Advanced Topics in Termination

ISR 2009 – lecture 4

Aart Middeldorp

Institute of Computer Science
University of Innsbruck

Topics

- polynomial interpretations
- matrix interpretations
- dependency pairs
- match-bounds
- semantic labeling
- derivational complexity
- decidable classes
- certification
- applications

Further Reading

- http://cl-informatik.uibk.ac.at/~ami/09isr/

Outline

- Derivational Complexity
  - Polynomial Interpretations
  - Context Dependent Interpretations
  - Triangular Matrix Interpretations
- Decidable Classes
- Certification
- Applications
- Further Reading

Definitions

- derivation length $\mathrm{dl}(t) = \max\{\, n \mid \exists\, u\colon t \to^n u \,\}$
- derivational complexity $\mathrm{dc}(n) = \max\{\, \mathrm{dl}(t) \mid |t| = n \,\}$

Example

rewrite rule $\mathsf{a}(\mathsf{b}(x)) \to \mathsf{b}(\mathsf{a}(x))$

term	derivation length	value
$\mathsf{a}(\mathsf{b}(\mathsf{b}(\mathsf{b}(\mathsf{c}))))$	3	6
$\mathsf{a}(\mathsf{a}(\mathsf{a}(\mathsf{b}(\mathsf{b}(\mathsf{b}(\mathsf{c}))))))$	9	24
$\mathsf{a}^m(\mathsf{b}^n(\mathsf{c}))$	mn	$2^m n$

polynomial interpretation

$$\mathsf{a}_{\mathbb{N}}(x) = 2x \qquad \mathsf{b}_{\mathbb{N}}(x) = x + 1 \qquad \mathsf{c}_{\mathbb{N}} = 0$$

Theorem

	interpretation in $\mathbb{N}$	bound on derivational complexity
	polynomial	double-exponential
$a_1 x_1 + \cdots + a_n x_n + b$	linear	exponential
$x_1 + \cdots + x_n + b$	strongly linear	linear

Example

rewrite system

$$x + 0 \to x \qquad \mathsf{d}(0) \to 0 \qquad \mathsf{q}(0) \to 0$$
$$x + \mathsf{s}(y) \to \mathsf{s}(x + y) \qquad \mathsf{d}(\mathsf{s}(x)) \to \mathsf{s}(\mathsf{s}(\mathsf{d}(x))) \qquad \mathsf{q}(\mathsf{s}(x)) \to \mathsf{q}(x) + \mathsf{s}(\mathsf{d}(x))$$

interpretations

$$0_{\mathbb{N}} = 2 \quad \mathsf{s}_{\mathbb{N}}(x) = x + 1 \quad +_{\mathbb{N}}(x, y) = x + 2y \quad \mathsf{d}_{\mathbb{N}}(x) = 3x \quad \mathsf{q}_{\mathbb{N}}(x) = x^3$$

Outline

Definitions

- context dependent $\mathcal{F}$-algebra (CDA) $\mathcal{C}$ assign to every n-ary $f \in \mathcal{F}$
  - interpretation function $f_\mathcal{C} \colon \mathbb{R}_+ \times \mathbb{R}_0^n \to \mathbb{R}_0$
  - parameter functions $f_\mathcal{C}^i \colon \mathbb{R}_+ \to \mathbb{R}_+$ for $1 \leqslant i \leqslant n$
- Δ-assignment is mapping $\alpha \colon \mathbb{R}_+ \times \mathcal{V} \to \mathbb{R}_0$
- extension to terms

$$[\alpha,\Delta]_\mathcal{C}(t) = \begin{cases} \alpha(\Delta,t) & \text{if } t \in \mathcal{V} \\ f_\mathcal{C}(\Delta, [\alpha, f_\mathcal{C}^1(\Delta)]_\mathcal{C}(t_1), \dots, [\alpha, f_\mathcal{C}^n(\Delta)]_\mathcal{C}(t_n)) & \\ & \text{if } t = f(t_1,\dots,t_n) \end{cases}$$

- relation $>_\mathcal{C}$ on terms:
  $s >_\mathcal{C} t$ if $[\alpha,\Delta]_\mathcal{C}(s) >_\Delta [\alpha,\Delta]_\mathcal{C}(t)$ for all $\Delta \in \mathbb{R}_+$ and Δ-assignments α

Definition

CDA $\mathcal{C}$ is Δ-monotone if

$$f_\mathcal{C}(\Delta, a_1, \dots, a_i, \dots, a_n) >_\Delta f_\mathcal{C}(\Delta, a_1, \dots, b, \dots, a_n)$$

for all $\Delta \in \mathbb{R}_+$, $a_1,\dots,a_n, b \in \mathbb{R}_0$ and $i \in \{1,\dots,n\}$ with $a_i >_{f_\mathcal{C}^i(\Delta)} b$

Theorem

if $\mathcal{R} \subseteq >_\mathcal{C}$ for Δ-monotone CDA $\mathcal{C}$ then $\mathcal{R}$ is terminating and

$$\mathrm{dl}(t) \leqslant \liminf_{\Delta \in \mathbb{R}_+} \frac{[\alpha,\Delta]_\mathcal{C}(t)}{\Delta}$$

for all terms t

Example

rewrite rule $\mathsf{a}(\mathsf{b}(x)) \to \mathsf{b}(\mathsf{a}(x))$

CDA $\mathcal{C}$

$$\mathsf{a}_{\mathcal{C}}[\Delta](x) = (1+\Delta)x \qquad \mathsf{a}^1_{\mathcal{C}}(\Delta) = \frac{\Delta}{1+\Delta}$$

$$\mathsf{b}_{\mathcal{C}}[\Delta](x) = x+1 \qquad \mathsf{b}^1_{\mathcal{C}}(\Delta) = \Delta$$

derivational complexity

- $\liminf_{\Delta \in \mathbb{R}_+} \frac{[\alpha,\Delta]_{\mathcal{C}}(\mathsf{a}^m(\mathsf{b}^n(\mathsf{c})))}{\Delta} = \liminf_{\Delta \in \mathbb{R}_+} (\frac{1}{\Delta} + n)m = nm = \mathsf{dl}(\mathsf{a}^m(\mathsf{b}^n(\mathsf{c})))$
- $[\alpha,\Delta]_{\mathcal{C}}(\mathsf{a}^m(\mathsf{b}^n(\mathsf{c}))) = (1+\Delta n)m$

Definition

Δ-linear interpretation is CDA $\mathcal{C}$ with

$$f_{\mathcal{C}}(\Delta, x_1, \ldots, x_n) = \sum_{i=1}^{n} (a_i + b_i\Delta)x_i + c + d\Delta \qquad f^i_{\mathcal{C}}(\Delta) = \frac{\Delta}{a_i + b_i\Delta}$$

Theorem

if $\mathcal{R} \subseteq >_{\mathcal{C}}$ *for* Δ*-linear interpretation* $\mathcal{C}$ *then* $\mathsf{dc}(n) = 2^{\mathcal{O}(n)}$

Example

rewrite rule $\mathsf{a}(\mathsf{b}(x)) \to \mathsf{b}(\mathsf{b}(\mathsf{a}(x)))$

Δ-linear interpretation $\mathcal{C}$

$$\mathsf{a}_{\mathcal{C}}[\Delta](x) = (2+2\Delta)x \qquad \mathsf{a}^1_{\mathcal{C}}(\Delta) = \frac{\Delta}{2+2\Delta}$$

$$\mathsf{b}_{\mathcal{C}}[\Delta](x) = x+1 \qquad \mathsf{b}^1_{\mathcal{C}}(\Delta) = \Delta$$

Definition

Δ-restricted interpretation is CDA $\mathcal{C}$ with

$$f_{\mathcal{C}}(\Delta, x_1, \ldots, x_n) = \sum_{i=1}^{n}(a_i + b_i\Delta)x_i + c + d\Delta \qquad f_{\mathcal{C}}^i(\Delta) = \frac{\Delta}{a_i + b_i\Delta}$$

and $a_i \in \{0, 1\}$

Theorem

if $\mathcal{R} \subseteq >_{\mathcal{C}}$ *for* Δ*-restricted interpretation* $\mathcal{C}$ *then* $\mathrm{dc}(n) = \mathcal{O}(n^2)$

Example

rewrite rule $\mathsf{a}(\mathsf{a}(x)) \to \mathsf{a}(\mathsf{b}(\mathsf{a}(x)))$

Δ-restricted interpretation $\mathcal{C}$

$$\mathsf{a}_{\mathcal{C}}[\Delta](x) = 2\Delta x + 2 \qquad \mathsf{a}_{\mathcal{C}}^1(\Delta) = \frac{1}{2} \qquad \mathsf{b}_{\mathcal{C}}[\Delta](x) = \Delta x \qquad \mathsf{b}_{\mathcal{C}}^1(\Delta) = 1$$

Outline

Definition

triangular matrix interpretation $\mathcal{M}$ over $\mathbb{N}$

- carrier of $\mathcal{M}$ is $\mathbb{N}^d$ with $d > 0$
- interpretations (for every n-ary f)

$$f_{\mathcal{M}}(\vec{x}_1, \dots, \vec{x}_n) = M_1\vec{x}_1 + \cdots + M_n\vec{x}_n + \vec{f}$$

  with

  - matrices $M_1, \dots, M_n \in \mathbb{N}^{d \times d}$ with

$$(M_i)_{1,1} \geqslant 1 \qquad (M_i)_{j,k} = 0 \text{ for all } j > k \qquad (M_i)_{j,j} \leqslant 1 \text{ for all } j$$

    for all $1 \leqslant i \leqslant n$
  - vector $\vec{f} \in \mathbb{N}^d$

- $(x_1, \dots, x_d)^{\mathsf{T}} > (y_1, \dots, y_d)^{\mathsf{T}} \iff x_1 > y_1 \wedge \bigwedge_{i=2}^{d} x_i \geqslant y_i$

Theorem

if $\mathcal{R} \subseteq >_{\mathcal{M}}$ for triangular matrix interpretation $\mathcal{M}$ of dimension d then

$$\mathsf{dc}(n) = \mathcal{O}(n^d)$$

Example

rewrite rules

$$\mathsf{a}(\mathsf{b}(x)) \to \mathsf{b}(\mathsf{a}(x)) \qquad \mathsf{c}(\mathsf{a}(x)) \to \mathsf{b}(\mathsf{c}(x)) \qquad \mathsf{c}(\mathsf{b}(x)) \to \mathsf{a}(\mathsf{c}(x))$$

triangular matrix interpretation

$$\mathsf{a}_{\mathcal{M}}(\vec{x}) = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix} \vec{x} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \qquad \mathsf{c}_{\mathcal{M}}(\vec{x}) = \begin{pmatrix} 1 & 1 & 3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \vec{x} + \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

$$\mathsf{b}_{\mathcal{M}}(\vec{x}) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix} \vec{x} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

Outline

- Derivational Complexity
- Decidable Classes
- Certification
- Applications
- Further Reading

Theorem (Matiyasevich and Sénizergues)

termination is undecidable for three-rule SRSs

Open Problem

is termination decidable for one-rule SRSs ?

Theorem (Dershowitz)

termination is decidable for right-ground TRSs

Definition

- term is shallow if variables appear only at depth 0 or 1
- rewrite rule $\ell \to r$ is right-shallow if r is shallow
- rewrite rule $\ell \to r$ is left-shallow if ℓ is shallow
- rewrite rule $\ell \to r$ is shallow if it is left- and right-shallow

Theorem

- *termination is decidable for left-linear shallow TRSs*
- *termination is decidable for right-linear right-shallow TRSs*
- *termination is undecidable for right-shallow TRSs*

Theorem

termination is undecidable for right-shallow TRSs

Proof

reduction from PCP

PCP instance $P = \{(u_1, v_1), \dots, (u_1, v_n)\} \subseteq \Gamma^* \times \Gamma^*$

TRS $\mathcal{R}_P$

$$\begin{aligned} \mathsf{f}(x) &\to \mathsf{g}(x,x,x) & \\ \mathsf{g}(x, u_i(y), v_i(z)) &\to \mathsf{h}(x,y,z) & \text{for } 1 \leqslant i \leqslant n \\ \mathsf{h}(x, u_i(y), v_i(z)) &\to \mathsf{h}(x,y,z) & \text{for } 1 \leqslant i \leqslant n \\ \mathsf{h}(x, \mathsf{c}, \mathsf{c}) &\to \mathsf{f}(x) & \end{aligned}$$

P has solution $\iff$ $\mathcal{R}_P$ is non-terminating

Definition

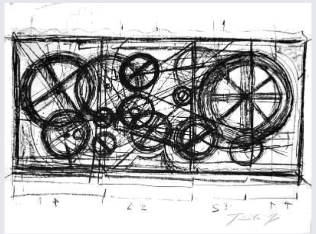
- term is shallow if variables appear only at depth 0 or 1
- rewrite rule $\ell \to r$ is right-shallow if r is shallow
- rewrite rule $\ell \to r$ is left-shallow if ℓ is shallow
- rewrite rule $\ell \to r$ is shallow if it is left- and right-shallow

Theorem

- *termination is decidable for left-linear shallow TRSs*
- *termination is decidable for right-linear right-shallow TRSs*
- *termination is undecidable for right-shallow TRSs*

Outline

Termination Tools



- termination tools are complex
- termination tools contain bugs
- termination tool output is unreadable
- termination tool output must be certified

Certification of Termination

- Coccinelle/C*i*ME
- CoLoR/Rainbow
- IsaFoR/CeTA
- PVS

Coccinelle/C*i*ME

- Coccinelle: formalization of rewriting in Coq
- shallow embedding
- C*i*ME: generates separate Coq proof for each TRS

CoLoR/Rainbow

- CoLoR: formalization of rewriting in Coq
- deep embedding
- Rainbow: generates separate Coq proof for each TRS

Problems

- calling proof assistant for every proof is slow
- error messages not readable since generated inside Coq
- goal is generated by tool (correct proof, but for wrong TRS)

IsaFoR/CeTA

- Isabelle Formalization of Rewriting
- Certified Termination Analysis

  input: TRS and proof (both in XML format)

  output: certified yes (if proof is correct), no (if proof is incorrect)
- CeTA is generated from IsaFoR, using Isabelle's code-generation facilities
- certification is fast
- error messages use terminology from rewriting

Example

$$\mathsf{add}(0, y) \to y \qquad \mathsf{add}(\mathsf{s}(x), y) \to \mathsf{s}(\mathsf{add}(x, y))$$

Coccinelle/C*i*ME (Ad Hoc Data Type)

```
Rule_0
 (Var 1)
 (Term id_add [Term id_0 []; Var 1])

Rule_1
 (Term id_s [Term id_add [Var 0; Var 1]])
 (Term id_add [Term id_s [Var 0]; Var 1])
```

Example

$\mathsf{add}(0, y) \to y \qquad \mathsf{add}(\mathsf{s}(x), y) \to \mathsf{s}(\mathsf{add}(x, y))$

CoLoR/Rainbow (List of Rules)

```
[Rule (add 0 (V 1)) (V 1);
 Rule (add (s (V 0)) (V 1)) (s (add (V 0) (V 1)))]
```

IsaFoR/CeTA (Set of Pairs of Terms)

```
{(Fun "add" [Fun "0" [], Var "y"], Var "y"),
 (Fun "add" [Fun "s" [Var "x"], Var "y"],
  Fun "s" [Fun "add" [Var "x", Var "y"]])}
```


Outline

Observation

programs can be translated to TRSs, DP problems, TRSs with built-in integers, . . .

Logic Programming

- pure logic programs
- Prolog programs

Functional Programming

- Haskell programs with starting expression

Imperative Programming

- Java bytecode

Further Information

- AProVE

Outline

Definition (Standard Completion)

set $\mathcal{E}$ of equations set $\mathcal{R}$ of rewrite rules reduction order $>$

inference system $\mathcal{SC}$ consists of six rules

delete	$\dfrac{\mathcal{E} \cup \{s \approx s\}, \mathcal{R}}{\mathcal{E}, \mathcal{R}}$	
compose	$\dfrac{\mathcal{E}, \mathcal{R} \cup \{s \to t\}}{\mathcal{E}, \mathcal{R} \cup \{s \to u\}}$	if $t \to_{\mathcal{R}} u$
simplify	$\dfrac{\mathcal{E} \cup \{s \dot{\approx} t\}, \mathcal{R}}{\mathcal{E} \cup \{s \approx u\}, \mathcal{R}}$	if $t \to_{\mathcal{R}} u$
orient	$\dfrac{\mathcal{E} \cup \{s \approx t\}, \mathcal{R}}{\mathcal{E}, \mathcal{R} \cup \{s \to t\}}$	if $s > t$
collapse	$\dfrac{\mathcal{E}, \mathcal{R} \cup \{t \to s\}}{\mathcal{E} \cup \{u \approx s\}, \mathcal{R}}$	if $t \to_{\mathcal{R}} u$ using $l \to r \in \mathcal{R}$ with $t \rhd l$
deduce	$\dfrac{\mathcal{E}, \mathcal{R}}{\mathcal{E} \cup \{s \approx t\}, \mathcal{R}}$	if $s \leftarrow_{\mathcal{R}} u \to_{\mathcal{R}} t$

Definitions

completion procedure is program that takes as input set of equations $\mathcal{E}$ and reduction order $>$ and generates (finite or infinite) run

$$(\mathcal{E}_0, \mathcal{R}_0) \vdash_{\mathcal{SC}} (\mathcal{E}_1, \mathcal{R}_1) \vdash_{\mathcal{SC}} (\mathcal{E}_2, \mathcal{R}_2) \vdash_{\mathcal{SC}} \cdots$$

with $\mathcal{E}_0 = \mathcal{E}$ and $\mathcal{R}_0 = \varnothing$

- $\mathcal{E}_\omega$ is set of persistent equations: $\mathcal{E}_\omega = \bigcup_{i \geqslant 0} \bigcap_{j \geqslant i} \mathcal{E}_j$
- $\mathcal{R}_\omega$ is set of persistent rules
- run succeeds if $\mathcal{E}_\omega = \varnothing$ and $\mathcal{R}_\omega$ is confluent and terminating
- run fails if $\mathcal{E}_\omega \neq \varnothing$

Definition

completion procedure it correct if every run that does not fail succeeds

Definitions

- run $(\mathcal{E}_0, \mathcal{R}_0) \vdash_{\mathcal{SC}} (\mathcal{E}_1, \mathcal{R}_1) \vdash_{\mathcal{SC}} (\mathcal{E}_2, \mathcal{R}_2) \vdash_{\mathcal{SC}} \cdots$ is fair if

$$\mathsf{CP}(\mathcal{R}_\omega) \subseteq \bigcup_{i \geqslant 0} \mathcal{E}_i$$

- completion procedure is fair if every run that does not fail is fair

Theorem

every fair completion procedure is correct

Remark

reduction order is used to determine orientation of rewrite rules

Definition (Standard Completion with Termination Tools)

inference system $\mathcal{SC}tt$ consists of six rules

delete $\dfrac{\mathcal{E} \cup \{s \approx s\}, \mathcal{R}, \mathcal{C}}{\mathcal{E}, \mathcal{R}, \mathcal{C}}$

compose $\dfrac{\mathcal{E}, \mathcal{R} \cup \{s \to t\}, \mathcal{C}}{\mathcal{E}, \mathcal{R} \cup \{s \to u\}, \mathcal{C}}$ if $t \to_{\mathcal{R}} u$

simplify $\dfrac{\mathcal{E} \cup \{s \approx t\}, \mathcal{R}, \mathcal{C}}{\mathcal{E} \cup \{s \approx u\}, \mathcal{R}, \mathcal{C}}$ if $t \to_{\mathcal{R}} u$

orient $\dfrac{\mathcal{E} \cup \{s \approx t\}, \mathcal{R}, \mathcal{C}}{\mathcal{E}, \mathcal{R} \cup \{s \to t\}, \mathcal{C} \cup \{s \to t\}}$ if $\mathcal{C} \cup \{s \to t\}$ is terminating

collapse $\dfrac{\mathcal{E}, \mathcal{R} \cup \{t \to s\}, \mathcal{C}}{\mathcal{E} \cup \{u \approx s\}, \mathcal{R}, \mathcal{C}}$ if $t \to_{\mathcal{R}} u$ using $l \to r \in \mathcal{R}$ with $t \rhd l$

deduce $\dfrac{\mathcal{E}, \mathcal{R}, \mathcal{C}}{\mathcal{E} \cup \{s \approx t\}, \mathcal{R}, \mathcal{C}}$ if $s \leftarrow_{\mathcal{R}} u \to_{\mathcal{R}} t$

Problem

if $\mathcal{E}, \varnothing \vdash^*_{\mathcal{SC}tt} \varnothing, \mathcal{R}$ is fair then $\mathcal{R}$ need not be confluent

details

Theorem

- $\mathcal{E}, \varnothing, \varnothing \vdash^*_{\mathcal{SC}tt} \varnothing, \mathcal{R}, \mathcal{C} \implies \mathcal{E}, \varnothing \vdash^*_{\mathcal{SC}} \varnothing, \mathcal{R}$ *with respect to* $> = \to^+_{\mathcal{C}}$
- $\mathcal{E}, \varnothing, \varnothing \vdash^*_{\mathcal{SC}tt} \varnothing, \mathcal{R}, \mathcal{C} \impliedby \mathcal{E}, \varnothing \vdash^*_{\mathcal{SC}} \varnothing, \mathcal{R}$

Remarks

- $\mathcal{SC}tt$ is implemented in Slothrop
- completion of CGE_2

$$\begin{aligned} \mathsf{e} \cdot x &\approx x & \mathsf{f}(x \cdot y) &\approx \mathsf{f}(x) \cdot \mathsf{f}(y) \\ x^- \cdot x &\approx \mathsf{e} & \mathsf{g}(x \cdot y) &\approx \mathsf{g}(x) \cdot \mathsf{g}(y) \\ (x \cdot y) \cdot z &\approx x \cdot (y \cdot z) & \mathsf{f}(x) \cdot \mathsf{g}(y) &\approx \mathsf{g}(y) \cdot \mathsf{f}(x) \end{aligned}$$

  is Slothrop's defining achievement
- CGE_2 cannot be completed by state-of-the-art implementations of $\mathcal{SC}$ like Waldmeister
- multi-completion with termination tools is more efficient (MKBtt)

Outline

- Termination Proofs and the Length of Derivations
  Dieter Hofbauer and Clemens Lautemann
  Proc. 3rd RTA, LNCS 355, pp. 167 – 177, 1989
- Termination Proofs by Context-Dependent Interpretations
  Dieter Hofbauer
  Proc. 12th RTA, LNCS 2051, pp. 108 – 121, 2001
- Proving Quadratic Derivational Complexities Using Context Dependent Interpretations
  Georg Moser and Andreas Schnabl
  Proc. 18th RTA, LNCS 5117, pp. 276 – 290, 2008
- Complexity Analysis of Term Rewriting Based on Matrix and Context Dependent Interpretations
  Georg Moser, Andreas Schnabl and Johannes Waldmann
  Proc. 28th FSTTCS, LIPI 2, pp. 304 – 315, 2008
- Termination of Rewriting with Right-Flat Rules
  Guillem Godoy, Eduard Huntingford and Ashish Tiwari
  Proc. 18th RTA, LNCS 4533, pp. 200 – 213, 2007

- Certification of Termination Proofs using CeTA
  René Thiemann and Christian Sternagel
  Proc. 22nd TPHOLs, LNCS 5674, 2009, to appear
- Automated Termination Proofs for Logic Programs by Term Rewriting
  Peter Schneider-Kamp, Jürgen Giesl, Alexander Serebrenik and René Thiemann
  ACM ToCL, 2009, to appear
- Automated Termination Analysis for Haskell: From Term Rewriting to Programming Languages
  Jürgen Giesl, Stephan Swiderski, Peter Schneider-Kamp and René Thiemann
  Proc. 17th RTA, LNCS 4098, pp. 297 – 312, 2006
- Slothrop: Knuth-Bendix Completion with a Modern Termination Checker
  Ian Wehrman, Aaron Stump and Edwin Westbrook
  Proc. 17th RTA, LNCS 4098, pp. 287 – 296, 2006
- Multi-Completion with Termination Tools (System Description)
  Haruhiko Sato, Sarah Winkler, Masahito Kurihara and Aart Middeldorp
  Proc. 4th IJCAR, LNAI 5195, pp. 306 – 312, 2008

http://cl-informatik.uibk.ac.at/~ami/09isr/

- slides (with active hyperlinks)
- papers
- tools
- ISR 2009 version of $\mathsf{T_TT_2}$

Problem

if $\mathcal{E}, \varnothing \vdash^*_{\mathcal{SC}tt} \varnothing, \mathcal{R}$ is fair then $\mathcal{R}$ need not be confluent

return

Example

1

$$\begin{array}{rclcrclcrcl} 129 & \to & 1 & & 127 & \to & 128 & & 34 & \to & 54 \\ 14 & \to & 4 & & 284 & \to & 294 & & 26 & \to & 27 \\ 254 & \to & 274 & & & & & & 23 & \to & 29 \end{array}$$

- LPO with precedence $3 > 5 > 6 > 7 > 8 > 9$
- confluent and terminating

2

$$\begin{array}{rclcrclcrcl} 123 & \to & 1 & & 127 & \leftarrow & 128 & & 34 & \to & 54 \\ 14 & \to & 4 & & 284 & \to & 234 & & 26 & \to & 27 \\ 254 & \to & 274 & & & & & & 23 & \leftarrow & 29 \end{array}$$

- LPO with precedence $8 > 9 > 3 > 5 > 6 > 7$
- terminating but not confluent: $1274 \leftarrow 1254 \leftarrow 1234 \to 14 \to 4$