

Rewriting, Explicit Substitutions and Normalisation

XXXVI Escola de Verão do MAT

Universidade de Brasília

Part 2/3

Eduardo Bonelli

LIFIA (Fac. de Informática, UNLP, Arg.) and CONICET
eduardo@lifia.info.unlp.edu.ar

February, 2006

Structure of Today's Talk

- 1 Residuals
- 2 Standardization
- 3 Needed Strategies

From hereon we work with left-linear TRS

1 Residuals

- Examples and Definition
- Equivalence of Derivations

2 Standardization

3 Needed Strategies

Example

Consider the TRS

$$\begin{aligned}\rho &: a && \rightarrow b \\ \vartheta &: f(x, a) && \rightarrow g(x, x)\end{aligned}$$

and the term

$$f(a, a)$$

It has three redexes, r , s and t :

$$f(a, a) \quad f(a, a) \quad f(a, a)$$

Example

$$\begin{aligned}\rho &: a && \rightarrow b \\ \vartheta &: f(x, a) && \rightarrow g(x, x)\end{aligned}$$

Consider the redexes r and s :

$$f(a, a) \quad f(a, a)$$

Reducing s leaves a “leftover” or *residual* of r

$$f(a, a) \rightarrow_{\rho} f(b, a)$$

Likewise reducing r leaves a “leftover” or *residual* (two actually) of s

$$f(a, a) \rightarrow_{\vartheta} g(a, a)$$

Note: r and s do *not* overlap

Example

$$\begin{aligned}\rho &: a && \rightarrow b \\ \vartheta &: f(x, a) && \rightarrow g(x, x)\end{aligned}$$

Consider the redexes r and t :

$$f(a, a) \quad f(a, a)$$

Reducing r leaves no residual of t ; Reducing t leaves no residual of r

- Note: r and t overlap
- Only other case where a redex leaves no residual: when it is **erased**.
Eg. replacing ϑ by $f(x, a) \rightarrow b$, note a is erased below

$$f(a, a) \rightarrow_{\vartheta} b$$

Definition of Residuals

Assume ρ -redex r and ϑ -redex s in M and $M \rightarrow_r N$

What happens with s after the r -step?

Consider all cases:

- 1 They are disjoint: s appears in N
- 2 They are equal: s is erased in N
- 3 s is in an argument of r : s appears $n \geq 0$ times in N
- 4 r is in an argument of s : s appears in N with a different argument
- 5 r and s overlap: s is erased

In general, there is no sense in defining the residual of a redex after an **overlapping** reduction step (case 5)

Residual relation

Let $r : M \rightarrow N$. The **residual relation** for r

$$-/r$$

is defined as above: it maps nonoverlapping redexes in M to the set of their residuals

Basic properties

- 1 $r/r = \emptyset$
- 2 s/r is a **finite** set of redexes

Redex Creation

Let $r : M \rightarrow N$. Redexes in N that are **not** residuals of those in M are called **created**

$$\begin{aligned}\rho &: a && \rightarrow b \\ \vartheta &: f(x, b) && \rightarrow g(x, x)\end{aligned}$$

The redex $f(a, b)$ is created in

$$f(a, a) \rightarrow_{\rho} f(a, b)$$

1 Residuals

- Examples and Definition
- Equivalence of Derivations

2 Standardization

3 Needed Strategies

Residual Relation on Derivations

The residual relation extends to **derivations**

$$u \in s/(r;d) \text{ iff } \exists v \text{ s.t. } v \in s/r \text{ and } u \in v/d$$

Informally,

$$\begin{array}{ccccc} M_1 & \xrightarrow{r} & M_2 & \xrightarrow{d} & M_n \\ s \downarrow & & v \downarrow & & u \downarrow \\ \downarrow & & \downarrow & & \downarrow \end{array}$$

Multi-redex

Multi-redex is a pair $\langle M, U \rangle$ where U is a finite set of nonoverlapping redexes in M

Residual relation $/_d$ extends to multi-redexes:

$$\langle M, U \rangle /_d \langle N, V \rangle$$

when

- 1 $d : M \rightarrow N$
- 2 V is the set of residuals of elements of U after d

$$V = \{v \mid \exists u \in U, u /_d v\}$$

Development

A derivation

$$d : M = M_1 \rightarrow_{r_1} M_2 \rightarrow_{r_2} M_3 \dots \rightarrow_{r_{n-1}} M_n \rightarrow_{r_n} M_{n+1}$$

develops a multi-redex $\langle M, U \rangle$ **partially** when every redex r_i is an element of the multi-redex

$$\langle M, U \rangle /_{r_1; \dots; r_{i-1}}$$

We say $d : M \rightarrow M_{n+1}$ **develops** the multi-redex $\langle M, U \rangle$ when d develops $\langle M, U \rangle$ partially and

$$\langle M, U \rangle /_d = \langle M_{n+1}, \emptyset \rangle$$

Example

Consider the TRS

$$\rho: a \rightarrow b$$

$$\vartheta: f(x, b) \rightarrow g(x, x)$$

and the multi-redex

$$\langle f(a, b), \{r, s\} \rangle$$

- 1 The derivation $f(a, b) \rightarrow f(b, b)$ partially develops this multi-redex
- 2 Both derivations below develop this multi-redex

$$f(a, b) \rightarrow f(b, b) \rightarrow g(b, b) \text{ and}$$

$$f(a, b) \rightarrow g(a, a) \rightarrow g(b, a) \rightarrow g(b, b)$$

- 3 For the multi-redex $\langle f(a, b), \{s\} \rangle$ the derivation

$$f(a, b) \rightarrow g(a, a)$$

is not a partial development

Finite Developments

Lemma

For every multi-redex $\langle M, U \rangle$, there does **not** exist any infinite derivation

$$M_1 \rightarrow_{r_1} M_2 \rightarrow_{r_2} M_3 \rightarrow_{r_3} \dots$$

s.t. for each i the derivation

$$M_1 \rightarrow_{r_1} M_2 \rightarrow_{r_2} M_3 \rightarrow_{r_3} \dots M_{i-1} \rightarrow_{r_{i-1}} M_i$$

develops $\langle M, U \rangle$ partially

Informally,

Contraction (only) of residuals of a fixed set U of redexes in M eventually terminates

Basic Tile

Lemma (Parallel Moves)

For every two cointial, non-overlapping redexes $r : M \rightarrow P$ and $s : M \rightarrow Q$ there exists two derivations d_r and d_s s.t.

- 1 d_r develops r/s and d_s develops s/r
- 2 d_r and d_s are cofinal and induce the same residual relation

$$\begin{array}{ccc} M & \xrightarrow{r} & P \\ s \downarrow & & \downarrow d_s \\ Q & \xrightarrow{d_r} & N \end{array}$$

Basic Tile and Equivalence of Derivations

- Basic tile provides a convenient mechanism for defining a notion of **equivalence of derivations**
- **Intuition:** $d : M \twoheadrightarrow N$ and $e : M \twoheadrightarrow N$ are “equivalent” if they do the same “work” but in different “order”

$$\begin{array}{ccc} f(a, a) & \xrightarrow{a} & f(b, a) \\ \downarrow a & & \downarrow a \\ f(a, b) & \xrightarrow{a} & f(b, b) \\ \downarrow f(\square, b) & & \downarrow f(\square, b) \\ g(a, a) & \xrightarrow{a} g(a, b) \xrightarrow{a} & g(b, b) \end{array}$$

Lévy Permutation Equivalence

Write $f \equiv^1 g$ if $f = f_1; r; d_s; f_2$ and $g = f_1; s; d_r; f_2$ and the diagram below is a basic tile

$$\begin{array}{ccccc} R & \xrightarrow{\gg} & M & \xrightarrow{r} & P \\ & & \downarrow s & & \downarrow d_s \\ & & Q & \xrightarrow{\gg} & N & \xrightarrow{\gg} & S \\ & & & \downarrow d_r & & & \end{array}$$

Lévy permutation equivalence is the least equivalence relation on derivations containing \equiv^1

Informally,

- $f \equiv g$ if there is a finite sequence of basic tilings connecting f and g

Lévy Permutation Equivalence - Example Revisited

$$\begin{array}{ccc} f(a, a) & \xrightarrow{a} & f(b, a) \\ a \downarrow & \equiv & a \downarrow \\ f(a, b) & \xrightarrow{a} & f(b, b) \\ f(\square, b) \downarrow & \equiv & f(\square, b) \downarrow \\ g(a, a) & \xrightarrow{a} g(a, b) \xrightarrow{a} & g(b, b) \end{array}$$

Epimorphism

Lemma ([Berry] for λ)

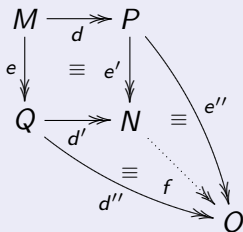
$d; e \equiv d; f$ implies $e \equiv f$

All arrows are epi in the category generated by the reduction graph of an OTRS with \equiv as identity on arrows

Algebraic Confluence

Thm ([Lévy1978] for λ)

Let $d : M \rightarrow P$ and $e : M \rightarrow Q$ be coinital in an OTRS. Then:



The category generated by the reduction graph of an OTRS with \equiv as identity on arrows enjoys pushouts

- 1 Residuals
- 2 Standardization
- 3 Needed Strategies

Standardization

The idea: for any derivation from M to N , there is a **canonical** derivation from M to N that computes redexes “outside-in”

$$\begin{array}{lcl} a & \rightarrow & b \\ f(x, b) & \rightarrow & g(x, x) \end{array}$$

Not standard

$$f(a, b) \rightarrow f(b, b) \rightarrow g(b, b)$$

Standard

$$f(a, b) \rightarrow g(a, a) \rightarrow g(b, a) \rightarrow g(b, b)$$

Examples - Uniqueness

$$\begin{array}{lcl} a & \rightarrow & b \\ f(x, b) & \rightarrow & g(x, x) \end{array}$$

Both these derivations are standard

$$\begin{array}{l} f(a, a) \rightarrow f(b, a) \rightarrow f(b, b) \\ f(a, a) \rightarrow f(a, b) \rightarrow f(b, b) \end{array}$$

- They are essentially the same (compute disjoint redexes in different order)!
- We thus identify derivations differing in this inessential way

Reversible Permutation Equivalence

Write $f \simeq^1 g$ if $f = f_1; r; d_s; f_2$ and $g = f_1; s; d_r; f_2$ and r, s are disjoint and the diagram below is a basic tile

$$\begin{array}{ccccc} R & \xrightarrow{\gg} & M & \xrightarrow{r} & P \\ & & \downarrow s & & \downarrow s' \\ & & Q & \xrightarrow{\gg} & N & \xrightarrow{\gg} & S \\ & & & & \downarrow r' & & \downarrow f_2 \end{array}$$

Reversible permutation equivalence is the least equivalence relation on derivations containing \simeq^1

Informally,

- $f \simeq g$ if there is a finite sequence of swappings of disjoint redexes from f to g

Defining Standard Derivations Through Permutation

Example 1

$$\begin{array}{ccc} f(a, b) & \xrightarrow{a} & f(b, b) \\ & & \downarrow f(\square, b) \\ & & g(b, b) \end{array}$$

Defining Standard Derivations Through Permutation

Example 1

$$\begin{array}{ccc} f(a, b) & \xrightarrow{a} & f(b, b) \\ f(\square, b) \downarrow & & f(\square, b) \downarrow \\ g(a, a) & \xrightarrow{a} & g(b, b) \end{array}$$

Defining Standard Derivations Through Permutation

Example 1

$$\begin{array}{ccc} f(a, b) & \xrightarrow{a} & f(b, b) \\ f(\square, b) \downarrow & & f(\square, b) \downarrow \\ g(a, a) & \xrightarrow{a} & g(b, b) \end{array}$$

Example 2

$$\begin{array}{ccc} f(a, a) & \xrightarrow{a} & f(b, a) \\ & & a \downarrow \\ & & f(b, b) \\ & & f(\square, b) \downarrow \\ & & g(b, b) \end{array}$$

Defining Standard Derivations Through Permutation

Example 1

$$\begin{array}{ccc} f(a, b) & \xrightarrow{a} & f(b, b) \\ f(\square, b) \downarrow & & f(\square, b) \downarrow \\ g(a, a) & \xrightarrow{a} & g(b, b) \end{array}$$

Example 2

$$\begin{array}{ccc} f(a, a) & \xrightarrow{a} & f(b, a) \\ a \downarrow & \simeq & a \downarrow \\ f(a, b) & \xrightarrow{a} & f(b, b) \\ & & f(\square, b) \downarrow \\ & & g(b, b) \end{array}$$

Defining Standard Derivations Through Permutation

Example 1

$$\begin{array}{ccc} f(a, b) & \xrightarrow{a} & f(b, b) \\ f(\square, b) \downarrow & & f(\square, b) \downarrow \\ g(a, a) & \xrightarrow{a} & g(b, b) \end{array}$$

Example 2

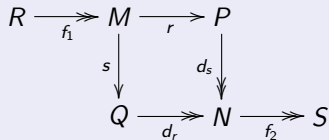
$$\begin{array}{ccc} f(a, a) & \xrightarrow{a} & f(b, a) \\ a \downarrow & \simeq & a \downarrow \\ f(a, b) & \xrightarrow{a} & f(b, b) \\ f(\square, b) \downarrow & & f(\square, b) \downarrow \\ g(a, a) & \xrightarrow{a} & g(b, b) \end{array}$$

Standardizing Permutation

We need one more ingredient for defining standard derivations

There is a **standardizing** permutation from $f : M \rightarrow N$ to $g : M \rightarrow N$ (written $f \Rightarrow g$) iff

- 1 $f = f_1; r; d_s; f_2$ and $g = f_1; s; d_r; f_2$
- 2 s nests r and
- 3 the diagram below is a basic tile



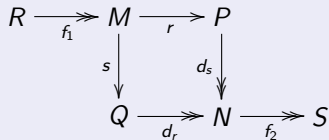
In fact, since s nests r , d_s will consist of just one redex
So we can write this definition more accurately as follows

Standardizing Permutation

We need one more ingredient for defining standard derivations

There is a **standardizing** permutation from $f : M \rightarrow N$ to $g : M \rightarrow N$ (written $f \Rightarrow g$) iff

- 1 $f = f_1; r; d_s; f_2$ and $g = f_1; s; d_r; f_2$
- 2 s nests r and
- 3 the diagram below is a basic tile



In fact, since s nests r , d_s will consist of just one redex
So we can write this definition more accurately as follows

Standardizing Permutation

There is a **standardizing** permutation from $f : M \twoheadrightarrow N$ to $g : M \twoheadrightarrow N$ (written $f \Rightarrow g$) iff

- 1 $f = f_1; r; s'; f_2$ and $g = f_1; s; d_r; f_2$
- 2 s nests r and
- 3 the diagram below is a basic tile

$$\begin{array}{ccccc} R & \twoheadrightarrow_{f_1} & M & \xrightarrow{r} & P \\ & & \downarrow s & & \downarrow s' \\ & & Q & \twoheadrightarrow_{d_r} & N \twoheadrightarrow_{f_2} S \end{array}$$

Standard Derivation

A derivation f is **standard** if there is no derivation g s.t.

$$[f]_{\simeq} \Rightarrow [g]_{\simeq}$$

Note: $[f]_{\simeq}$ is the reversible permutation equivalence class of f

Informally,

f is standard if no disjoint permutation of redexes of f gives rise to a standardizing permutation.

Example 1 - Revisited

$$\begin{array}{l} a \quad \rightarrow \quad b \\ f(x, b) \rightarrow g(x, x) \end{array}$$

The derivation $d : f(a, b) \rightarrow f(b, b) \rightarrow g(b, b)$ is not standard

$$\begin{array}{ccc} f(a, b) & \xrightarrow{a} & f(b, b) \\ \downarrow f(\square, b) & & \downarrow f(\square, b) \\ g(a, a) & \xrightarrow{a} & g(b, b) \end{array}$$

Indeed,

$$d \Rightarrow f(a, b) \rightarrow g(a, a) \rightarrow g(b, a) \rightarrow g(b, b)$$

Example 2 - Revisited

$$\begin{array}{l} a \quad \rightarrow \quad b \\ f(x, b) \rightarrow g(x, x) \end{array}$$

The derivation $d : f(a, a) \rightarrow f(b, a) \rightarrow f(b, b) \rightarrow g(b, b)$ is not standard

$$\begin{array}{ccc} f(a, a) & \xrightarrow{a} & f(b, a) \\ \downarrow a & \simeq & \downarrow a \\ f(a, b) & \xrightarrow{a} & f(b, b) \\ \downarrow f(\square, b) & \leftarrow & \downarrow f(\square, b) \\ g(a, a) & \xrightarrow{a} & g(b, b) \end{array}$$

Indeed, $d \simeq f(a, a) \rightarrow f(a, b) \rightarrow f(b, b) \rightarrow g(b, b)$
 $\Rightarrow f(a, a) \rightarrow f(a, b) \rightarrow g(a, a) \rightarrow g(b, a) \rightarrow g(b, b)$

Standardization Theorem

Thm

① Existence

For every $d : M \rightarrow N$, there exists a standard derivation $\text{std}(d) : M \rightarrow N$ and d_1, \dots, d_n s.t.

$$[d]_{\simeq} \Rightarrow [d_1]_{\simeq} \Rightarrow \dots \Rightarrow [d_n]_{\simeq} \Rightarrow [\text{std}(d)]_{\simeq}$$

② Uniqueness

Let $d, e : M \rightarrow N$ s.t. $d \equiv e$ (i.e. d and e are Lévy permutation equivalent). Then

$$[\text{std}(d)]_{\simeq} = [\text{std}(e)]_{\simeq}$$

Computing Standard Derivations

- 1 Repeatedly apply standardizing permutations on \simeq -equivalence classes
 - ▶ See [Terese, Sec.8.5.3 (“Inversion Parallel Standardization”)] where this process is shown to be CR and SN
- 2 Alternative standardization procedure
 - ▶ Given $d : M \rightarrow N$ compute $\text{std}(d)$ by repeatedly **extracting** outermost redexes contracted in d
 - ▶ This yields a standard derivation \equiv -equivalent to d
 - ▶ See [Terese, Sec.8.5.2. (“Selection Parallel Standardization”)]

1 Residuals

2 Standardization

3 Needed Strategies

- Needed Redexes
- Needed Redexes and Standardization
- Neededness for Non-Orthogonal Systems

Reduction Strategy

A (one-step or many step) **reduction strategy** for a TRS \mathcal{R} is a function $\mathbb{F} : \mathcal{T}(\Sigma) \rightarrow \mathcal{T}(\Sigma)$ s.t.

- 1 $\mathbb{F}(M) = M$, if M is in \mathcal{R} -normal form
- 2 $M \rightarrow^+ \mathbb{F}(M)$, otherwise

\mathbb{F} is **normalizing** iff for every WN term M there is no infinite reduction sequence

$$M \rightarrow^+ \mathbb{F}(M) \rightarrow^+ \mathbb{F}(\mathbb{F}(M)) \rightarrow^+ \mathbb{F}(\mathbb{F}(\mathbb{F}(M))) \rightarrow^+ \dots$$

Example

Consider the TRS

$$\begin{array}{l} f(a, x) \rightarrow x \quad f(b, x) \rightarrow b \\ g(a, x) \rightarrow a \quad g(b, x) \rightarrow x \end{array}$$

$f(f(a, f(a, b)), g(f(a, b), g(b, a)))$ leftmost-innermost
 $f(f(a, f(a, b)), g(f(a, b), g(b, a)))$ leftmost-outermost
 $f(f(a, f(a, b)), g(f(a, b), g(b, a)))$ parallel-innermost

Needed Redexes

A redex r in M is **needed** if in any reduction to normal form from M either

- 1 some residual of r is reduced or
- 2 a redex that overlaps with a residual of r is reduced

- Intuition: a redex is needed if it is unavoidable
- In the case of OTRS only the first item above can hold

A **needed strategy** performs needed steps

Example

$$\begin{aligned}\rho &: a && \rightarrow b \\ \vartheta &: f(x, b) && \rightarrow c\end{aligned}$$

a is needed in $f(a, a)$

$$f(a, a) \rightarrow_{\rho} f(b, a) \rightarrow_{\rho} f(b, b) \rightarrow_{\vartheta} c$$

a is **not** needed in $f(a, a)$

$$f(a, a) \rightarrow_{\rho} f(a, b) \rightarrow_{\vartheta} c$$

Problem - Needed redexes need not exist

$$\begin{aligned}\rho &: a && \rightarrow b \\ \vartheta &: f(b, x) && \rightarrow c \\ \theta &: f(x, b) && \rightarrow c\end{aligned}$$

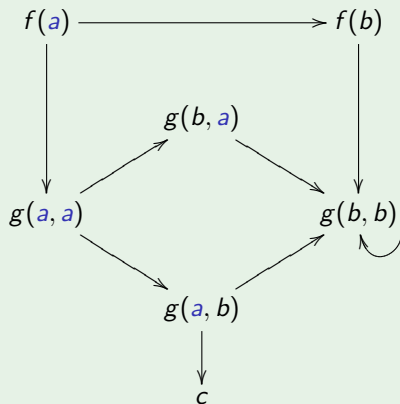
$f(a, a)$ has no needed redex

① $f(a, a) \rightarrow_{\rho} f(a, b) \rightarrow_{\theta} c$

② $f(a, a) \rightarrow_{\rho} f(b, a) \rightarrow_{\vartheta} c$

Problem - Needed redexes need not normalise

$a \rightarrow b$
 $f(x) \rightarrow g(x, x)$
 $g(a, b) \rightarrow c$
 $g(b, b) \rightarrow g(b, b)$



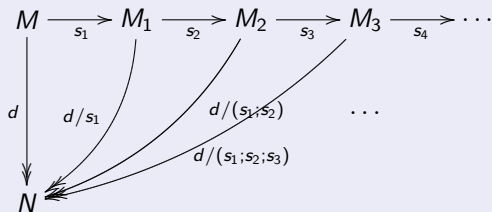
Why bother? - Normalisation Theorem I

Thm ([Huet,Lévy1991])

For orthogonal systems needed strategies normalise

Proof

Consider a **standard** normalising reduction sequence $d : M \rightarrow N$ and an infinite reduction sequence of needed steps from M : $M \rightarrow_{s_1} M_1 \rightarrow_{s_2} M_2 \rightarrow_{s_3} \dots$



Each $d/(s_1; \dots; s_i)$ is std and $|d| > |d/s_1| > |d/s_1; s_2| > \dots$

On Deciding Neededness

- Every term in an OTRS has a needed redex
- However, it is **not** decidable whether a redex is needed or not

Consider the OTRS consisting of Combinatory Logic plus the rules

$$g(a, b, x) \rightarrow c$$

$$g(x, a, b) \rightarrow c$$

$$g(b, x, a) \rightarrow c$$

Consider determining whether any of the redexes s_1, s_2, s_3 are needed in

$f(s_1, s_2, s_3)$

(Recall that the word problem for CL is undecidable)

On Deciding Neededness

- Nevertheless, for certain classes of OTRS some strategies can be proved needed
 - 1 Leftmost-outermost for left-normal (all function symbols occur to the left of variables). Eg. Combinatory Logic
 - 2 Leftmost-outermost for β

1 Residuals

2 Standardization

3 Needed Strategies

- Needed Redexes
- Needed Redexes and Standardization
- Neededness for Non-Orthogonal Systems

Needed Redexes and Standardization

- Recall

A redex $r : M \rightarrow N$ is needed if it is unavoidable to perform r in order to reach a normal form

- Alternatively

It is needed if one cannot get rid of it in any coinital derivation

Q: How can one “get rid of r ”?

A: Erase r from above

Needed Redexes and Standardization

Q: How can one “get rid of r ”?

A: Erase r from above by

- 1 reducing an **existing** redex in M above r that has r in one of its erased arguments or
- 2 A derivation from M that **creates** a redex above r that has r in one of its erased the arguments

Consider the TRS $a \rightarrow b, f(x, b) \rightarrow c$

- 1 $f(b, b) \rightarrow c$
- 2 $f(b, a) \rightarrow f(b, b) \rightarrow c$

Note: These are standard derivations that erase b

Needed Redexes and Standardization

Consider the TRS $a \rightarrow b$, $f(x, b) \rightarrow c$

- 1 $f(b, b) \rightarrow c$
- 2 $f(b, a) \rightarrow f(b, b) \rightarrow c$

Note:

- 1 If we extend each derivation by prefixing it with an $r : f(a, b) \rightarrow f(b, b)$ we get a **nonstandard** derivation
- 2 Moreover, standardizing these extended derivations **eliminates** r

Consider the second item above:

$$f(a, a) \rightarrow f(b, a) \rightarrow f(b, b) \rightarrow c$$

Needed Redexes and Standardization

Consider the TRS $a \rightarrow b, f(x, b) \rightarrow c$

- 1 $f(b, b) \rightarrow c$
- 2 $f(b, a) \rightarrow f(b, b) \rightarrow c$

Note:

- 1 If we extend each derivation by prefixing it with an $r : f(a, b) \rightarrow f(b, b)$ we get a **nonstandard** derivation
- 2 Moreover, standardizing these extended derivations **eliminates** r

Consider the second item above:

$$\underline{f(a, a) \rightarrow f(b, a) \rightarrow f(b, b) \rightarrow c}$$

Reversible permutation

Needed Redexes and Standardization

Consider the TRS $a \rightarrow b$, $f(x, b) \rightarrow c$

① $f(b, b) \rightarrow c$

② $f(b, a) \rightarrow f(b, b) \rightarrow c$

Note:

① If we extend each derivation by prefixing it with an $r : f(a, b) \rightarrow f(b, b)$ we get a **nonstandard** derivation

② Moreover, standardizing these extended derivations **eliminates** r

Consider the second item above:

$$f(a, a) \rightarrow f(b, a) \rightarrow f(b, b) \rightarrow c$$

$$f(a, a) \rightarrow f(a, b) \rightarrow f(b, b) \rightarrow c$$

Needed Redexes and Standardization

Consider the TRS $a \rightarrow b, f(x, b) \rightarrow c$

① $f(b, b) \rightarrow c$

② $f(b, a) \rightarrow f(b, b) \rightarrow c$

Note:

① If we extend each derivation by prefixing it with an $r : f(a, b) \rightarrow f(b, b)$ we get a **nonstandard** derivation

② Moreover, standardizing these extended derivations **eliminates** r

Consider the second item above:

$$f(a, a) \rightarrow f(b, a) \rightarrow f(b, b) \rightarrow c$$

$$f(a, a) \rightarrow \underline{f(a, b)} \rightarrow f(b, b) \rightarrow c$$

Standardizing permutation

Needed Redexes and Standardization

Consider the TRS $a \rightarrow b$, $f(x, b) \rightarrow c$

① $f(b, b) \rightarrow c$

② $f(b, a) \rightarrow f(b, b) \rightarrow c$

Note:

① If we extend each derivation by prefixing it with an $r : f(a, b) \rightarrow f(b, b)$ we get a **nonstandard** derivation

② Moreover, standardizing these extended derivations **eliminates** r

Consider the second item above:

$$f(a, a) \rightarrow f(b, a) \rightarrow f(b, b) \rightarrow c$$

$$f(a, a) \rightarrow f(a, b) \rightarrow f(b, b) \rightarrow c$$

$$f(a, a) \rightarrow f(a, b) \rightarrow c$$

Neededness - Alternative Definition

A redex $r : M \rightarrow N$ is **needed** iff

$$\forall P \forall e : N \rightarrow P, |\text{std}(r; e)| > |\text{std}(e)|$$

- This definition coincides with the previous one
- Its appeal: allows generalization to needed **derivations**

1 Residuals

2 Standardization

3 Needed Strategies

- Needed Redexes
- Needed Redexes and Standardization
- Neededness for Non-Orthogonal Systems

Needed Derivations

We generalize our previous notion of needed redexes to needed **derivations**

A derivation $d : M \rightarrow N$ is **needed** if

$$\forall P \forall e : N \rightarrow P, |\text{std}(d; e)| > |\text{std}(e)|$$

A **needed strategy** is a (multi-step) strategy \mathbb{F} s.t. $\forall M$

$$M \rightarrow^+ \mathbb{F}(M) \text{ is a needed derivation}$$

Needed Derivations

In Non-Orthogonal TRS needed **redexes** may not exist (as already seen)
But needed **derivations** always do!

Prop.

Every standard, normalising derivation is needed

Proof

Immediate from definition

Example

$$\begin{array}{lcl} a & \rightarrow & b \\ f(b, x) & \rightarrow & g(c) \\ f(x, b) & \rightarrow & g(c) \\ g(c) & \rightarrow & d \end{array}$$

Although a in $f(a, a)$ is not needed, it extends to a needed derivation

$$d : f(a, a) \rightarrow f(a, b) \rightarrow g(c)$$

External Redexes

One way of constructing needed derivations is by contracting external redexes

A redex is **external to a cointial derivation** if its residuals are not nested by other redexes in the course of the derivation. A redex is **external** if it is external to any derivation.

$$\begin{array}{l} a \quad \rightarrow \quad b \\ f(x, b) \rightarrow g(a) \end{array}$$

a is not external, a is in the term $f(a, a)$

Note: External redexes are needed (the converse does not hold)

Finite Normalisation Cones

Idea:

- 1 Suppose there are only a finite number of different normalising derivations from M modulo Lévy permutation equivalence
- 2 Measure M by the longest such one
- 3 Show that needed derivations decrease this measure

Finite Normalisation Cones

A **normalisation cone** from M is a set $\{e_i^M : M \rightarrow P_i\}$ of normalising derivations s.t. for each normalising derivation $f : M \rightarrow N$, there exists a unique i , $f \equiv e_i$.

A TRS enjoys **finite normalisation cones** (FNC) when for any M there exists a **finite** normalisation cone for M .

Finite Normalisation Cone

Example

All OTRS: normalising derivations are unique modulo \equiv in that setting

Non-Example

Consider the TRS

$$a \rightarrow b$$

$$a \rightarrow a$$

and the derivations

$$a \rightarrow b$$

$$a \rightarrow a \rightarrow b$$

$$a \rightarrow a \rightarrow a \rightarrow b$$

...

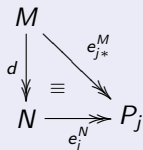
Normalisation Theorem II

Thm

Needed strategies normalise for TRS enjoying finite normalisation cones

Proof

- 1 Define $\text{depth}(M)$ to be the longest derivation in the finite normalisation cone of M : $\{e_i^M : M \twoheadrightarrow P_i\}$ (each e_i^M may be assumed standard)
- 2 Show that if $d : M \twoheadrightarrow N$ is a needed derivation, then $\text{depth}(M) > \text{depth}(N)$



$$|e_{j*}^M| = |\text{std}(d; e_j^N)| > |e_j^N|$$

How do we use this result?

- Find classes of TRS that enjoy FNC
- As mentioned, all OTRS do (normalising cones are not only finite, they are singletons)
- But, what about non-orthogonal TRS?

How do we use this result?

- Weakly OTRS (admit trivial critical pairs)? No (van Oostrom)

$$\begin{array}{lcl} a & \rightarrow & f(a) \\ f(b) & \rightarrow & b \\ f(x) & \rightarrow & b \end{array}$$

- Even though FNC fails already for weakly OTRS, the story is different for calculi with explicit substitutions
- **Next talk:** We'll spell out the details
- **Problem:** Characterize interesting classes of TRS that satisfy FNC

- Neededness and Normalisation Theorem I: [Huet,Lévy1991]
- Normalisation Theorem II (i.e. extension to non-orthogonal case): [Melliès1996,2000]
 - ▶ He developed the results in an **axiomatic rewriting framework** and in terms of 2-categorical models of rewriting
 - ▶ This framework is syntax free (i.e. independent of the structure of rewritten objects)
 - ▶ Many rewriting formats are thus captured