Constraints for Argument Filterings

Harald Zankl

Nao Hirokawa

Aart Middeldorp

Institute of Computer Science
University of Innsbruck
Austria

January 24, 2007

- Motivation
- Term Rewriting
- SAT Encoding
- Implementation Issues
- Experimental Results
- Remarks

Why Encode Termination Problems as Satisfiability Problems?

Why Encode Termination Problems as Satisfiability Problems?

- execution speed

Why Encode Termination Problems as Satisfiability Problems?

- execution speed
- ease of implementation

Why Encode Termination Problems as Satisfiability Problems?

- execution speed
- ease of implementation
- developments in SAT community are directly available

- Motivation
- **Term Rewriting**
- SAT Encoding
- Implementation Issues
- Experimental Results
- Remarks

signature

0 constant $\mathsf{s}, \mathsf{p}, \mathsf{fac}$ unary $+, \times$ binary

rewrite rules

$$\begin{aligned}
\mathsf{fac}(0) &\to \mathsf{s}(0) \\
\mathsf{fac}(\mathsf{s}(x)) &\to \mathsf{s}(x) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(x))) \\
\mathsf{p}(\mathsf{s}(x)) &\to x
\end{aligned}
\qquad
\begin{aligned}
0 + y &\to y \\
\mathsf{s}(x) + y &\to \mathsf{s}(x + y) \\
0 \times y &\to 0 \\
\mathsf{s}(x) \times y &\to x \times y + y
\end{aligned}$$

TRS

signature

0 constant s, p, fac unary $+$, $\times$ binary

rewrite rules

$$\begin{aligned}
\mathsf{fac}(0) &\to \mathsf{s}(0) & 0 + y &\to y \\
\mathsf{fac}(\mathsf{s}(x)) &\to \mathsf{s}(x) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(x))) & \mathsf{s}(x) + y &\to \mathsf{s}(x + y) \\
\mathsf{p}(\mathsf{s}(x)) &\to x & 0 \times y &\to 0 \\
& & \mathsf{s}(x) \times y &\to x \times y + y
\end{aligned}$$

TRS

rewriting

$\mathsf{fac}(\mathsf{s}(0))$

signature

0 constant $\quad$ $\mathsf{s}, \mathsf{p}, \mathsf{fac}$ unary $\quad$ $+, \times$ binary

rewrite rules

$$\begin{aligned} \mathsf{fac}(0) &\to \mathsf{s}(0) & 0 + y &\to y \\ \mathsf{fac}(\mathsf{s}(x)) &\to \mathsf{s}(x) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(x))) & \mathsf{s}(x) + y &\to \mathsf{s}(x + y) \\ \mathsf{p}(\mathsf{s}(x)) &\to x & 0 \times y &\to 0 \\ & \qquad\qquad \text{TRS} & \mathsf{s}(x) \times y &\to x \times y + y \end{aligned}$$

rewriting

$$\mathsf{fac}(\mathsf{s}(0)) \to \mathsf{s}(0) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(0)))$$

signature

0 constant $\quad$ s, p, fac unary $\quad$ $+$, $\times$ binary

rewrite rules

$$\begin{aligned} \mathsf{fac}(0) &\to \mathsf{s}(0) \\ \mathsf{fac}(\mathsf{s}(x)) &\to \mathsf{s}(x) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(x))) \\ \mathsf{p}(\mathsf{s}(x)) &\to x \end{aligned}$$

$$\begin{aligned} 0 + y &\to y \\ \mathsf{s}(x) + y &\to \mathsf{s}(x + y) \\ 0 \times y &\to 0 \\ \mathsf{s}(x) \times y &\to x \times y + y \end{aligned}$$

TRS

rewriting

$$\mathsf{fac}(\mathsf{s}(0)) \to \mathsf{s}(0) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(0)))$$

signature

0 constant $\quad$ $\mathsf{s}, \mathsf{p}, \mathsf{fac}$ unary $\quad$ $+, \times$ binary

rewrite rules

$$\begin{aligned} \mathsf{fac}(0) &\to \mathsf{s}(0) \\ \mathsf{fac}(\mathsf{s}(x)) &\to \mathsf{s}(x) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(x))) \\ \mathsf{p}(\mathsf{s}(x)) &\to x \end{aligned}$$

$$\begin{aligned} 0 + y &\to y \\ \mathsf{s}(x) + y &\to \mathsf{s}(x + y) \\ 0 \times y &\to 0 \\ \mathsf{s}(x) \times y &\to x \times y + y \end{aligned}$$

TRS

rewriting

$$\mathsf{fac}(\mathsf{s}(0)) \to \mathsf{s}(0) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(0))) \to \mathsf{s}(0) \times \mathsf{fac}(0)$$

signature

0 constant $\mathsf{s}, \mathsf{p}, \mathsf{fac}$ unary $+, \times$ binary

rewrite rules

$$\begin{aligned} \mathsf{fac}(0) &\to \mathsf{s}(0) & 0 + y &\to y \\ \mathsf{fac}(\mathsf{s}(x)) &\to \mathsf{s}(x) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(x))) & \mathsf{s}(x) + y &\to \mathsf{s}(x + y) \\ \mathsf{p}(\mathsf{s}(x)) &\to x & 0 \times y &\to 0 \\ & \quad \text{TRS} & \mathsf{s}(x) \times y &\to x \times y + y \end{aligned}$$

rewriting

$$\mathsf{fac}(\mathsf{s}(0)) \to \mathsf{s}(0) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(0))) \to \mathsf{s}(0) \times \mathsf{fac}(0)$$

signature

0 constant $\mathsf{s}, \mathsf{p}, \mathsf{fac}$ unary $+, \times$ binary

rewrite rules

$$\begin{aligned}
\mathsf{fac}(0) &\to \mathsf{s}(0) \\
\mathsf{fac}(\mathsf{s}(x)) &\to \mathsf{s}(x) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(x))) \\
\mathsf{p}(\mathsf{s}(x)) &\to x
\end{aligned}
\qquad
\begin{aligned}
0 + y &\to y \\
\mathsf{s}(x) + y &\to \mathsf{s}(x + y) \\
0 \times y &\to 0 \\
\mathsf{s}(x) \times y &\to x \times y + y
\end{aligned}$$

TRS

rewriting

$$\mathsf{fac}(\mathsf{s}(0)) \to \mathsf{s}(0) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(0))) \to \mathsf{s}(0) \times \mathsf{fac}(0) \to \mathsf{s}(0) \times \mathsf{s}(0)$$

signature

0 constant s, p, fac unary $+$, $\times$ binary

rewrite rules

$$\begin{aligned} \mathsf{fac}(0) &\to \mathsf{s}(0) & 0 + y &\to y \\ \mathsf{fac}(\mathsf{s}(x)) &\to \mathsf{s}(x) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(x))) & \mathsf{s}(x) + y &\to \mathsf{s}(x + y) \\ \mathsf{p}(\mathsf{s}(x)) &\to x & 0 \times y &\to 0 \\ & \qquad\qquad \text{TRS} & \mathsf{s}(x) \times y &\to x \times y + y \end{aligned}$$

rewriting

$$\mathsf{fac}(\mathsf{s}(0)) \to \mathsf{s}(0) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(0))) \to \mathsf{s}(0) \times \mathsf{fac}(0) \to \mathsf{s}(0) \times \mathsf{s}(0)$$

signature

0 constant s, p, fac unary $+$, $\times$ binary

rewrite rules

$$\begin{aligned} \mathsf{fac}(0) &\to \mathsf{s}(0) \\ \mathsf{fac}(\mathsf{s}(x)) &\to \mathsf{s}(x) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(x))) \\ \mathsf{p}(\mathsf{s}(x)) &\to x \end{aligned}$$

$$\begin{aligned} 0 + y &\to y \\ \mathsf{s}(x) + y &\to \mathsf{s}(x + y) \\ 0 \times y &\to 0 \\ \mathsf{s}(x) \times y &\to x \times y + y \end{aligned}$$

TRS

rewriting

$$\begin{aligned} \mathsf{fac}(\mathsf{s}(0)) &\to \mathsf{s}(0) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(0))) \to \mathsf{s}(0) \times \mathsf{fac}(0) \to \mathsf{s}(0) \times \mathsf{s}(0) \\ &\to 0 \times \mathsf{s}(0) + \mathsf{s}(0) \end{aligned}$$

signature

0 constant   s, p, fac unary $+, \times$ binary

rewrite rules

$$\mathsf{fac}(0) \to \mathsf{s}(0)$$
$$\mathsf{fac}(\mathsf{s}(x)) \to \mathsf{s}(x) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(x)))$$
$$\mathsf{p}(\mathsf{s}(x)) \to x$$

$$0 + y \to y$$
$$\mathsf{s}(x) + y \to \mathsf{s}(x + y)$$
$$0 \times y \to 0$$
$$\mathsf{s}(x) \times y \to x \times y + y$$

TRS

rewriting

$$\mathsf{fac}(\mathsf{s}(0)) \to \mathsf{s}(0) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(0))) \to \mathsf{s}(0) \times \mathsf{fac}(0) \to \mathsf{s}(0) \times \mathsf{s}(0)$$
$$\to 0 \times \mathsf{s}(0) + \mathsf{s}(0)$$

signature

0 constant $\mathsf{s}, \mathsf{p}, \mathsf{fac}$ unary $+, \times$ binary

rewrite rules

$$\begin{aligned} \mathsf{fac}(0) &\to \mathsf{s}(0) \\ \mathsf{fac}(\mathsf{s}(x)) &\to \mathsf{s}(x) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(x))) \\ \mathsf{p}(\mathsf{s}(x)) &\to x \end{aligned}$$

$$\begin{aligned} 0 + y &\to y \\ \mathsf{s}(x) + y &\to \mathsf{s}(x + y) \\ 0 \times y &\to 0 \\ \mathsf{s}(x) \times y &\to x \times y + y \end{aligned}$$

TRS

rewriting

$$\begin{aligned} \mathsf{fac}(\mathsf{s}(0)) &\to \mathsf{s}(0) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(0))) \to \mathsf{s}(0) \times \mathsf{fac}(0) \to \mathsf{s}(0) \times \mathsf{s}(0) \\ &\to 0 \times \mathsf{s}(0) + \mathsf{s}(0) \to 0 + \mathsf{s}(0) \end{aligned}$$

signature

0 constant $\mathsf{s}, \mathsf{p}, \mathsf{fac}$ unary $+, \times$ binary

rewrite rules

$$\begin{aligned} \mathsf{fac}(0) &\to \mathsf{s}(0) \\ \mathsf{fac}(\mathsf{s}(x)) &\to \mathsf{s}(x) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(x))) \\ \mathsf{p}(\mathsf{s}(x)) &\to x \end{aligned}$$

$$\begin{aligned} 0 + y &\to y \\ \mathsf{s}(x) + y &\to \mathsf{s}(x + y) \\ 0 \times y &\to 0 \\ \mathsf{s}(x) \times y &\to x \times y + y \end{aligned}$$

TRS

rewriting

$$\begin{aligned} \mathsf{fac}(\mathsf{s}(0)) &\to \mathsf{s}(0) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(0))) \to \mathsf{s}(0) \times \mathsf{fac}(0) \to \mathsf{s}(0) \times \mathsf{s}(0) \\ &\to 0 \times \mathsf{s}(0) + \mathsf{s}(0) \to 0 + \mathsf{s}(0) \end{aligned}$$

signature

0 constant s, p, fac unary $+$, $\times$ binary

rewrite rules

$$\begin{aligned} \mathsf{fac}(0) &\to \mathsf{s}(0) \\ \mathsf{fac}(\mathsf{s}(x)) &\to \mathsf{s}(x) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(x))) \\ \mathsf{p}(\mathsf{s}(x)) &\to x \end{aligned}$$

$$\begin{aligned} 0 + y &\to y \\ \mathsf{s}(x) + y &\to \mathsf{s}(x + y) \\ 0 \times y &\to 0 \\ \mathsf{s}(x) \times y &\to x \times y + y \end{aligned}$$

TRS

rewriting

$$\begin{aligned} \mathsf{fac}(\mathsf{s}(0)) &\to \mathsf{s}(0) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(0))) \to \mathsf{s}(0) \times \mathsf{fac}(0) \to \mathsf{s}(0) \times \mathsf{s}(0) \\ &\to 0 \times \mathsf{s}(0) + \mathsf{s}(0) \to 0 + \mathsf{s}(0) \to \mathsf{s}(0) \end{aligned}$$

signature

0 constant $\mathsf{s}, \mathsf{p}, \mathsf{fac}$ unary $+, \times$ binary

rewrite rules

$$\begin{aligned} \mathsf{fac}(0) &\to \mathsf{s}(0) \\ \mathsf{fac}(\mathsf{s}(x)) &\to \mathsf{s}(x) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(x))) \\ \mathsf{p}(\mathsf{s}(x)) &\to x \end{aligned} \qquad \begin{aligned} 0 + y &\to y \\ \mathsf{s}(x) + y &\to \mathsf{s}(x + y) \\ 0 \times y &\to 0 \\ \mathsf{s}(x) \times y &\to x \times y + y \end{aligned}$$

TRS

rewriting

$$\begin{aligned} \mathsf{fac}(\mathsf{s}(0)) &\to \mathsf{s}(0) \times \mathsf{fac}(\mathsf{p}(\mathsf{s}(0))) \to \mathsf{s}(0) \times \mathsf{fac}(0) \to \mathsf{s}(0) \times \mathsf{s}(0) \\ &\to 0 \times \mathsf{s}(0) + \mathsf{s}(0) \to 0 + \mathsf{s}(0) \to \mathsf{s}(0) \end{aligned}$$

normal form

Definition

TRS is terminating if there are no infinite rewrite sequences

Definition

TRS is terminating if there are no infinite rewrite sequences

Theorem

TRS $\mathcal{R}$ is terminating if there is a reduction order $>$ with $\mathcal{R} \subseteq >$.

- $\rhd_{\mathsf{emb}}$, $>_{\mathsf{lpo}}$, $>_{\mathsf{kbo}}$, ... *are reduction orders*

Definition

TRS is terminating if there are no infinite rewrite sequences

Theorem

TRS $\mathcal{R}$ is terminating if there is a reduction order $>$ with $\mathcal{R} \subseteq >$.

- $\rhd_{\mathsf{emb}}, >_{\mathsf{lpo}}, >_{\mathsf{kbo}}, \ldots$ *are reduction orders*

Example (emb)

signature

0	constant	s, p, fac	unary	$+, \times$	binary

Definition

TRS is terminating if there are no infinite rewrite sequences

Theorem

TRS $\mathcal{R}$ is terminating if there is a reduction order $>$ with $\mathcal{R} \subseteq >$.

- $\rhd_{\mathsf{emb}}, >_{\mathsf{lpo}}, >_{\mathsf{kbo}}, \ldots$ *are reduction orders*

Example (emb)

signature

0 constant $\qquad$ $\mathsf{s}, \mathsf{p}, \mathsf{fac}$ unary $\qquad$ $+, \times$ binary

rules

$$\mathsf{s}(x) \to x \qquad \mathsf{fac}(x) \to x \qquad \mathsf{p}(x) \to x$$

$$x + y \to x \qquad x + y \to y \qquad x \times y \to x \qquad x \times y \to y$$

Definition

TRS is terminating if there are no infinite rewrite sequences

Theorem

TRS $\mathcal{R}$ is terminating if there is a reduction order $>$ with $\mathcal{R} \subseteq >$.

- $\rhd_{\mathsf{emb}}$, $>_{\mathsf{lpo}}$, $>_{\mathsf{kbo}}$, $\ldots$ *are reduction orders*

Example (emb)

signature

0 constant s, p, fac unary $+$, $\times$ binary

rules

$\mathsf{s}(x) \to x \qquad \mathsf{fac}(x) \to x \qquad \mathsf{p}(x) \to x$

$x + y \to x \qquad x + y \to y \qquad x \times y \to x \qquad x \times y \to y$

relation

$\mathsf{s}(\mathsf{p}(x)) \rhd_{\mathsf{emb}} x \qquad x + 0 \rhd_{\mathsf{emb}} x$

$\mathsf{s}(x) \times y \not\rhd_{\mathsf{emb}} (x \times y) + y \qquad \mathsf{s}(x) \times y \rhd_{\mathsf{emb}} x \times y$

Theorem

TRS $\mathcal{R}$ is terminating if $\forall$ cycle $\mathcal{C}$ in dependency graph of $\mathcal{R}$
$\exists$ reduction pair $(\geqslant, >)$ such that

1. $\mathcal{R} \cup \mathcal{C} \subseteq \geqslant$
2. $\mathcal{C} \cap > \neq \varnothing$

Theorem

TRS $\mathcal{R}$ is terminating if $\forall$ cycle $\mathcal{C}$ in dependency graph of $\mathcal{R}$ $\exists$ argument filtering π $\exists$ reduction pair $(\geqslant, >)$ such that

1. $\pi(\mathcal{R} \cup \mathcal{C}) \subseteq \geqslant$
2. $\pi(\mathcal{C}) \cap > \neq \varnothing$

Theorem

TRS $\mathcal{R}$ is terminating if $\forall$ cycle $\mathcal{C}$ in dependency graph of $\mathcal{R}$
$\exists$ argument filtering π $\exists$ reduction pair $(\geqslant, >)$ such that

1. $\pi(\mathcal{R} \cup \mathcal{C}) \subseteq \geqslant$
2. $\pi(\mathcal{C}) \cap > \neq \varnothing$

- $(\trianglerighteq_{\mathsf{emb}}, \triangleright_{\mathsf{emb}})$, $(\geqslant_{\mathsf{lpo}}, >_{\mathsf{lpo}})$, $(\geqslant_{\mathsf{kbo}}, >_{\mathsf{kbo}})$, ... *are reduction pairs*

Theorem

TRS $\mathcal{R}$ is terminating if $\forall$ cycle $\mathcal{C}$ in dependency graph of $\mathcal{R}$
$\exists$ argument filtering π $\exists$ reduction pair $(\geqslant, >)$ such that

1. $\pi(\mathcal{R} \cup \mathcal{C}) \subseteq \geqslant$
2. $\pi(\mathcal{C}) \cap > \neq \varnothing$

- $(\trianglerighteq_{\mathsf{emb}}, \triangleright_{\mathsf{emb}})$, $(\geqslant_{\mathsf{lpo}}, >_{\mathsf{lpo}})$, $(\geqslant_{\mathsf{kbo}}, >_{\mathsf{kbo}})$, $\ldots$ *are reduction pairs*

Definition

argument filtering is mapping π such that for every *n*-ary function symbol *f* one of following alternatives holds:

Theorem

TRS $\mathcal{R}$ is terminating if $\forall$ cycle $\mathcal{C}$ in dependency graph of $\mathcal{R}$ $\exists$ argument filtering π $\exists$ reduction pair $(\geqslant, >)$ such that

1. $\pi(\mathcal{R} \cup \mathcal{C}) \subseteq \geqslant$
2. $\pi(\mathcal{C}) \cap > \neq \varnothing$

- $(\trianglerighteq_{\mathsf{emb}}, \triangleright_{\mathsf{emb}})$, $(\geqslant_{\mathsf{lpo}}, >_{\mathsf{lpo}})$, $(\geqslant_{\mathsf{kbo}}, >_{\mathsf{kbo}})$, ... *are reduction pairs*

Definition

argument filtering is mapping π such that for every n-ary function symbol f one of following alternatives holds:

① $\pi(f) = i$ with $i \in \{1, \ldots, n\}$

Theorem

TRS $\mathcal{R}$ is terminating if $\forall$ cycle $\mathcal{C}$ in dependency graph of $\mathcal{R}$ $\exists$ argument filtering π $\exists$ reduction pair $(\geqslant, >)$ such that

1. $\pi(\mathcal{R} \cup \mathcal{C}) \subseteq \geqslant$
2. $\pi(\mathcal{C}) \cap > \neq \varnothing$

- $(\trianglerighteq_{\mathsf{emb}}, \triangleright_{\mathsf{emb}})$, $(\geqslant_{\mathsf{lpo}}, >_{\mathsf{lpo}})$, $(\geqslant_{\mathsf{kbo}}, >_{\mathsf{kbo}})$, $\ldots$ *are reduction pairs*

Definition

argument filtering is mapping π such that for every n-ary function symbol f one of following alternatives holds:

① $\pi(f) = i$ with $i \in \{1, \ldots, n\}$

② $\pi(f) = [i_1, \ldots, i_m]$ with $1 \leqslant i_1 \leqslant \cdots \leqslant i_m \leqslant n$

Theorem

TRS $\mathcal{R}$ is terminating if $\forall$ cycle $\mathcal{C}$ in dependency graph of $\mathcal{R}$ $\exists$ argument filtering π $\exists$ reduction pair $(\geqslant, >)$ such that

1. $\pi(\mathcal{R} \cup \mathcal{C}) \subseteq \geqslant$
2. $\pi(\mathcal{C}) \cap > \neq \varnothing$

- $(\trianglerighteq_{\mathsf{emb}}, \triangleright_{\mathsf{emb}})$, $(\geqslant_{\mathsf{lpo}}, >_{\mathsf{lpo}})$, $(\geqslant_{\mathsf{kbo}}, >_{\mathsf{kbo}})$, $\ldots$ *are reduction pairs*

Definition

argument filtering is mapping π such that for every n-ary function symbol f one of following alternatives holds:

① $\pi(f) = i$ with $i \in \{1, \ldots, n\}$

② $\pi(f) = [i_1, \ldots, i_m]$ with $1 \leqslant i_1 \leqslant \cdots \leqslant i_m \leqslant n$

- $$\pi(t) = \begin{cases} t & \text{if } t \text{ is variable} \\ \pi(t_i) & \text{if } t = f(t_1, \ldots, t_n) \text{ and ①} \\ f(\pi(t_{i_1}), \ldots, \pi(t_{i_m})) & \text{if } t = f(t_1, \ldots, t_n) \text{ and ②} \end{cases}$$

Example (argument filtering)

signature

0 constant $\quad$ s, p, fac unary $\quad$ $+, \times$ binary

Example (argument filtering)

signature

0	constant	s, p, fac	unary	$+$, $\times$	binary

filtering

$$\begin{aligned} \pi(0) &= [] \\ \pi(\mathsf{s}) &= [] \\ \pi(\mathsf{p}) &= 1 \\ \pi(\mathsf{fac}) &= [1] \\ \pi(+) &= [2] \\ \pi(\times) &= [1,2] \end{aligned}$$

Example (argument filtering)

signature

0 constant s, p, fac unary $+$, $\times$ binary

filtering

$$\begin{aligned}\pi(0) &= [] \\ \pi(s) &= [] \\ \pi(p) &= 1 \\ \pi(fac) &= [1] \\ \pi(+) &= [2] \\ \pi(\times) &= [1,2]\end{aligned}$$

$$\pi(0) = 0$$

Example (argument filtering)

signature

0 constant   s, p, fac unary $+$, $\times$ binary

filtering

$$\begin{aligned} \pi(0) &= [] \\ \pi(\mathsf{s}) &= [] \\ \pi(\mathsf{p}) &= 1 \\ \pi(\mathsf{fac}) &= [1] \\ \pi(+) &= [2] \\ \pi(\times) &= [1,2] \end{aligned}$$

$$\begin{aligned} \pi(0) &= 0 \\ \pi(\mathsf{s}(x)) &= \mathsf{s} \end{aligned}$$

Example (argument filtering)

signature

0 constant $\mathsf{s}, \mathsf{p}, \mathsf{fac}$ unary $+, \times$ binary

filtering

$$\pi(0) = [] \qquad \pi(0) = 0$$
$$\pi(\mathsf{s}) = [] \qquad \pi(\mathsf{s}(x)) = \mathsf{s}$$
$$\pi(\mathsf{p}) = 1 \qquad \pi(\mathsf{p}(x)) = x$$
$$\pi(\mathsf{fac}) = [1]$$
$$\pi(+) = [2]$$
$$\pi(\times) = [1, 2]$$

Example (argument filtering)

signature

0 constant $\mathsf{s}, \mathsf{p}, \mathsf{fac}$ unary $+, \times$ binary

filtering

$$\begin{aligned} \pi(0) &= [] \\ \pi(\mathsf{s}) &= [] \\ \pi(\mathsf{p}) &= 1 \\ \pi(\mathsf{fac}) &= [1] \\ \pi(+) &= [2] \\ \pi(\times) &= [1,2] \end{aligned}$$

$$\begin{aligned} \pi(0) &= 0 \\ \pi(\mathsf{s}(x)) &= \mathsf{s} \\ \pi(\mathsf{p}(x)) &= x \\ \pi(\mathsf{fac}(x)) &= \mathsf{fac}(x) \end{aligned}$$

Example (argument filtering)

signature

0 constant   s, p, fac unary $+$, $\times$ binary

filtering

$$\begin{aligned} \pi(0) &= [] \\ \pi(s) &= [] \\ \pi(p) &= 1 \\ \pi(fac) &= [1] \\ \pi(+) &= [2] \\ \pi(\times) &= [1,2] \end{aligned} \qquad \begin{aligned} \pi(0) &= 0 \\ \pi(s(x)) &= s \\ \pi(p(x)) &= x \\ \pi(fac(x)) &= fac(x) \\ \pi(+(x,y)) &= +(y) \end{aligned}$$

Example (argument filtering)

signature

0 constant   s, p, fac unary $+$, $\times$ binary

filtering

$$\begin{aligned} \pi(0) &= [] & \pi(0) &= 0 \\ \pi(s) &= [] & \pi(s(x)) &= s \\ \pi(p) &= 1 & \pi(p(x)) &= x \\ \pi(fac) &= [1] & \pi(fac(x)) &= fac(x) \\ \pi(+) &= [2] & \pi(+(x,y)) &= +(y) \\ \pi(\times) &= [1,2] & \pi(\times(x,y)) &= \times(x,y) \end{aligned}$$

- Motivation
- Term Rewriting
- **SAT Encoding**
- Implementation Issues
- Experimental Results
- Remarks

Definition

- $n + 1$ propositional variables for n-ary function symbol f:
  - X_f select between ① and ②
  - $X_f^1, \ldots, X_f^n$ select arguments

Definition

- $n + 1$ propositional variables for n-ary function symbol f:
  - X_f select between ① and ②
  - $X_f^1, \ldots, X_f^n$ select arguments
- $\mathcal{X}_\mathcal{F} = \{\, X_f, X_f^1, \ldots, X_f^n \mid f \in \mathcal{F} \text{ has arity } n \,\}$

Definition

- $n+1$ propositional variables for n-ary function symbol f:
  - X_f select between ① and ②
  - $X_f^1, \ldots, X_f^n$ select arguments
- $\mathcal{X}_\mathcal{F} = \{ X_f, X_f^1, \ldots, X_f^n \mid f \in \mathcal{F}$ has arity $n \}$

Definition

induced assignment α_π for argument filtering π:

$$\alpha_\pi(X_f) = \begin{cases} \text{true} & \text{if } \pi(f) = [i_1, \ldots, i_m] \\ \text{false} & \text{if } \pi(f) = i \end{cases}$$

$$\alpha_\pi(X_f^i) = \begin{cases} \text{true} & \text{if } i \in \pi(f) \\ \text{false} & \text{if } i \notin \pi(f) \end{cases}$$

Definition

- assignment α is argument filtering consistent if for all f

$$\text{if } \alpha \nvDash X_f \text{ then } \exists! \; i \text{ such that } \alpha \vDash X_f^i$$

Definition

- assignment α is argument filtering consistent if for all f

$$\text{if } \alpha \not\models X_f \text{ then } \exists!\ i \text{ such that } \alpha \models X_f^i$$

- propositional formula $$\mathrm{AF}(\mathcal{F}) = \bigwedge_{f \in \mathcal{F}} \left(X_f \vee \bigvee_{i=1}^{\mathrm{arity}(f)} \left(X_f^i \wedge \bigwedge_{j \neq i} \neg X_f^j \right) \right)$$

Definition

- assignment α is argument filtering consistent if for all f
$$\text{if } \alpha \nvDash X_f \text{ then } \exists! \; i \text{ such that } \alpha \vDash X_f^i$$
- propositional formula $\quad \mathsf{AF}(\mathcal{F}) = \bigwedge_{f \in \mathcal{F}} \left(X_f \vee \bigvee_{i=1}^{\mathsf{arity}(f)} (X_f^i \wedge \bigwedge_{j \neq i} \neg X_f^j) \right)$

Lemma

assignment α for $\mathcal{X}_\mathcal{F}$ is argument filtering consistent iff $\alpha \vDash \mathsf{AF}(\mathcal{F})$

Definition

- assignment α is argument filtering consistent if for all f

$$\text{if } \alpha \not\models X_f \text{ then } \exists!\ i \text{ such that } \alpha \models X_f^i$$

- propositional formula $$\mathsf{AF}(\mathcal{F}) = \bigwedge_{f \in \mathcal{F}} \Bigl(X_f \vee \bigvee_{i=1}^{\mathsf{arity}(f)} (X_f^i \wedge \bigwedge_{j \neq i} \neg X_f^j) \Bigr)$$

Lemma

assignment α for $\mathcal{X}_{\mathcal{F}}$ is argument filtering consistent iff $\alpha \models \mathsf{AF}(\mathcal{F})$

Definition

induced argument filtering π_α for argument filtering consistent assignment α:

$$\pi_\alpha(f) = \begin{cases} [\, i \mid \alpha \models X_f^i \,] & \text{if } \alpha \models X_f \\ i & \text{if } \alpha \not\models X_f \text{ and } \alpha \models X_f^i \end{cases}$$

Aim

define propositional formulas $\ulcorner s \rhd^{\pi}_{\mathsf{emb}} t \urcorner$ and $\ulcorner s \unrhd^{\pi}_{\mathsf{emb}} t \urcorner$ such that

$$\pi_\alpha(s) \rhd_{\mathsf{emb}} \pi_\alpha(t) \quad \text{when} \quad \alpha \vDash \ulcorner s \rhd^{\pi}_{\mathsf{emb}} t \urcorner \wedge \mathsf{AF}(\mathcal{F})$$

and

$$\pi_\alpha(s) \unrhd_{\mathsf{emb}} \pi_\alpha(t) \quad \text{when} \quad \alpha \vDash \ulcorner s \unrhd^{\pi}_{\mathsf{emb}} t \urcorner \wedge \mathsf{AF}(\mathcal{F})$$

Definition ($\ulcorner s =^{\pi} t \urcorner$)

Definition ($\ulcorner s =^{\pi} t \urcorner$)

if $s \in \mathcal{V}$ then

$$\ulcorner s =^{\pi} t \urcorner = \begin{cases} \top & \text{if } s = t \\ \bot & \text{if } t \in \mathcal{V} \text{ and } s \neq t \\ \neg X_g \wedge \bigvee_{j=1}^{m} \left(X_g^j \wedge \ulcorner s =^{\pi} t_j \urcorner \right) & \text{if } t = g(t_1, \ldots, t_m) \end{cases}$$

Definition ($\ulcorner s =^\pi t \urcorner$)

if $s \in \mathcal{V}$ then

$$\ulcorner s =^\pi t \urcorner = \begin{cases} \top & \text{if } s = t \\ \bot & \text{if } t \in \mathcal{V} \text{ and } s \neq t \\ \neg X_g \wedge \bigvee_{j=1}^{m} \left(X_g^j \wedge \ulcorner s =^\pi t_j \urcorner \right) & \text{if } t = g(t_1, \ldots, t_m) \end{cases}$$

if $s = f(s_1, \ldots, s_n)$ then

Definition ($\ulcorner s =^\pi t \urcorner$)

if $s \in \mathcal{V}$ then

$$\ulcorner s =^\pi t \urcorner = \begin{cases} \top & \text{if } s = t \\ \bot & \text{if } t \in \mathcal{V} \text{ and } s \neq t \\ \neg X_g \wedge \bigvee_{j=1}^{m} \left(X_g^j \wedge \ulcorner s =^\pi t_j \urcorner \right) & \text{if } t = g(t_1, \ldots, t_m) \end{cases}$$

if $s = f(s_1, \ldots, s_n)$ then

- if $t \in \mathcal{V}$ then $$\ulcorner s =^\pi t \urcorner = \neg X_f \wedge \bigvee_{i=1}^{n} \left(X_f^i \wedge \ulcorner s_i =^\pi t \urcorner \right)$$

Definition ($\ulcorner s =^\pi t \urcorner$)

if $s \in \mathcal{V}$ then

$$\ulcorner s =^\pi t \urcorner = \begin{cases} \top & \text{if } s = t \\ \bot & \text{if } t \in \mathcal{V} \text{ and } s \neq t \\ \neg X_g \wedge \bigvee_{j=1}^{m} \left(X_g^j \wedge \ulcorner s =^\pi t_j \urcorner \right) & \text{if } t = g(t_1, \ldots, t_m) \end{cases}$$

if $s = f(s_1, \ldots, s_n)$ then

- if $t \in \mathcal{V}$ then $$\ulcorner s =^\pi t \urcorner = \neg X_f \wedge \bigvee_{i=1}^{n} \left(X_f^i \wedge \ulcorner s_i =^\pi t \urcorner \right)$$
- if $t = g(t_1, \ldots, t_m)$ with $f \neq g$ then

$$\ulcorner s =^\pi t \urcorner = \neg X_f \wedge \bigvee_{i=1}^{n} \left(X_f^i \wedge \ulcorner s_i =^\pi t \urcorner \right) \vee \neg X_g \wedge \bigvee_{j=1}^{m} \left(X_g^j \wedge \ulcorner s =^\pi t_j \urcorner \right)$$

Definition ($\ulcorner s =^{\pi} t \urcorner$)

if $s \in \mathcal{V}$ then

$$\ulcorner s =^{\pi} t \urcorner = \begin{cases} \top & \text{if } s = t \\ \bot & \text{if } t \in \mathcal{V} \text{ and } s \neq t \\ \neg X_g \wedge \bigvee_{j=1}^{m} \left(X_g^j \wedge \ulcorner s =^{\pi} t_j \urcorner \right) & \text{if } t = g(t_1, \ldots, t_m) \end{cases}$$

if $s = f(s_1, \ldots, s_n)$ then

- if $t \in \mathcal{V}$ then $$\ulcorner s =^{\pi} t \urcorner = \neg X_f \wedge \bigvee_{i=1}^{n} \left(X_f^i \wedge \ulcorner s_i =^{\pi} t \urcorner \right)$$
- if $t = g(t_1, \ldots, t_m)$ with $f \neq g$ then $$\ulcorner s =^{\pi} t \urcorner = \neg X_f \wedge \bigvee_{i=1}^{n} \left(X_f^i \wedge \ulcorner s_i =^{\pi} t \urcorner \right) \vee \neg X_g \wedge \bigvee_{j=1}^{m} \left(X_g^j \wedge \ulcorner s =^{\pi} t_j \urcorner \right)$$
- if $t = f(t_1, \ldots, t_n)$ then $$\ulcorner s =^{\pi} t \urcorner = \neg X_f \wedge \bigvee_{i=1}^{n} \left(X_f^i \wedge \ulcorner s_i =^{\pi} t_i \urcorner \right) \vee X_f \wedge \bigwedge_{i=1}^{n} \left(X_f^i \rightarrow \ulcorner s_i =^{\pi} t_i \urcorner \right)$$

Definition ($\ulcorner s \rhd^{\pi}_{\mathsf{emb}} t \urcorner$ $\quad \ulcorner s \unrhd^{\pi}_{\mathsf{emb}} t \urcorner = \ulcorner s \rhd^{\pi}_{\mathsf{emb}} t \urcorner \vee \ulcorner s =^{\pi} t \urcorner$)

Definition ($\ulcorner s \rhd^{\pi}_{\mathsf{emb}} t \urcorner$ $\quad \ulcorner s \trianglerighteq^{\pi}_{\mathsf{emb}} t \urcorner = \ulcorner s \rhd^{\pi}_{\mathsf{emb}} t \urcorner \vee \ulcorner s =^{\pi} t \urcorner$)

if $s \in \mathcal{V}$ then $\ulcorner s \rhd^{\pi}_{\mathsf{emb}} t \urcorner = \bot$

Definition ($\ulcorner s \rhd^{\pi}_{\text{emb}} t \urcorner$ $\quad \ulcorner s \unrhd^{\pi}_{\text{emb}} t \urcorner = \ulcorner s \rhd^{\pi}_{\text{emb}} t \urcorner \vee \ulcorner s =^{\pi} t \urcorner$)

if $s \in \mathcal{V}$ then $\ulcorner s \rhd^{\pi}_{\text{emb}} t \urcorner = \bot$ if $s = f(s_1, \ldots, s_n)$ then

- if $t \in \mathcal{V}$ then

$$\ulcorner s \rhd^{\pi}_{\text{emb}} t \urcorner = X_f \wedge \bigvee_{i=1}^{n} (X_f^i \wedge \ulcorner s_i \unrhd^{\pi}_{\text{emb}} t \urcorner) \vee \neg X_f \wedge \bigvee_{i=1}^{n} (X_f^i \wedge \ulcorner s_i \rhd^{\pi}_{\text{emb}} t \urcorner)$$

Definition ($\ulcorner s \rhd^{\pi}_{\text{emb}} t \urcorner$ $\quad \ulcorner s \unrhd^{\pi}_{\text{emb}} t \urcorner = \ulcorner s \rhd^{\pi}_{\text{emb}} t \urcorner \vee \ulcorner s =^{\pi} t \urcorner$)

if $s \in \mathcal{V}$ then $\ulcorner s \rhd^{\pi}_{\text{emb}} t \urcorner = \bot$ if $s = f(s_1, \ldots, s_n)$ then

- if $t \in \mathcal{V}$ then

$$\ulcorner s \rhd^{\pi}_{\text{emb}} t \urcorner = X_f \wedge \bigvee_{i=1}^{n} (X_f^i \wedge \ulcorner s_i \unrhd^{\pi}_{\text{emb}} t \urcorner) \vee \neg X_f \wedge \bigvee_{i=1}^{n} (X_f^i \wedge \ulcorner s_i \rhd^{\pi}_{\text{emb}} t \urcorner)$$

- if $t = g(t_1, \ldots, t_m)$ with $f \neq g$ then

$$\ulcorner s \rhd^{\pi}_{\text{emb}} t \urcorner = X_f \wedge \Big(X_g \wedge \bigvee_{i=1}^{n} (X_f^i \wedge \ulcorner s_i \unrhd^{\pi}_{\text{emb}} t \urcorner) \vee \neg X_g \wedge \bigvee_{j=1}^{m} (X_g^j \wedge \ulcorner s \rhd^{\pi}_{\text{emb}} t_j \urcorner)\Big)$$
$$\vee \neg X_f \wedge \bigvee_{i=1}^{n} (X_f^i \wedge \ulcorner s_i \rhd^{\pi}_{\text{emb}} t \urcorner)$$

Definition ($\ulcorner s \rhd^{\pi}_{\mathsf{emb}} t \urcorner$ $\quad \ulcorner s \unrhd^{\pi}_{\mathsf{emb}} t \urcorner = \ulcorner s \rhd^{\pi}_{\mathsf{emb}} t \urcorner \vee \ulcorner s =^{\pi} t \urcorner$)

if $s \in \mathcal{V}$ then $\ulcorner s \rhd^{\pi}_{\mathsf{emb}} t \urcorner = \bot$ if $s = f(s_1, \ldots, s_n)$ then

- if $t \in \mathcal{V}$ then

$$\ulcorner s \rhd^{\pi}_{\mathsf{emb}} t \urcorner = X_f \wedge \bigvee_{i=1}^{n} (X_f^i \wedge \ulcorner s_i \unrhd^{\pi}_{\mathsf{emb}} t \urcorner) \vee \neg X_f \wedge \bigvee_{i=1}^{n} (X_f^i \wedge \ulcorner s_i \rhd^{\pi}_{\mathsf{emb}} t \urcorner)$$

- if $t = g(t_1, \ldots, t_m)$ with $f \neq g$ then

$$\ulcorner s \rhd^{\pi}_{\mathsf{emb}} t \urcorner = X_f \wedge \left(X_g \wedge \bigvee_{i=1}^{n} (X_f^i \wedge \ulcorner s_i \unrhd^{\pi}_{\mathsf{emb}} t \urcorner) \vee \neg X_g \wedge \bigvee_{j=1}^{m} (X_g^j \wedge \ulcorner s \rhd^{\pi}_{\mathsf{emb}} t_j \urcorner)\right) \\ \vee \neg X_f \wedge \bigvee_{i=1}^{n} (X_f^i \wedge \ulcorner s_i \rhd^{\pi}_{\mathsf{emb}} t \urcorner)$$

- if $t = f(t_1, \ldots, t_n)$ then

$$\ulcorner s \rhd^{\pi}_{\mathsf{emb}} t \urcorner = X_f \wedge \Bigg(\bigvee_{i=1}^{n} (X_f^i \wedge \ulcorner s_i \unrhd^{\pi}_{\mathsf{emb}} t \urcorner) \vee \bigwedge_{i=1}^{n} (X_f^i \rightarrow \ulcorner s_i \unrhd^{\pi}_{\mathsf{emb}} t_i \urcorner) \wedge \\ \bigvee_{i=1}^{n} (X_f^i \wedge \ulcorner s_i \rhd^{\pi}_{\mathsf{emb}} t_i \urcorner) \Bigg) \vee \neg X_f \wedge \bigvee_{i=1}^{n} (X_f^i \wedge \ulcorner s_i \rhd^{\pi}_{\mathsf{emb}} t_i \urcorner)$$

- BDDs vs SAT

- BDDs vs SAT MiniSat

- BDDs vs SAT MiniSat
- Tseitin's translation to CNF

- BDDs vs SAT MiniSat
- Tseitin's translation to CNF
- optimizations to reduce size of generated formulas

  e.g. $\ulcorner f(s_1,\ldots,s_n) =^{\pi} f(t_1,\ldots,t_n) \urcorner$

- BDDs vs SAT MiniSat
- Tseitin's translation to CNF
- optimizations to reduce size of generated formulas

  e.g. $\ulcorner f(s_1, \ldots, s_n) =^{\pi} f(t_1, \ldots, t_n) \urcorner$

  under AF($\mathcal{F}$) assumption

- BDDs vs SAT MiniSat
- Tseitin's translation to CNF
- optimizations to reduce size of generated formulas

  e.g. $\ulcorner f(s_1, \ldots, s_n) =^\pi f(t_1, \ldots, t_n) \urcorner$

  under AF($\mathcal{F}$) assumption

$$\neg X_f \wedge \bigvee_{i=1}^{n} \left(X_f^i \wedge \ulcorner s_i =^\pi t_i \urcorner \right) \vee X_f \wedge \bigwedge_{i=1}^{n} \left(X_f^i \rightarrow \ulcorner s_i =^\pi t_i \urcorner \right)$$

- BDDs vs SAT MiniSat
- Tseitin's translation to CNF
- optimizations to reduce size of generated formulas

  e.g. $\ulcorner f(s_1, \ldots, s_n) =^\pi f(t_1, \ldots, t_n) \urcorner$

  under AF($\mathcal{F}$) assumption

$$\neg X_f \wedge \bigvee_{i=1}^{n} \left(X_f^i \wedge \ulcorner s_i =^\pi t_i \urcorner \right) \vee X_f \wedge \bigwedge_{i=1}^{n} \left(X_f^i \rightarrow \ulcorner s_i =^\pi t_i \urcorner \right)$$

$$\Downarrow$$

$$\bigwedge_{i=1}^{n} \left(X_f^i \rightarrow \ulcorner s_i =^\pi t_i \urcorner \right)$$

- Motivation
- Term Rewriting
- SAT Encoding
- Implementation Issues
- **Experimental Results**
- Remarks

Embedding

865 TRSs in version 3.2 of TPDB

embedding	AProVE	$\mathsf{T_TT}$	sat
solved	194	194	194
timeout (60 seconds)	12	6	0
time (in seconds)	735	407	146

KBO and LPO

865 TRSs in 2006 edition of TPDB

KBO ∥ LPO	T$_T$T (L ; K)	T$_T$T (K ; L)	sat(2)	sat(3)	sat(4)
solved	310	295	305	338	343
timeout	121	136	6	9	14
time	7025	9025	1664	2076	2623

KBO and LPO

865 TRSs in 2006 edition of TPDB

KBO ‖ LPO	$\mathsf{T_TT}$ (L ; K)	$\mathsf{T_TT}$ (K ; L)	sat(2)		sat(3)		sat(4)	
solved	310	295	305	337	338	369	343	377
timeout	121	136	6	9	9	11	14	16
time	7025	9025	1664	1940	2076	2351	2623	2898

advanced usable rules

- Motivation
- Term Rewriting
- SAT Encoding
- Implementation Issues
- Experimental Results
- Remarks

- SAT encoding of other termination methods

- SAT encoding of other termination methods
  - MPO, RPO (with status)

- SAT encoding of other termination methods
  - MPO, RPO (with status)
  - polynomial interpretations

- SAT encoding of other termination methods
  - MPO, RPO (with status)
  - polynomial interpretations
  - matrix interpretations

- SAT encoding of other termination methods
  - MPO, RPO (with status)
  - polynomial interpretations
  - matrix interpretations
  - general version of KBO (weight function $\Rightarrow$ monotone algebra)

- SAT encoding of other termination methods
  - MPO, RPO (with status)
  - polynomial interpretations
  - matrix interpretations
  - general version of KBO (weight function $\Rightarrow$ monotone algebra)
  - ...

- SAT encoding of other termination methods
  - MPO, RPO (with status)
  - polynomial interpretations
  - matrix interpretations
  - general version of KBO (weight function $\Rightarrow$ monotone algebra)
  - $\cdots$
- Pseudo Boolean Constraints

- SAT encoding of other termination methods
  - MPO, RPO (with status)
  - polynomial interpretations
  - matrix interpretations
  - general version of KBO (weight function $\Rightarrow$ monotone algebra)
  - $\cdots$
- Pseudo Boolean Constraints
  - $(\sum_{1 \leqslant i \leqslant n} a_i * x_i) \circ m \qquad \circ \in \{\geqslant, =, \leqslant\}$

- SAT encoding of other termination methods
  - MPO, RPO (with status)
  - polynomial interpretations
  - matrix interpretations
  - general version of KBO (weight function $\Rightarrow$ monotone algebra)
  - $\cdots$
- Pseudo Boolean Constraints
  - $(\sum_{1 \leqslant i \leqslant n} a_i * x_i) \circ m \qquad \circ \in \{\geqslant, =, \leqslant\}$
  - get $+$ for free (KBO)

- SAT encoding of other termination methods
  - MPO, RPO (with status)
  - polynomial interpretations
  - matrix interpretations
  - general version of KBO (weight function $\Rightarrow$ monotone algebra)
  - $\cdots$
- Pseudo Boolean Constraints
  - $(\sum_{1 \leqslant i \leqslant n} a_i * x_i) \circ m \qquad \circ \in \{\geqslant, =, \leqslant\}$
  - get $+$ for free (KBO)
  - get $+$ and $\times$ for free (polynomial-, matrix interpretations, ...)