi_1 \qquad \mu_2 \mathbin{/\!/\!/} \nu_2 \Rightarrow \xi_2}{\mu_1\,\mu_2 \mathbin{/\!/\!/} \nu_1\,\nu_2 \Rightarrow \xi_1\,\xi_2}$$

$$\frac{}{\alpha \mathbin{/\!/\!/} \alpha \Rightarrow \alpha^{\mathsf{tgt}}} \qquad \frac{}{\alpha \mathbin{/\!/\!/} \alpha^{\mathsf{src}} \Rightarrow \alpha} \qquad \frac{}{\alpha^{\mathsf{src}} \mathbin{/\!/\!/} \alpha \Rightarrow \alpha^{\mathsf{tgt}}}$$

Projection can be extended to **flat** proof terms:

$$\begin{array}{rcll} \mu^\flat \mathbin{/\!/} \nu^\flat & \stackrel{\text{def}}{=} & \xi^\flat & \text{if } \mu \mathbin{/\!/\!/} \nu \Rightarrow \xi \\ \rho \mathbin{/\!/} (\sigma\,;\,\tau) & \stackrel{\text{def}}{=} & (\rho \mathbin{/\!/} \sigma) \mathbin{/\!/} \tau & \\ (\rho\,;\,\sigma) \mathbin{/\!/} \tau & \stackrel{\text{def}}{=} & (\rho \mathbin{/\!/} \tau)\,;\,(\sigma \mathbin{/\!/} (\tau \mathbin{/\!/} \rho)) & \end{array}$$

And finally, to **arbitrary** proof terms, by flattening first:

$$\rho/\sigma \stackrel{\text{def}}{=} \rho^\flat \mathbin{/\!/} \sigma^\flat$$

Projection equivalence

Theorem (Permutation equivalence through projection)

$\rho \approx \sigma$ if and only if ρ/σ and σ/ρ are empty.

Projection equivalence

Theorem (Permutation equivalence through projection)

$\rho \approx \sigma$ if and only if ρ/σ and σ/ρ are empty.

Corollary

Permutation equivalence is decidable.

Outline

First-order proof terms

Higher-order proof terms

This work

Conclusion

Conclusion

We have proposed a notion of **proof terms for higher-order rewriting** that allows to freely use the composition operator.

We have characterized permutation equivalence in three equivalent ways:

1. $\rho \approx \sigma$
2. $\rho^\flat \sim \sigma^\flat$
3. ρ/σ and σ/ρ are empty

Conclusion

We have proposed a notion of **proof terms for higher-order rewriting** that allows to freely use the composition operator.

We have characterized permutation equivalence in three equivalent ways:

1. $\rho \approx \sigma$
2. $\rho^\flat \sim \sigma^\flat$
3. ρ/σ and σ/ρ are empty

Future work

- Formulate a standardization procedure.
- Study labeling equivalence.
- Study 2-categorical models (Hirschowitz, 2013).
- Develop tools to manipulate reductions in higher-order rewriting.