Argument Filterings and Usable Rules for Simply Typed Depenency Pairs

Toshiyuki Yamada

Mie University

joint work with

Takahito Aoto

Tohoku University

Simply typed TRS [Yamada, RTA '01]

- natural extension of first-order TRS
- higher-order functions are available
- no bound variable

$$
\begin{array}{lcl}
(+\ \mathsf{0})\ y & \rightarrow & y \\
(+\ (\mathsf{s}\ x))\ y & \rightarrow & \mathsf{s}\ ((+\ x)\ y) \\
(\mathsf{fold}\ F\ x)\ [\,] & \rightarrow & x \\
(\mathsf{fold}\ F\ x)\ (:\ y\ ys) & \rightarrow & (F\ y)\ ((\mathsf{fold}\ F\ x)\ ys) \\
\mathsf{sum} & \rightarrow & \mathsf{fold}\ +\ \mathsf{0}
\end{array}
$$

➜ termination proof techniques ?

Related works

termination for higher-order rewriting without bound variable

- [Linfantsev–Bachmair, TPHOL '98] path ordering
- [Yamada, RTA '01] interpretation
- [Kusakari, IPSJ '01] path ordering, dependency pairs + filtering
- [Kusakari, IPSJ '03] path ordering
- [Aoto–Yamada, RTA '03] forst-ordering encoding + labelling
- [Toyama, RTA '04] path ordering
- [Aoto–Yamada, RTA '05] dependency pairs + subterm criterion
- [Hirokawa–Middeldorp, HOR '05] first-ordering encoding
- [Toyama, RTA '08] path ordering

Today's talk

automatic termination proof techniques for simply typed TRSs

- argument filtering
- usable rules

➜ powerful and efficient termination proof (in modular way)

Outline

1. introduction
2. simply typed term rewriting
3. dependency pairs
4. argument filtering & usable rules
5. experiments
6. conclusiton

Outline

Types

$$T \;::=\; o \;\mid\; T \times \cdots \times T \to T$$

$x \quad y \quad ys \quad \mathsf{0} \quad [\,]$	o
$\mathsf{s} \quad \mathsf{sum}$	$o \to o$
$:$	$o \times o \to o$
$F \quad +$	$o \to o \to o$
fold	$(o \to o \to o) \times o \to o \to o$

Terms

$$\frac{t \in \Sigma_\tau \cup V_\tau}{t : \tau}$$

$$\frac{t : \tau_1 \times \cdots \times \tau_n \to \tau \quad t_1 : \tau_1 \quad \cdots \quad t_n : \tau_n}{(t\ t_1\ \cdots\ t_n) : \tau}$$

$$\mathsf{s}\ ((+\ x)\ y) \qquad (\mathsf{fold}\ F\ x)\ (:\ y\ ys)$$

Rewrite system

$$
\begin{array}{lcl}
(+\ 0)\ y & \rightarrow & y \\
(+\ (\mathsf{s}\ x))\ y & \rightarrow & \mathsf{s}\ ((+\ x)\ y) \\
(\mathsf{fold}\ F\ x)\ [\,] & \rightarrow & x \\
(\mathsf{fold}\ F\ x)\ (:\ y\ ys) & \rightarrow & (F\ y)\ ((\mathsf{fold}\ F\ x)\ ys) \\
\mathsf{sum} & \rightarrow & \mathsf{fold}\ +\ 0
\end{array}
$$

Rewrite sequence

$$
\begin{array}{ll}
 & \mathsf{sum}\ (:\ (\mathsf{s}\ 0)\ [\,]) \\
\rightarrow & (\mathsf{fold}\ +\ 0)\ (:\ (\mathsf{s}\ 0)\ [\,]) \\
\rightarrow & (+\ (\mathsf{s}\ 0))\ ((\mathsf{fold}\ +\ 0)\ [\,]) \\
\rightarrow & (+\ (\mathsf{s}\ 0))\ 0 \\
\rightarrow & \mathsf{s}\ ((+\ 0)\ 0) \\
\rightarrow & \mathsf{s}\ 0
\end{array}
$$

Basic property of rewrite chains

infinite head rewrite steps are obained from any rewrite chain by selecting innermost non-terminating subterms

$$\left\{ \begin{array}{lcl} \mathsf{count1}\ x & \rightarrow & \mathsf{s}\ (\mathsf{count2}\ x) \\ \mathsf{count2}\ x & \rightarrow & \mathsf{s}\ (\mathsf{count1}\ (\mathsf{s}\ x)) \end{array} \right\}$$

Basic property of rewrite chains

infinite head rewrite steps are obained from any rewrite chain by selecting innermost non-terminating subterms

$$\left\{ \begin{array}{lcl} \mathsf{count1}\ x & \to & \mathsf{s}\ (\mathsf{count2}\ x) \\ \mathsf{count2}\ x & \to & \mathsf{s}\ (\mathsf{count1}\ (\mathsf{s}\ x)) \end{array} \right\}$$

$$\underline{\mathsf{count1}\ 0} \xrightarrow{h} \mathsf{s}\ (\underline{\mathsf{count2}\ 0}) \to \mathsf{s}^2\ (\underline{\mathsf{count1}\ (\mathsf{s}\ 0)}) \to \mathsf{s}^3\ (\underline{\mathsf{count2}\ (\mathsf{s}\ 0)}) \to \cdots$$

$$\triangledown$$

$$\underline{\mathsf{count2}\ 0} \xrightarrow{h} \mathsf{s}\ (\underline{\mathsf{count1}\ (\mathsf{s}\ 0)})$$

$$\triangledown$$

$$\underline{\mathsf{count1}\ (\mathsf{s}\ 0)} \xrightarrow{h} \mathsf{s}\ (\underline{\mathsf{count2}\ (\mathsf{s}\ 0)})$$

$$\triangledown$$

$$\underline{\mathsf{count2}\ (\mathsf{s}\ 0)} \xrightarrow{h} \cdots$$

$\xrightarrow{h}$ captures change of leading symbols

Head rewrite step

Definition

$$s \xrightarrow{\text{h}} t \quad :\Longleftrightarrow \quad s = l\sigma,\ \ t = r\sigma \quad \text{for some } \ l \rightarrow r,\ \ \sigma \quad \text{or}$$

$$s = (s_0\ s_1\ \cdots\ s_n),\ \ t = (t_0\ s_1\ \cdots\ s_n),\quad s_0 \xrightarrow{\text{h}} t_0$$

$$\xrightarrow{\text{nh}} \quad := \quad \rightarrow \setminus \xrightarrow{\text{h}}$$

$$\underline{\mathsf{sum}}\ (\,:\ \underline{((\mathsf{fold} + 0)\ [\,])}\ (\,:\ \underline{((\mathsf{fold} + 0)\ [\,])}\ [\,]))$$

Characterising termination

Definition

$\mathrm{NT}_{\mathrm{min}}$ ... set of minimal (wrt. $\trianglerighteq$) non-terminating terms

Lemma

$\forall\, s \in \mathrm{NT}_{\mathrm{min}}\ \exists\, t \in \mathrm{NT}_{\mathrm{min}} \quad s \xrightarrow{\mathrm{nh}}{}^{*} \cdot \xrightarrow{\mathrm{h}} \cdot \trianglerighteq t$

➜ every minimal non-terminating term admits $\xrightarrow{\mathrm{nh}}{}^{*} \cdot \xrightarrow{\mathrm{h}} \cdot \trianglerighteq$ -chain

Corollary

$\rightarrow$ is terminating $\iff$ $\xrightarrow{\mathrm{nh}}{}^{*} \cdot \xrightarrow{\mathrm{h}} \cdot \trianglerighteq$ (on $\mathrm{NT}_{\mathrm{min}}$) is terminating

Outline

1. introduction
2. simply typed term rewriting
3. dependency pairs
4. argument filtering & usable rules
5. experiments
6. conclusiton

Dependency pairs [Arts–Giesl, TCS ’00]

Key idea ... tracing head defined subterms in reduction

$$\mathrm{DP}(l \to r) \;:=\; \{\, (l,\ r') \mid r' \trianglelefteq r,\ r' \text{ is head defined},\ r' \ntrianglelefteq l \,\}$$

Dependency pairs [Arts–Giesl, TCS '00]

Key idea ... tracing head defined subterms in reduction

$$\mathrm{DP}(l \to r) \;:=\; \{\, (l,\ r') \mid r' \trianglelefteq r,$$

r' is head defined,

$$r' \not\trianglelefteq l \,\}$$

$$\begin{array}{lcl}
+\ 0\ y & \longrightarrow & y \\
\underline{+\ (\mathsf{s}\ x)\ y} & \longrightarrow & \mathsf{s}\ (\underline{+\ x\ y}) \\
\times\ 0\ y & \longrightarrow & 0 \\
\underline{\times\ (\mathsf{s}\ x)\ y} & \longrightarrow & \underline{+\ (\underline{\times\ x\ y})\ y}
\end{array}$$

Simply typed dependency pairs [Aoto–Yamada, RTA '05]

Definition

$$\mathrm{DP}(l \to r) \;:=\; \{\, l \rightarrowtail r' \mid r' \trianglelefteq r,\ \ r' \text{ is head defined},\ r' \ntriangleleft l \,\}$$
$$\cup \quad \{\, l' \rightarrowtail r' \in \mathrm{Exp}(l \to r) \mid r' \text{ is head defined} \,\}$$

- head-variable subterm considered to be head defined
- rule of function type should be expanded $\mathrm{Exp}(l \to r)$

Simply typed dependency pairs (example)

$$\mathcal{R} \qquad \begin{array}{lcl} (+\ 0)\ y & \longrightarrow & y \\ \underline{(+\ (\mathsf{s}\ x))\ y} & \longrightarrow & \mathsf{s}\ \underline{(\underline{(+\ x)}\ y)} \\ (\mathsf{fold}\ F\ x)\ [\,] & \longrightarrow & x \\ \underline{\underline{(\mathsf{fold}\ F\ x)\ (:\ y\ ys)}} & \longrightarrow & \underline{\underline{(F\ y)}\ \underline{((\mathsf{fold}\ F\ x)\ ys)}} \\ \underline{\mathsf{sum}} & \longrightarrow & \underline{\underline{\mathsf{fold}}\ \underline{+}\ 0} \end{array}$$

Simply typed dependency pairs (example)

$$
\mathcal{R} \quad
\begin{array}{lcl}
(+\ 0)\ y & \rightarrow & y \\
(+\ (\mathsf{s}\ x))\ y & \rightarrow & \mathsf{s}\ ((+\ x)\ y) \\
(\mathsf{fold}\ F\ x)\ [\,] & \rightarrow & x \\
(\mathsf{fold}\ F\ x)\ (:\ y\ ys) & \rightarrow & (F\ y)\ ((\mathsf{fold}\ F\ x)\ ys) \\
\mathsf{sum} & \rightarrow & \mathsf{fold}\ +\ 0
\end{array}
$$

$$
\mathrm{DP}(\mathcal{R}) \quad
\begin{array}{lcl}
(+\ (\mathsf{s}\ x))\ y & \rightarrowtail & (+\ x)\ y \\
(+\ (\mathsf{s}\ x))\ y & \rightarrowtail & +\ x \\
(\mathsf{fold}\ F\ x)\ (:\ y\ ys) & \rightarrowtail & (F\ y)\ ((\mathsf{fold}\ F\ x)\ ys) \\
(\mathsf{fold}\ F\ x)\ (:\ y\ ys) & \rightarrowtail & F\ y \\
(\mathsf{fold}\ F\ x)\ (:\ y\ ys) & \rightarrowtail & (\mathsf{fold}\ F\ x)\ ys \\
\mathsf{sum} & \rightarrowtail & \mathsf{fold}\ +\ 0 \\
\mathsf{sum} & \rightarrowtail & \mathsf{fold} \\
\mathsf{sum} & \rightarrowtail & + \\
\mathsf{sum}\ x & \rightarrowtail & (\mathsf{fold}\ +\ 0)\ x
\end{array}
$$

Termination by dependency pairs

Definition

$s \; \rightarrowtail_{D} \; t \quad :\Longleftrightarrow \quad (s,t)$ is instance of some DP in D

Termination by dependency pairs

Definition

$s \ \rightarrowtail_D \ t \quad :\Longleftrightarrow \quad (s,t)$ is instance of some DP in D

Theorem

following statements are equivallent:

- $\rightarrow_{\mathcal{R}}$ is terminating
- $\xrightarrow{\text{nh}}{}^{*} \cdot \xrightarrow{\text{h}} \cdot \trianglerighteq$ (on NT_{min}) is terminating
- $\xrightarrow{\text{nh}}{}^{*} \cdot \rightarrowtail_{\text{DP}(\mathcal{R})}$ (on NT_{min}) is terminating

Dependency graph [Arts–Giesl, TCS '00]

Key idea . . . approximating $\stackrel{\mathrm{nh}}{\rightarrow}{}^{*} \cdot \rightarrowtail_{D}$ -sequence
by path in (finite) graph

Definition (same as first-order case)

vertexes . . . set of all dependency pairs

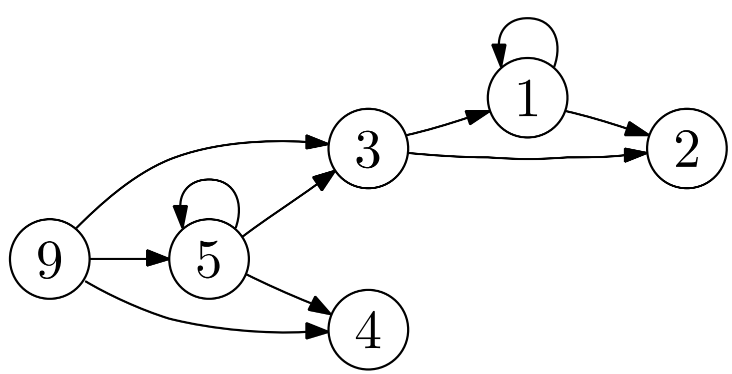
edges . . . edge from $l \rightarrowtail r$ to $l' \rightarrowtail r'$ exists
if and only if $r\sigma \stackrel{\mathrm{nh}}{\rightarrow}{}^{*} l'\sigma'$ for some σ, σ'

Dependency graph (example)

dependency pairs

(1)	$(+\ (\mathsf{s}\ x))\ y$	$\rightarrowtail$	$(+\ x)\ y$
(2)	$(+\ (\mathsf{s}\ x))\ y$	$\rightarrowtail$	$+\ x$
(3)	$(\mathsf{fold}\ F\ x)\ (:\ y\ ys)$	$\rightarrowtail$	$(F\ y)\ ((\mathsf{fold}\ F\ x)\ xs)$
(4)	$(\mathsf{fold}\ F\ x)\ (:\ y\ ys)$	$\rightarrowtail$	$F\ y$
(5)	$(\mathsf{fold}\ F\ x)\ (:\ y\ ys)$	$\rightarrowtail$	$(\mathsf{fold}\ F\ x)\ ys$
(6)	sum	$\rightarrowtail$	$\mathsf{fold}\ +\ \mathsf{0}$
(7)	sum	$\rightarrowtail$	fold
(8)	sum	$\rightarrowtail$	$+$
(9)	$\mathsf{sum}\ x$	$\rightarrowtail$	$(\mathsf{fold}\ +\ \mathsf{0})\ x$

dependency graph



Subterm criterion [Hirokawa–Middeldorp, RTA '04]

Key idea . . . for every DP on cycle of DG

projection π selects one argument of defined symbol

to obtain ordering constraint w.r.t. subterm relation $\rhd$

simple projection $\pi : \Sigma_{\text{def}} \to \mathbb{N}_+$

Subterm criterion [Hirokawa–Middeldorp, RTA '04]

Key idea ... for every DP on cycle of DG

projection π selects one argument of defined symbol

to obtain ordering constraint w.r.t. subterm relation $\rhd$

simple projection $\pi : \Sigma_{\mathrm{def}} \to \mathbb{N}_+$

$\pi(+) = 1$ ... select 1st argument for $+$

$$(+\ (\mathsf{s}\ x))\ y \quad \overset{\pi}{\longmapsto} \quad (+\ (\mathsf{s}\ x))\ y$$

$\pi(\mathsf{fold}) = 3$ ... select 3rd argument for fold

$$(\mathsf{fold}\ F\ x)\ (:\ y\ ys) \quad \overset{\pi}{\longmapsto} \quad (\mathsf{fold}\ F\ x)\ (:\ y\ ys)$$

Termination by subterm criterion

Theorem [Aoto–Yamada, RTA '05]

- $\mathcal{R}$: finite simply typed TRS
- $\forall D$: set of DPs admitting cycle in DG
  $\exists\ \pi$: simple projection for D
  s.t. $\pi(D) \subseteq \trianglerighteq$ and $\pi(D) \cap \triangleright \neq \emptyset$
  (π satisfies subterm criterion for D)

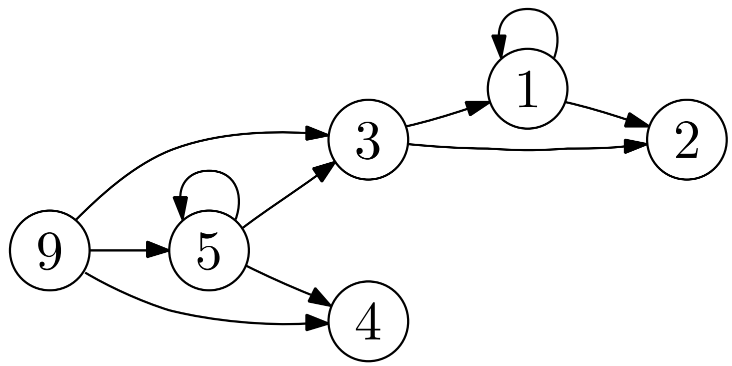
$\Longrightarrow$ no $\xrightarrow{\text{nh}}_{*} \cdot \rightarrowtail_{\mathrm{DP}(\mathcal{R})}$ -chain   no $(\mathcal{R}, \mathrm{DP}(\mathcal{R}))$ chain

$\Longrightarrow$ $\mathcal{R}$ is terminating

note: ordering constraints only on DPs

Termination by subterm criterion (example)

dependency graph (for running example)



dependency pairs admitting cycle

$$\begin{array}{llcl} (1) & (+\ (\mathsf{s}\ x))\ y & \rightarrowtail & (+\ x)\ y \\ (5) & (\mathsf{fold}\ F\ x)\ (:\ y\ ys) & \rightarrowtail & (\mathsf{fold}\ F\ x)\ ys \end{array}$$

simple projection satisfying subterm criterion: $\pi(+) := 1,\ \pi(\mathsf{fold}) := 3$

$$\begin{array}{llcl} \pi(\ (1)\ ) \subseteq \rhd & (+\ (\mathsf{s}\ x))\ y & \rhd & (+\ x)\ y \\ \pi(\ (5)\ ) \subseteq \rhd & (\mathsf{fold}\ F\ x)\ (:\ y\ ys) & \rhd & (\mathsf{fold}\ F\ x)\ ys \end{array}$$

Outline

1. introduction
2. simply typed term rewriting
3. dependency pairs
4. argument filtering & usable rules
5. experimental results
6. conclusiton

Argument filtering [Arts–Giesl, TCS '00]

Key idea . . . simplify ordering constraints

by selecting/dropping arguments recursively

$$\pi : \Sigma \longrightarrow \mathbb{N}_+ \cup \mathrm{List}(\mathbb{N}_+)$$

$$\pi(\neg) = 1 \quad \pi(\wedge) = [1,2] \quad \pi(\vee) = [2]$$

$$\neg\ (\neg\ x) \quad \overset{\pi}{\longmapsto} \quad x$$

$$\neg\ (\wedge\ x\ y) \quad \overset{\pi}{\longmapsto} \quad \wedge\ x\ y$$

$$\neg\ (\vee\ x\ y) \quad \overset{\pi}{\longmapsto} \quad \vee\ y$$

Argument filtering for simply typed terms

Definition

$$\pi \ : \ (\Sigma \cup V_{\mathrm{fun}}) \times \mathbb{N} \ \rightarrow \ \mathbb{N} \cup \mathrm{List}(\mathbb{N})$$

$\pi(s, d)$ ... argument filtering for symbol s at depth d

Examples

$$s = \underline{\ _1(\ \underline{_0(\text{twice } U)}\ \ \underline{_0(H\ \ x)}\ )}$$

$0, 1 \ldots$ depth of twice and H

$\pi_1(\text{twice}, 1) = [0, 1]$ $\quad$ $\pi_2(\text{twice}, 1) = 0$ $\quad$ $\pi_3(\text{twice}, 1) = 1$

$\pi_1(\text{twice}, 0) = [0]$ $\quad$ $\pi_2(\text{twice}, 0) = 1$

$\pi_1(H, 0) = [1]$ $\quad$ $\pi_3(H, 0) = [\,]$

$\pi_1(s) = ((\text{twice})\ \ (x))$ $\quad$ $\pi_2(s) = U$ $\quad$ $\pi_3(s) = (\,)$

note: filtered term may be ill-typed (use S-expression [Toyama, RTA '05])

Termination by argument filtering (example)

simply typed TRS

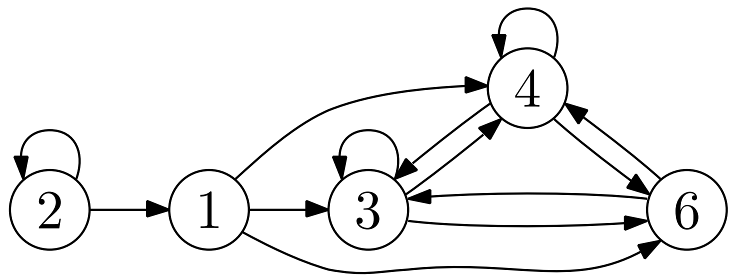
$$
\begin{array}{lcl}
\mathsf{map}\ G\ [\,] & \to & [\,] \\
\mathsf{map}\ G\ (:\ x\ xs) & \to & :\ (G\ x)\ (\mathsf{map}\ G\ xs) \\
(\circ\ G\ H)\ x & \to & G\ (H\ x) \\
\mathsf{twice}\ G & \to & \circ\ G\ G
\end{array}
$$

dependency pairs

$$
\begin{array}{llcl}
(1) & \mathsf{map}\ G\ (:\ x\ xs) & \rightarrowtail & G\ x \\
(2) & \mathsf{map}\ G\ (:\ x\ xs) & \rightarrowtail & \mathsf{map}\ G\ xs \\
(3) & (\circ\ G\ H)\ x & \rightarrowtail & G\ (H\ x) \\
(4) & (\circ\ G\ H)\ x & \rightarrowtail & H\ x \\
(5) & \mathsf{twice}\ G & \rightarrowtail & \circ\ G\ G \\
(6) & (\mathsf{twice}\ G)\ x & \rightarrowtail & (\circ\ G\ G)\ x \\
(7) & \mathsf{twice}\ G & \rightarrowtail & \circ
\end{array}
$$

Termination by argument filtering (example continued)

dependency graph



dependency pairs admitting cycle

(2)	map G $(:\ x\ xs)$	$\rightarrowtail$	map G xs
(3)	$(\circ\ G\ H)\ x$	$\rightarrowtail$	$G\ (H\ x)$
(4)	$(\circ\ G\ H)\ x$	$\rightarrowtail$	$H\ x$
(6)	(twice G) x	$\rightarrowtail$	$(\circ\ G\ G)\ x$

Termination by argument filtering (example continued)

simple projection for (2) satisfying subterm criterion: $\pi(\mathsf{map}) := 2$

$$\pi(\ (2)\ ) \subseteq \rhd \qquad \mathsf{:}\ x\ xs \rhd xs$$

head instantiated dependency pairs [Aoto–Yamada, RTA'05]

$$
\begin{array}{lllll}
(3a) & (\circ^\sharp\ (\circ\ U\ V)\ H)\ x & \rightarrowtail & (\circ^\sharp\ U\ V)\ (H\ x) \\
(3b) & (\circ^\sharp\ (\mathsf{twice}\ U)\ H)\ x & \rightarrowtail & (\mathsf{twice}^\sharp\ U)\ (H\ x) \\
(4a) & (\circ^\sharp\ G\ (\circ\ U\ V))\ x & \rightarrowtail & (\circ^\sharp\ U\ V)\ x \\
(4b) & (\circ^\sharp\ G\ (\mathsf{twice}\ U))\ x & \rightarrowtail & (\mathsf{twice}^\sharp\ U)\ x \\
(6') & (\mathsf{twice}^\sharp\ G)\ x & \rightarrowtail & (\circ^\sharp\ G\ G)\ x
\end{array}
$$

Termination by argument filtering (example continued)

argument filtering

$$\pi(G,0) = \pi(H,0) = \pi(\circ,1) = \pi(\mathsf{twice},1) = 1$$
$$\pi(\mathsf{twice},0) = \pi(\mathsf{twice}^\sharp,0) = \pi(\circ^\sharp,1) = [0,1]$$
$$\pi(\mathsf{map},0) = \pi(\,:,0) = \pi(\circ,0) = \pi(\circ^\sharp,0) = [0,1,2]$$

ordering constraints after filtering

$\mathsf{map}\ G\ [\,]$	$\succsim$	$[\,]$
$\mathsf{map}\ G\ (:\ x\ xs)$	$\succsim$	$:\ (G\ x)\ (\mathsf{map}\ G\ xs)$
$(\circ\ G\ H)\ x$	$\succsim$	$G\ (H\ x)$
$\mathsf{twice}\ G$	$\succsim$	$\circ\ G\ G$

$(\circ^\sharp\ (\circ\ U\ V)\ H)\ x$	$\succ$	$(\circ^\sharp\ U\ V)\ (H\ x)$
$(\circ^\sharp\ (\mathsf{twice}\ U)\ H)\ x$	$\succ$	$(\mathsf{twice}^\sharp\ U)\ (H\ x)$
$(\circ^\sharp\ G\ (\circ\ U\ V))\ x$	$\succ$	$(\circ^\sharp\ U\ V)\ x$
$(\circ^\sharp\ G\ (\mathsf{twice}\ U))\ x$	$\succ$	$(\mathsf{twice}^\sharp\ U)\ x$
$(\mathsf{twice}^\sharp\ G)\ x$	$\succ$	$(\circ^\sharp\ G\ G)\ x$

path ordering for S-expressions [Toyama, RTA'08] satisfies these

➜ $\mathcal{R}$ is terminating

Termination by argument filtering

Theorem (new)

- $\mathcal{R}$: finite simply typed TRS
- $D \subseteq \mathrm{DP}(\mathcal{R})$ and D : head-instantiated
- $(\succsim, \succ)$: reduction pair
- $\pi\ :\ \Sigma_{\mathrm{def}} \cup V_{\mathrm{fun}} \to \mathbb{N}_+ \cup \mathrm{List}(\mathbb{N}_+)$
  s.t. $\pi(\mathcal{R}) \subseteq \succsim$, $\pi(D^\sharp) \subseteq \succsim$ and π : stable w.r.t. $(\mathcal{R}, D)$
- no $(\mathcal{R}, D \setminus \{l \rightarrowtail r \in D \mid \pi(l^\sharp) \succ \pi(r^\sharp)\})$ chain

$\Longrightarrow$ no $(\mathcal{R}, D)$ chain

note: stability condition is essential for simply typed case

Unsound argument filtering

$$\begin{array}{llll} \mathcal{R} & \mathsf{f}\ (F\ x) & \to & \mathsf{f}\ (\mathrm{s}\ x) \\ \mathsf{DP}(\mathcal{R}) & \mathsf{f}\ (F\ x) & \rightarrowtail & \mathsf{f}\ (\mathrm{s}\ x) \end{array}$$

$$\mathsf{f}\ (\mathsf{s}\ x) \rightarrowtail_{\mathsf{DP}(\mathcal{R})} \mathsf{f}\ (\mathsf{s}\ x) \rightarrowtail_{\mathsf{DP}(\mathcal{R})} \cdots$$

dependency step is not preserved by filtering

$$\pi(\mathsf{f}, 0) = \pi(F, 0) = [0, 1] \qquad \pi(\mathsf{s}, 0) = 1$$

$$\pi(\mathsf{DP}(\mathcal{R})) \qquad \mathsf{f}\ (F\ x) \rightarrowtail \mathsf{f}\ (\mathrm{s}\ x)$$

$$\mathsf{f}\ (\mathsf{s}\ x) \not\rightarrowtail_{\pi(\mathsf{DP}(\mathcal{R}))} \mathsf{f}\ (\mathsf{s}\ x)$$

➜ stability of π is essential

Usable rules [Arts–Giesl, TCS '00] [Giesl et al., JAR '06]

Key idea . . . simplify ordering constraints

by extracting only rules relevant for $(D, \mathcal{R})$ chain

➜ modular termination proof

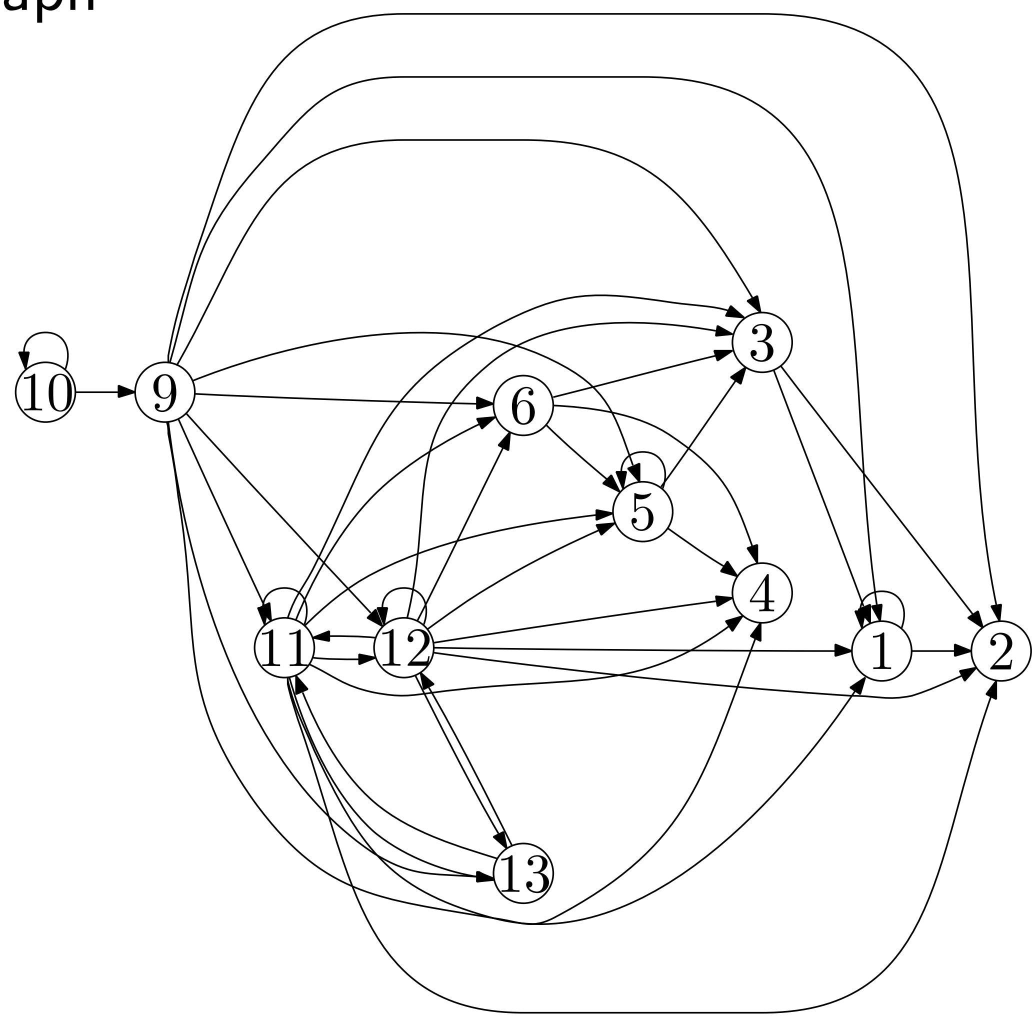
Termination using usable rules (example)

simply typed TRS $\mathcal{R}_1 \cup \mathcal{R}_2$

$$
\begin{array}{ll}
\mathcal{R}_1 \left\{ \begin{array}{lcl}
(+\ \mathsf{0})\ y & \to & y \\
(+\ (\mathsf{s}\ x))\ y & \to & \mathsf{s}\ ((+\ x)\ y) \\
(\mathsf{fold}\ F\ x)\ [\,] & \to & x \\
(\mathsf{fold}\ F\ x)\ (:\ y\ ys) & \to & (F\ y)\ ((\mathsf{fold}\ F\ x)\ ys) \\
\mathsf{sum} & \to & \mathsf{fold}\ +\ \mathsf{0}
\end{array} \right. \\
\mathcal{R}_2 \left\{ \begin{array}{lcl}
\mathsf{map}\ G\ [\,] & \to & [\,] \\
\mathsf{map}\ G\ (:\ x\ xs) & \to & :\ (G\ x)\ (\mathsf{map}\ G\ xs) \\
(\circ\ G\ H)\ x & \to & G\ (H\ x) \\
\mathsf{twice}\ G & \to & \circ\ G\ G
\end{array} \right.
\end{array}
$$

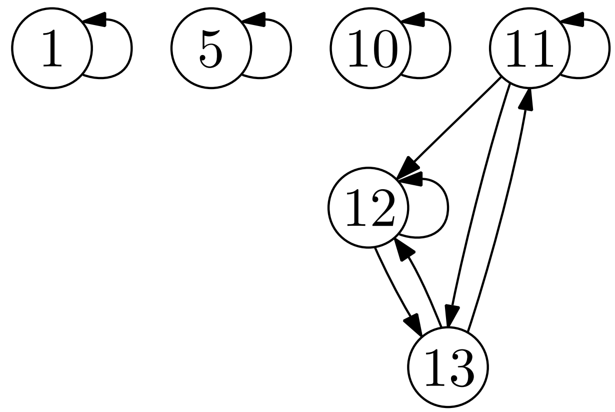
Termination using usable rules (example continued)

dependency graph



Termination using usable rules (example continued)

cycles in dependency graph



dependency pairs admitting cycles

$$
\begin{array}{ll}
\mathrm{DP}(\mathcal{R}_1) \left\{ \begin{array}{lllll}
(1) & (+\ (\mathsf{s}\ x))\ y & \rightarrowtail & (+\ x)\ y \\
(5) & (\mathsf{fold}\ F\ x)\ (:\ y\ ys) & \rightarrowtail & (\mathsf{fold}\ F\ x)\ ys
\end{array} \right. \\
\mathrm{DP}(\mathcal{R}_2) \left\{ \begin{array}{lllll}
(10) & \mathsf{map}\ G\ (:\ x\ xs) & \rightarrowtail & \mathsf{map}\ G\ xs \\
(11) & (\circ\ G\ H)\ x & \rightarrowtail & G\ (H\ x) \\
(12) & (\circ\ G\ H)\ x & \rightarrowtail & H\ x \\
(13) & (\mathsf{twice}\ G)\ x & \rightarrowtail & (\circ\ G\ G)\ x
\end{array} \right.
\end{array}
$$

Termination using usable rules (example continued)

absence of $(\mathcal{R}_1 \cup \mathcal{R}_2, \{(1), (5), (10)\})$ chain by subterm criterion

$$\pi(+) := 1 \quad \pi(\text{fold}) := 3 \quad \pi(\text{map}) := 2$$

$$\begin{array}{lll} \pi(1) \subseteq \rhd & (+\ (\mathsf{s}\ x))\ y & \rhd\ (+\ x)\ y \\ \pi(5) \subseteq \rhd & (\text{fold}\ F\ x)\ (:\ y\ ys) & \rhd\ (\text{fold}\ F\ x)\ ys \\ \pi(10) \subseteq \rhd & \text{map}\ G\ (:\ x\ xs) & \rhd\ \text{map}\ G\ xs \end{array}$$

remaining dependency pairs (head-instantiated and head-marked)

$$D \left\{ \begin{array}{lll} (11a) & ((\circ^\sharp\ (\circ\ U\ V)\ H)\ x) & \rightarrowtail\ ((\circ^\sharp\ U\ V)\ (H\ x)) \\ (11b) & ((\circ^\sharp\ (\text{twice}\ U)\ H)\ x) & \rightarrowtail\ ((\text{twice}^\sharp\ U)\ (H\ x)) \\ (12a) & ((\circ^\sharp\ G\ (\circ\ U\ V))\ x) & \rightarrowtail\ ((\circ^\sharp\ U\ V)\ x) \\ (12b) & ((\circ^\sharp\ G\ (\text{twice}\ U))\ x) & \rightarrowtail\ ((\text{twice}^\sharp\ U)\ x) \\ (13) & ((\text{twice}^\sharp\ G)\ x) & \rightarrowtail\ ((\circ^\sharp\ G\ G)\ x) \end{array} \right.$$

➜ no appropriate filtering for $\mathcal{R}_1 \cup \mathcal{R}_2$ and D

Termination using usable rules (example continued)

argument filtering

$$\pi(\circ, 1) = [\,]$$
$$\pi(\circ^\sharp, 1) = \pi(\mathsf{twice}^\sharp, 1) = [0]$$
$$\pi(\mathsf{twice}^\sharp, 0) = [0, 1]$$
$$\pi(\circ^\sharp, 0) = \pi(\circ, 0) = [0, 1, 2]$$

ordering constraints (satisfied by path ordering)

$$\pi(D) \subseteq \succ \qquad \begin{array}{lcl} ((\circ^\sharp\ (\circ\ U\ V)\ H)\ x) & \succ & ((\circ^\sharp\ U\ V)\ (H\ x)) \\ ((\circ^\sharp\ (\mathsf{twice}\ U)\ H)\ x) & \succ & ((\mathsf{twice}^\sharp\ U)\ (H\ x)) \\ ((\circ^\sharp\ G\ (\circ\ U\ V))\ x) & \succ & ((\circ^\sharp\ U\ V)\ x) \\ ((\circ^\sharp\ G\ (\mathsf{twice}\ U))\ x) & \succ & ((\mathsf{twice}^\sharp\ U)\ x) \\ ((\mathsf{twice}^\sharp\ G)\ x) & \succ & ((\circ^\sharp\ G\ G)\ x) \end{array}$$

no usable rules for D $\quad$ ($\xrightarrow{\text{nh}}_{\mathcal{R}_1 \cup \mathcal{R}_2}$-step impossible in RHSs of $\pi(D)$)

➜ $\mathcal{R}_1 \cup \mathcal{R}_2$ imposes no further constraints, hence is terminating

Termination using usable rules

Theorem (new)

- $\mathcal{R}$: finite simply typed TRS
- $D \subseteq \mathrm{DP}(\mathcal{R})$ and D : head-instantiated
- $(\succsim, \succ)$: reduction pair
- π : argument filtering stable w.r.t. $(\mathrm{Usable}(\mathcal{R}, D, \pi), D)$
  s.t. $\pi(\mathrm{Usable}(\mathcal{R}, D, \pi)) \subseteq \succsim$ and $\pi(D^\sharp) \subseteq \succsim$
  and $(\mathsf{cons}\ x\ y) \succsim x,\ y$
- no $(\mathcal{R}, D \backslash \{l \rightarrowtail r \in D \mid \pi(l^\sharp) \succ \pi(r^\sharp)\})$ chain

$\Longrightarrow$ no $(\mathcal{R}, D)$ chain

Usable rules for simply typed DPs

rule usability determined by def-use relationship on symbols

in simply typed case

- depth of symbol
- instantiation of function variables
- argument expantion for rules of function type

need to be taken into account

Outline

1. introduction
2. simply typed term rewriting
3. dependency pairs
4. rgument filtering & usable rules
5. experiments
6. conclusiton

Experiments

automatic termination prover for simply typed TRSs

- basic dependency pair method [AotoYamada '05]
- reduction pairs [Toyama '08]
- argument filtering + usable rules
- 8000-line-code written in SML/NJ
- external SAT solver

collection of examples

- functional programs using typical higher-order functions
- 122 examples

Experiments

122 terminating examples

	SC	+ AF	+ UR	FO encoding + TTT2
success	98	115	121	94
success ratio	80%	94%	99%	77%
total time	3.8s	9.4s	12.2s	1246s

FO encoding . . . $\varphi(t_0\ t_1\ \cdots\ t_n) := \mathsf{a}_n(\varphi(t_0),\ \varphi(t_1),\ \ldots,\ \varphi(t_n))$

TTT2 . . . [Hirokawa–Middeldorp, I&C '07]

Outline

1. introduction
2. simply typed term rewriting
3. dependency pairs
4. rgument filtering & usable rules
5. experiments
6. conclusiton

Summary

extension of DP method to simply-typed case with

- argument filtering
- usable rules criterion

enables powerfull and efficient termination proof

Summary

extension of DP method to simply-typed case with

- argument filtering
- usable rules criterion

enables powerfull and efficient termination proof

Further work

- higher-order rewriting with bound variables
- solving ordering constrains by interpretation
- comparison with other works (e.g. labelling transformation)