Automatic Certification of Termination Proofs

IsaFoR/CeTA

Christian Sternagel[1]

University of Innsbruck

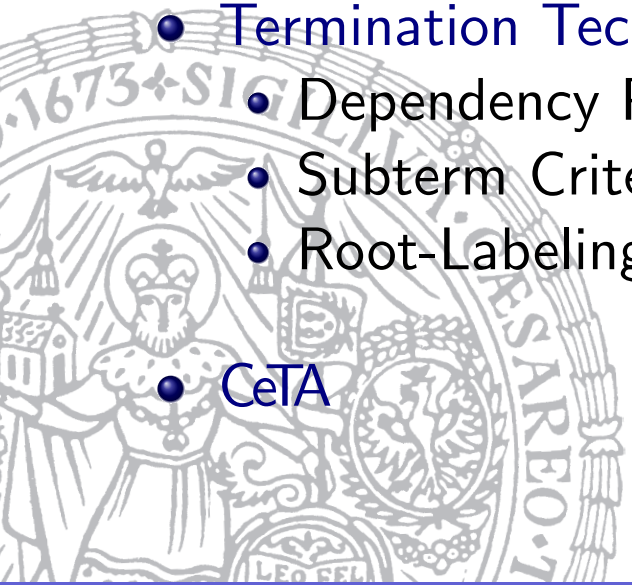
3rd Austria - Japan Summer Workshop on Rewriting

August 6, 2010

[1]FWF-Project: P18763

Overview

- Motivation
- Preliminaries
  - Isabelle/HOL
  - Code-Extraction
  - IsaFoR
- Termination Techniques
  - Dependency Pairs
  - Subterm Criterion
  - Root-Labeling
- CeTA



Correctness of Programs

Example ($\sum_{i=1}^{n} i$)

```
int sum(int n) {
  if (n == 0) return 0;
  else return (n - 1) + sum(n);
}
```

Correctness of Programs

Example ($\sum_{i=1}^{n} i$)

```
int sum(int n) {
  if (n == 0) return 0;
  else return (n - 1) + sum(n);
}
```

nonterminating

Correctness of Programs

Example ($\sum_{i=1}^{n} i$)

```
int sum(int n) {
  if (n == 0) return 0;
  else return (n - 1) + sum(n);
}
```

Problem

$$\texttt{sum}(2) = 1 + \texttt{sum}(2)$$

Correctness of Programs

Example ($\sum_{i=1}^{n} i$)

```
int sum(int n) {
  if (n == 0) return 0;
  else return (n - 1) + sum(n);
}
```

Problem

$$\cancel{\texttt{sum}(2)} = 1 + \cancel{\texttt{sum}(2)}$$

$$0 = 1$$

Correctness of Programs

Example ($\sum_{i=1}^{n} i$)

```
int sum(int n) {
  if (n == 0) return 0;
  else return (n - 1) + sum(n);
}
```

Problem

inconsistent

$$\cancel{\texttt{sum}}(2) = 1 + \cancel{\texttt{sum}}(2)$$

$$0 = 1$$

Correctness of Programs

Example ($\sum_{i=1}^{n} i$)

```
int sum(int n) {
  if (n == 0) return 0;
  else return (n - 1) + sum(n);
}
```

Problem

$$\cancel{\texttt{sum}(2)} = 1 + \cancel{\texttt{sum}(2)}$$

$$0 = 1$$

Need

terminating programs / total functions

Proving Termination

Example (TRS for $\sum_{i=1}^{n} i$)

$$\mathsf{sum}(0) \to 0$$
$$\mathsf{sum}(\mathsf{s}(n)) \to n + \mathsf{sum}(\mathsf{s}(n))$$

Proving Termination

Example (TRS for $\sum_{i=1}^{n} i$)

$$\mathsf{sum}(0) \to 0$$
$$\mathsf{sum}(\mathsf{s}(n)) \to \mathsf{s}(n) + \mathsf{sum}(n)$$

Proving Termination

Example (TRS for $\sum_{i=1}^{n} i$)

$$\mathsf{sum}(0) \to 0$$
$$\mathsf{sum}(\mathsf{s}(n)) \to \mathsf{s}(n) + \mathsf{sum}(n)$$

Informal Argument

Proving Termination

Example (TRS for $\sum_{i=1}^{n} i$)

$$\mathsf{sum}(0) \to 0$$
$$\mathsf{sum}(\mathsf{s}(n)) \to \mathsf{s}(n) + \mathsf{sum}(n)$$

Informal Argument

- analyze 'function calls': dependency pairs

$$\mathsf{sum}(\mathsf{s}(n)) \quad \mathsf{sum}(n) \tag{1}$$

Proving Termination

Example (TRS for $\sum_{i=1}^{n} i$)

$$\mathsf{sum}(0) \to 0$$
$$\mathsf{sum}(\mathsf{s}(n)) \to \mathsf{s}(n) + \mathsf{sum}(n)$$

Informal Argument

- analyze 'function calls': dependency pairs

$$\mathsf{sum}(\mathsf{s}(n)) \quad \mathsf{sum}(n) \tag{1}$$

- analyze the possible 'call order': dependency graph

$$(1) \circlearrowright$$

Proving Termination

Example (TRS for $\sum_{i=1}^{n} i$)

$$\mathsf{sum}(0) \to 0$$
$$\mathsf{sum}(\mathsf{s}(n)) \to \mathsf{s}(n) + \mathsf{sum}(n)$$

Informal Argument

- analyze 'function calls': dependency pairs

$$\mathsf{sum}(\mathsf{s}(n)) \quad \mathsf{sum}(n) \tag{1}$$

- analyze the possible 'call order': dependency graph

$$(1) \circlearrowleft$$

- number of s decreases in only cycle

State-Of-The-Art Termination Techniques

Overview

reduction orders (polynomial orders, KBO, RPO, matrix-interpretations, arctic matrices), dependency pairs, matchbounds, dependency graph decomposition, semantic labeling (predictive labeling, root-labeling), argument filters, subterm criterion, usable rules, size-change termination, bounded increase, string reversal, increasing interpretations, loops, ...

State-Of-The-Art Termination Techniques

Overview

reduction orders (polynomial orders, KBO, RPO, matrix-interpretations, arctic matrices), dependency pairs, matchbounds, dependency graph decomposition, semantic labeling (predictive labeling, root-labeling), argument filters, subterm criterion, usable rules, size-change termination, bounded increase, string reversal, increasing interpretations, loops, ...

Remark

State-Of-The-Art Termination Techniques

Overview

reduction orders (polynomial orders, KBO, RPO, matrix-interpretations, arctic matrices), dependency pairs, matchbounds, dependency graph decomposition, semantic labeling (predictive labeling, root-labeling), argument filters, subterm criterion, usable rules, size-change termination, bounded increase, string reversal, increasing interpretations, loops, ...

Remark

- proofs are generated by termination tools

State-Of-The-Art Termination Techniques

Overview

reduction orders (polynomial orders, KBO, RPO, matrix-interpretations, arctic matrices), dependency pairs, matchbounds, dependency graph decomposition, semantic labeling (predictive labeling, root-labeling), argument filters, subterm criterion, usable rules, size-change termination, bounded increase, string reversal, increasing interpretations, loops, ...

Remark

- proofs are generated by termination tools
- proofs tend to get huge (several MB)

State-Of-The-Art Termination Techniques

Overview

reduction orders (polynomial orders, KBO, RPO, matrix-interpretations, arctic matrices), dependency pairs, matchbounds, dependency graph decomposition, semantic labeling (predictive labeling, root-labeling), argument filters, subterm criterion, usable rules, size-change termination, bounded increase, string reversal, increasing interpretations, loops, ...

Remark

- proofs are generated by termination tools
- proofs tend to get huge (several MB)
- termination tools may have bugs

State-Of-The-Art Termination Techniques

Overview

reduction orders (polynomial orders, KBO, RPO, matrix-interpretations, arctic matrices), dependency pairs, matchbounds, dependency graph decomposition, semantic labeling (predictive labeling, root-labeling), argument filters, subterm criterion, usable rules, size-change termination, bounded increase, string reversal, increasing interpretations, loops, ...

Remark

- proofs are generated by termination tools
- proofs tend to get huge (several MB)
- termination tools may have bugs
- checking proofs by hand is infeasible

State-Of-The-Art Termination Techniques

Overview

reduction orders (polynomial orders, KBO, RPO, matrix-interpretations, arctic matrices), dependency pairs, matchbounds, dependency graph decomposition, semantic labeling (predictive labeling, root-labeling), argument filters, subterm criterion, usable rules, size-change termination, bounded increase, string reversal, increasing interpretations, loops, ...

Remark

- proofs are generated by termination tools
- proofs tend to get huge (several MB)
- termination tools may have bugs
- checking proofs by hand is infeasible
- automatic certification of proofs

State-Of-The-Art Termination Techniques

not supported

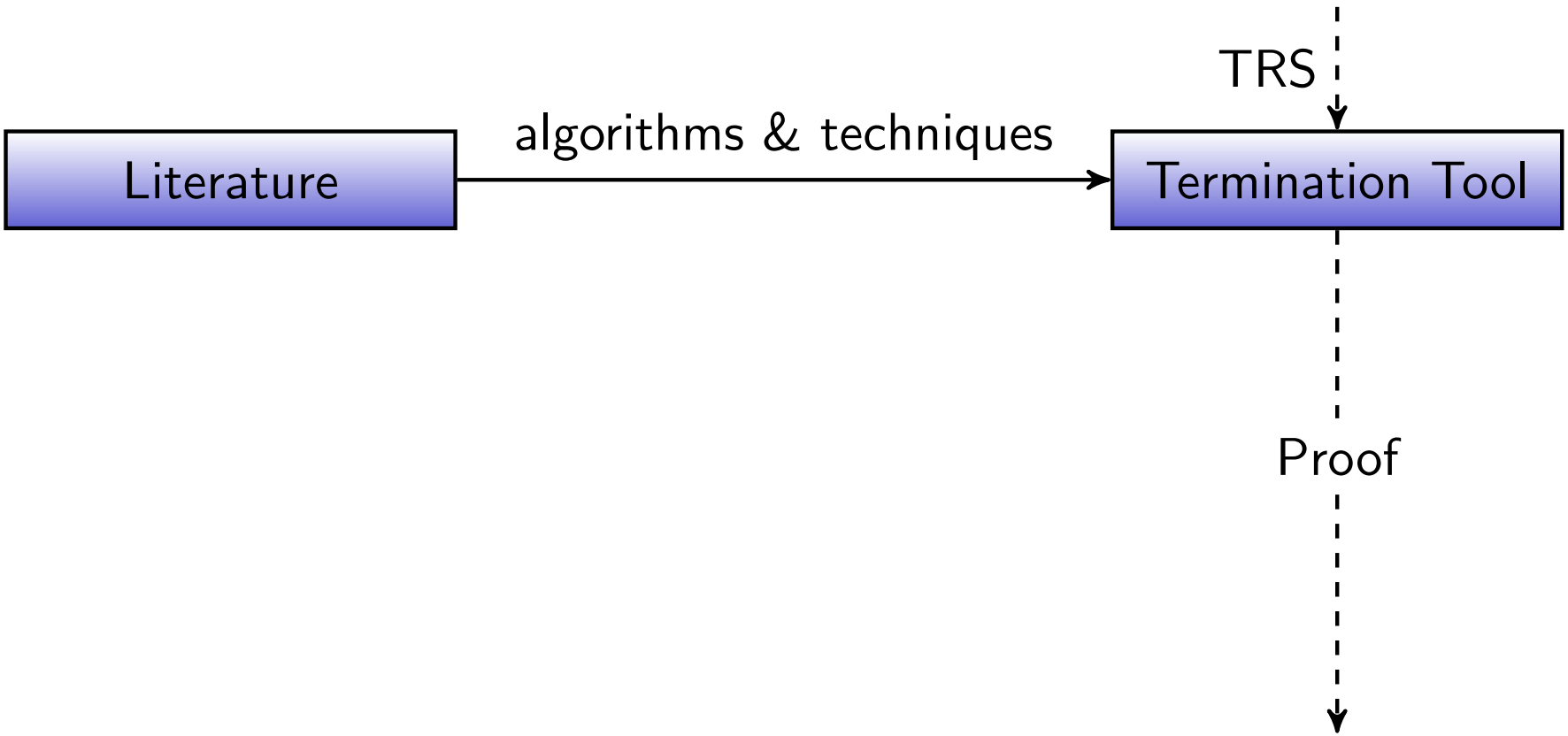
Overview

reduction orders (polynomial orders, KBO, RPO, matrix-interpretations, arctic matrices), dependency pairs, matchbounds, dependency graph decomposition, semantic labeling (predictive labeling, root-labeling), argument filters, subterm criterion, usable rules, size-change termination, bounded increase, string reversal, increasing interpretations, loops, ...

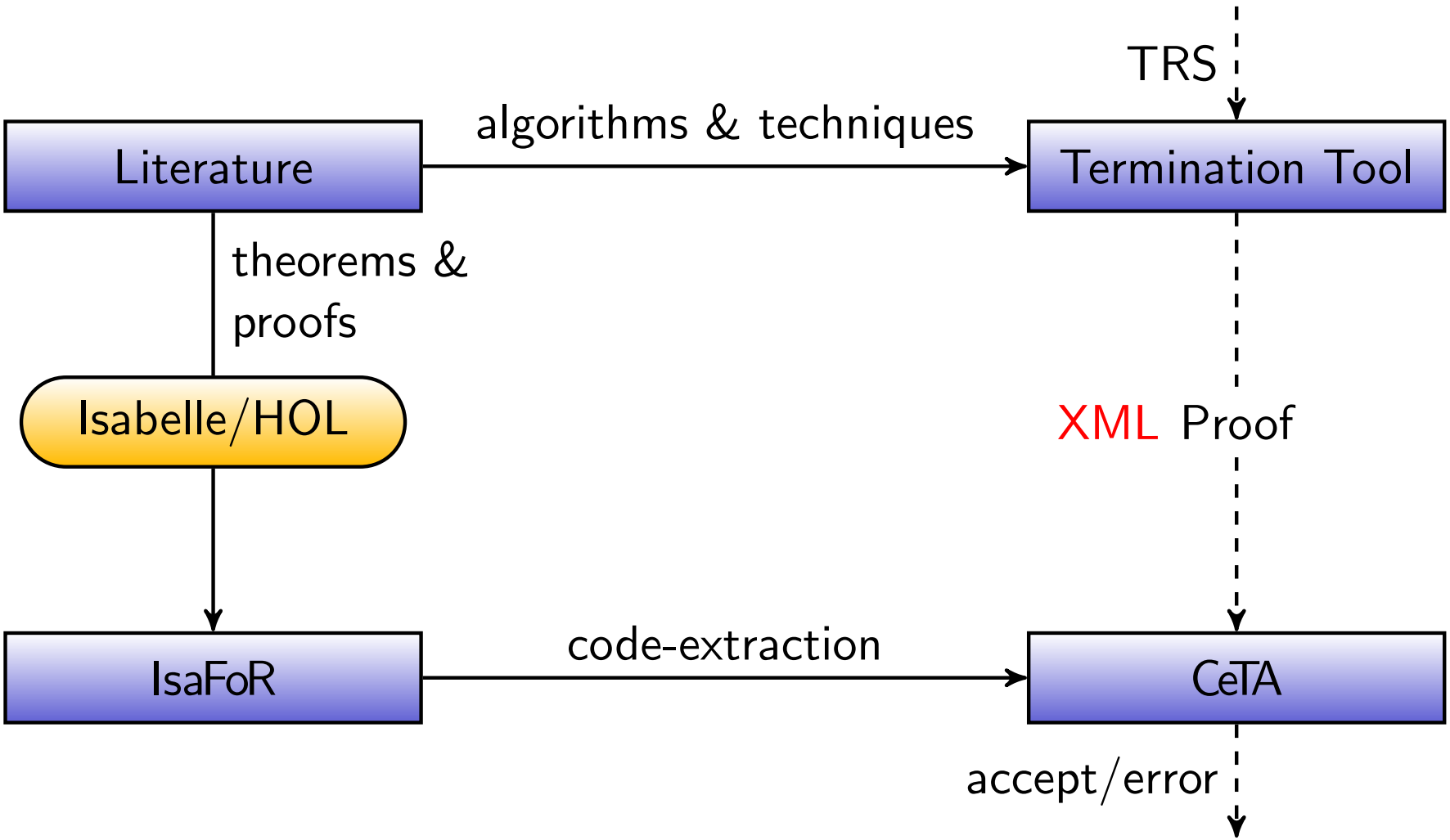
Remark

- proofs are generated by termination tools
- proofs tend to get huge (several MB)
- termination tools may have bugs
- checking proofs by hand is infeasible
- automatic certification of proofs

General View



General View



Overview

- Motivation
- Preliminaries
  - Isabelle/HOL
  - Code-Extraction
  - IsaFoR
- Termination Techniques
  - Dependency Pairs
  - Subterm Criterion
  - Root-Labeling
- CeTA

HOL = Functional Programming + Logic

Functional Programming in HOL

- algebraic data types
- abbreviations/definitions
- (total) recursive functions

HOL = Functional Programming + Logic

Functional Programming in HOL

- algebraic data types
- abbreviations/definitions
- (total) recursive functions

Example

datatype (α, β) term = Var β | Fun α $((\alpha, \beta)$ term list)

HOL = Functional Programming + Logic

Functional Programming in HOL

- algebraic data types
- abbreviations/definitions
- (total) recursive functions

Example

datatype (α, β) term = Var β | Fun α $((\alpha, \beta)$ term list)

HOL = Functional Programming + Logic

Functional Programming in HOL

- algebraic data types
- abbreviations/definitions
- (total) recursive functions

Example

datatype (α, β) term = Var β | Fun α ((α, β) term list)

definition vars_rule_list :: (α, β) rule $\Rightarrow$ β list
where vars_rule_list r = vars_term_list (fst r) @ vars_term_list (snd r)

HOL = Functional Programming + Logic

Functional Programming in HOL

- algebraic data types
- abbreviations/definitions
- (total) recursive functions

Example

```
datatype (α, β) term = Var β | Fun α ((α, β) term list)

definition vars_rule_list :: (α, β) rule ⇒ β list
where vars_rule_list r = vars_term_list (fst r) @ vars_term_list (snd r)

fun vars_term_list :: (α, β) term ⇒ β list
where
  vars_term_list (Var x) = [x] |
  vars_term_list (Fun f ts) = concat (map vars_term_list ts)
```

HOL = Functional Programming + Logic

Logic in HOL

- logical connectives: $=$, $\forall$, $\exists$, $\wedge$, $\vee$, $\neg$, $\longrightarrow$, True, False, ...
- abstract data types: *set*, $\Rightarrow$, ...
- natural deduction, induction, ...

HOL = Functional Programming + Logic

Logic in HOL

- logical connectives: $=$, $\forall$, $\exists$, $\wedge$, $\vee$, $\neg$, $\longrightarrow$, True, False, ...
- abstract data types: *set*, $\Rightarrow$, ...
- natural deduction, induction, ...

Example (Abstract Specification vs. Concrete Implementation)

```
fun vars_term :: ('α, 'β) term ⇒ 'β set
where
  vars_term (Var x) = {x}
  vars_term (Fun f ts) = ⋃set (map vars_term ts)
```

HOL = Functional Programming + Logic

Logic in HOL

- logical connectives: $=$, $\forall$, $\exists$, $\wedge$, $\vee$, $\neg$, $\longrightarrow$, True, False, ...
- abstract data types: *set*, $\Rightarrow$, ...
- natural deduction, induction, ...

Example (Abstract Specification vs. Concrete Implementation)

```
fun vars_term :: ('α, 'β) term ⇒ 'β set
where
  vars_term (Var x) = {x}
  vars_term (Fun f ts) = ⋃set (map vars_term ts)
```

HOL = Functional Programming + Logic

Logic in HOL

- logical connectives: $=$, $\forall$, $\exists$, $\wedge$, $\vee$, $\neg$, $\longrightarrow$, True, False, ...
- abstract data types: *set*, $\Rightarrow$, ...
- natural deduction, induction, ...

Example (Abstract Specification vs. Concrete Implementation)

```
fun vars_term :: (α, β) term ⇒ β set
where
  vars_term (Var x) = {x}
  vars_term (Fun f ts) = ⋃set (map vars_term ts)

lemma vars_term_list_sound: set (vars_term_list t) = vars_term t
by (induct t) simp_all
```

Generating Fully-Verified Executable Code

demo

Abstract Rewriting

Abstract Rewrite Systems

in IsaFoR: α ars $= (\alpha \times \alpha)$ set

Termination / Strong Normalization

$$\mathsf{SN}_{\mathcal{A}}(a) \equiv \nexists S.\, S\ 0 = a \wedge (\forall i.\, (S\ i, S\ (i+1)) \in \mathcal{A})$$
$$\mathsf{SN}(\mathcal{A}) \equiv \forall a.\, \mathsf{SN}_{\mathcal{A}}(a)$$

Abstract Rewriting

Abstract Rewrite Systems

in IsaFoR: α ars $= (\alpha \times \alpha)$ set

Termination / Strong Normalization

$$\mathsf{SN}_{\mathcal{A}}(a) \equiv \nexists S.\, S\ 0 = a \wedge (\forall i.\, (S\ i, S\ (i+1)) \in \mathcal{A})$$
$$\mathsf{SN}(\mathcal{A}) \equiv \forall a.\, \mathsf{SN}_{\mathcal{A}}(a)$$

Other Properties and Results

- (local) confluence
- Newman's Lemma

Term Rewriting

Terms

- datatype (α, β) term $=$ Var β | Fun α $((\alpha, \beta)$ term list)
- no well-formedness restrictions
- same function symbol may be used with different arities

Term Rewriting

Terms

- datatype (α, β) term $=$ Var β | Fun α $((\alpha, \beta)$ term list)
- no well-formedness restrictions
- same function symbol may be used with different arities

Example ((string, string) term)

$$x \approx \text{Var “x”}$$
$$\text{sum}(\text{s}(x)) \approx \text{Fun “sum” [Fun “s” [Var “x”]]}$$
$$x + y \approx \text{Fun “+” [Var “x”, Var “y”]}$$
$$\text{f}(\text{f}(x), \text{f}) \approx \text{Fun “f” [Fun “f” [Var “x”], Fun “f” []]}$$

Term Rewriting (cont'd)

Term Rewrite Systems

- (α, β) rule $= (\alpha, \beta)$ term $\times$ (α, β) term
- (α, β) trs $= (\alpha, \beta)$ rule set
- well-formed TRSs
  $wf\_trs\ \mathcal{R} \equiv \forall (l, r) \in \mathcal{R}.\ vars\_term\ r \subseteq vars\_term\ l \wedge \neg\ is\_var\ l$

Term Rewriting (cont'd)

Term Rewrite Systems

- (α, β) rule $= (\alpha, \beta)$ term $\times$ (α, β) term
- (α, β) trs $= (\alpha, \beta)$ rule set
- well-formed TRSs
  wf_trs $\mathcal{R} \equiv \forall (l, r) \in \mathcal{R}.$ *vars_term r* $\subseteq$ *vars_term l* $\wedge \neg$ *is_var l*

Example

{ (Fun sum [Fun 0 []], Fun 0),
(Fun sum [Fun s [Var n]], Fun + [Fun s [Var n],Fun sum [Var n]])}

Term Rewriting (cont'd)

Contexts

- datatype (α, β) ctxt $= \Box \mid$
  More α $((\alpha, \beta)$ term list$)$ $((\alpha, \beta)$ ctxt$)$ $((\alpha, \beta)$ term list$)$
- by definition: exactly one "hole"
- applying contexts to terms

$$\Box[t] = t$$
$$(\mathit{More}\ f\ ss\ C\ ts)[t] = \mathit{Fun}\ f\ (ss\ @\ C[t]\ \#\ ts)$$

Term Rewriting (cont'd)

Contexts

- datatype (α, β) ctxt $= \Box \mid$
  More α $((\alpha, \beta)$ term list$)$ $((\alpha, \beta)$ ctxt$)$ $((\alpha, \beta)$ term list$)$
- by definition: exactly one "hole"
- applying contexts to terms

$$\Box[t] = t$$
$$(\mathit{More}\ f\ ss\ C\ ts)[t] = \mathit{Fun}\ f\ (ss\ @\ C[t]\ \#\ ts)$$

Subterms

$$s \trianglerighteq t \equiv \exists C.\, s = C[t]$$
$$s \triangleright t \equiv \exists C.\, C \neq \Box \wedge s = C[t]$$

Term Rewriting (cont'd)

Substitutions

- datatype (α, β) subst = Subst $(\beta \Rightarrow (\alpha, \beta)$ term)
- auxiliary function: get_subst (Subst s) = s
- applying substitutions to terms

$$(Var\ x)\sigma = get\_subst\ \sigma\ x$$
$$(Fun\ f\ \ ts)\sigma = Fun\ f\ (map\ (\lambda t.\ t\sigma)\ ts)$$

Term Rewriting (cont'd)

Substitutions

- datatype (α, β) subst = Subst $(\beta \Rightarrow (\alpha, \beta)$ term)
- auxiliary function: get_subst (Subst s) = s
- applying substitutions to terms

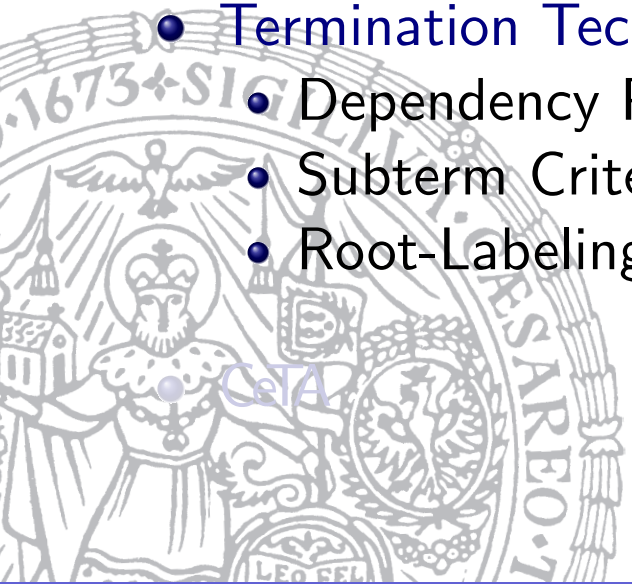
$$(Var\ x)\sigma = get\_subst\ \sigma\ x$$
$$(Fun\ f\ \ ts)\sigma = Fun\ f\ (map\ (\lambda t.\ t\sigma)\ ts)$$

Rewrite Relation

$$\frac{(s,t) \in \mathcal{R}}{s \to_{\mathcal{R}} t} \qquad \frac{s \to_{\mathcal{R}} t}{s\sigma \to_{\mathcal{R}} t\sigma} \qquad \frac{s \to_{\mathcal{R}} t}{C[s] \to_{\mathcal{R}} C[t]}$$

Overview

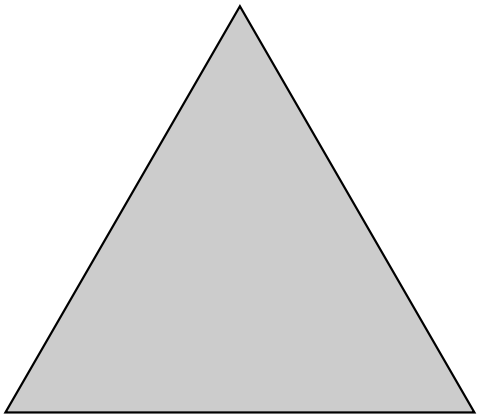
- Motivation
- Preliminaries
  - Isabelle/HOL
  - Code-Extraction
  - IsaFoR
- Termination Techniques
  - Dependency Pairs
  - Subterm Criterion
  - Root-Labeling
- CeTA



Minimal Counterexamples

Observations

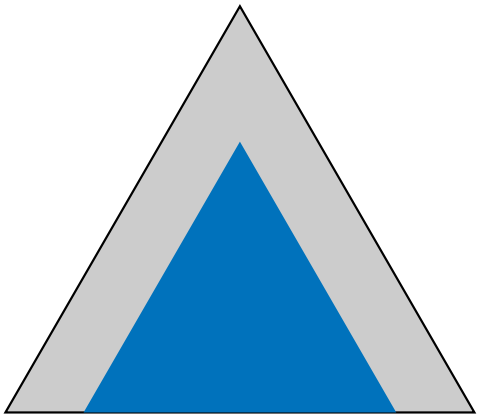
- minimal nontermination
  $t \in \mathcal{T}^{\infty}_{\mathcal{R}} \equiv \neg\,\mathsf{SN}_{\mathcal{R}}(t) \wedge (\forall s \lhd t.\, \mathsf{SN}_{\mathcal{R}}(s))$
- every nonterminating term has subterm in $\mathcal{T}^{\infty}_{\mathcal{R}}$
- in every infinite $\mathcal{R}$-sequence starting at some $t \in \mathcal{T}^{\infty}_{\mathcal{R}}$, eventually, an $\mathcal{R}$-step at the root takes place



Minimal Counterexamples

Observations

- minimal nontermination
  $t \in \mathcal{T}^{\infty}_{\mathcal{R}} \equiv \neg\, \mathsf{SN}_{\mathcal{R}}(t) \wedge (\forall s \lhd t.\, \mathsf{SN}_{\mathcal{R}}(s))$
- every nonterminating term has subterm in $\mathcal{T}^{\infty}_{\mathcal{R}}$
- in every infinite $\mathcal{R}$-sequence starting at some $t \in \mathcal{T}^{\infty}_{\mathcal{R}}$, eventually, an $\mathcal{R}$-step at the root takes place

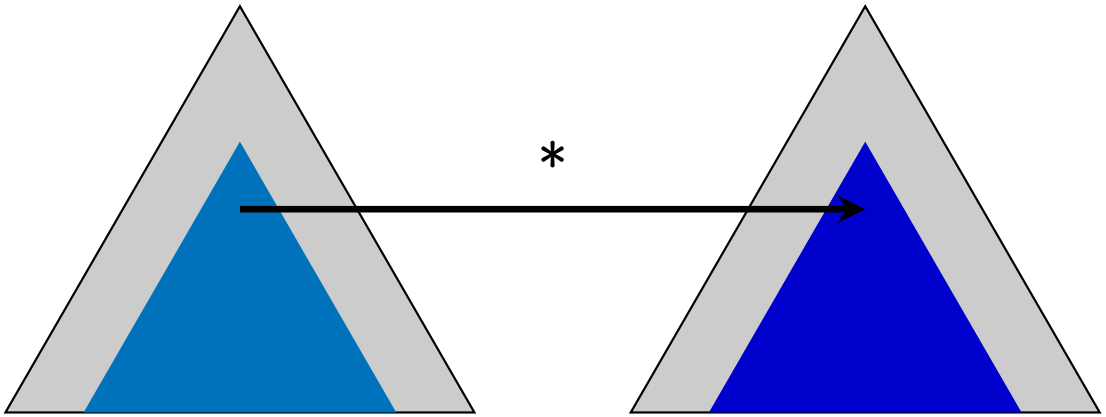


$t \in \mathcal{T}^{\infty}_{\mathcal{R}}$

Minimal Counterexamples

Observations

- minimal nontermination
  $t \in \mathcal{T}^\infty_\mathcal{R} \equiv \neg\,\mathsf{SN}_\mathcal{R}(t) \wedge (\forall s \lhd t.\, \mathsf{SN}_\mathcal{R}(s))$
- every nonterminating term has subterm in $\mathcal{T}^\infty_\mathcal{R}$
- in every infinite $\mathcal{R}$-sequence starting at some $t \in \mathcal{T}^\infty_\mathcal{R}$, eventually, an $\mathcal{R}$-step at the root takes place

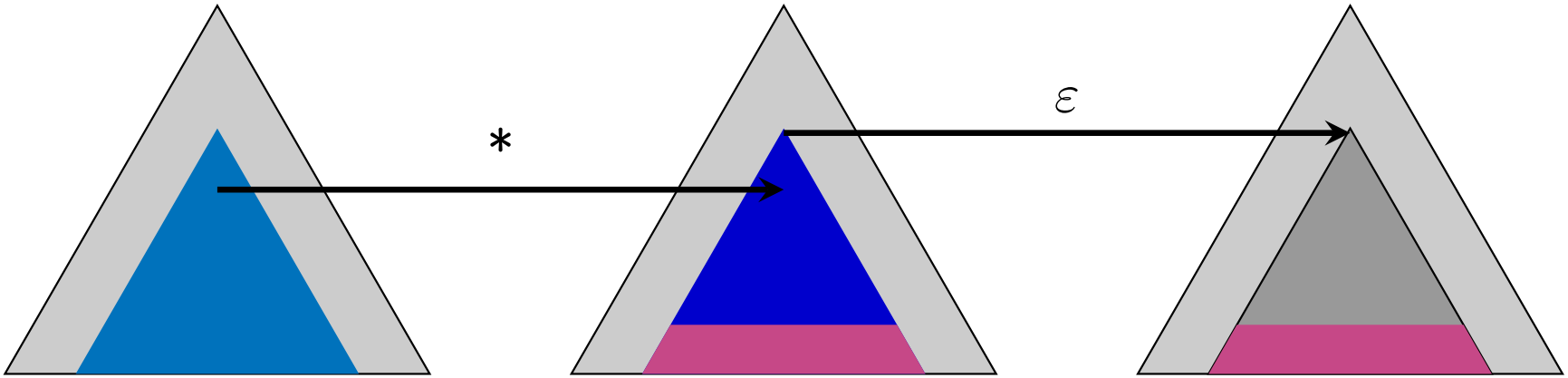


$t \in \mathcal{T}^\infty_\mathcal{R}$

Minimal Counterexamples

Observations

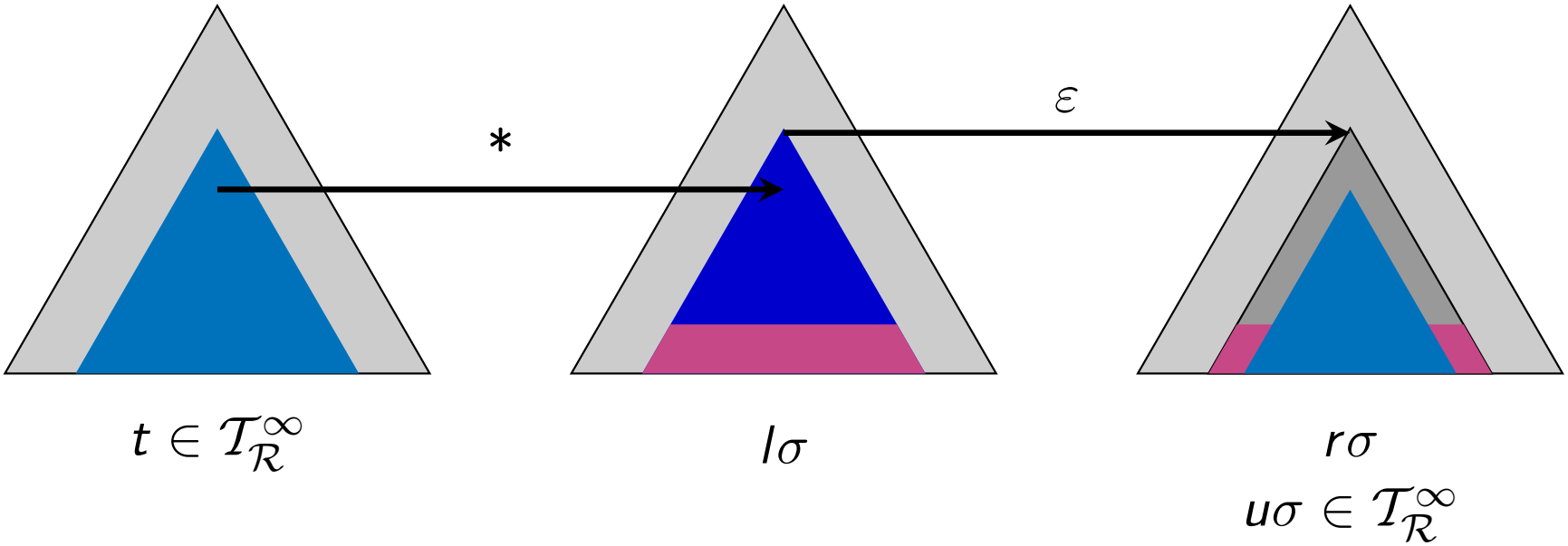
- minimal nontermination
  $t \in \mathcal{T}^\infty_\mathcal{R} \equiv \neg\,\mathsf{SN}_\mathcal{R}(t) \wedge (\forall s \lhd t.\, \mathsf{SN}_\mathcal{R}(s))$
- every nonterminating term has subterm in $\mathcal{T}^\infty_\mathcal{R}$
- in every infinite $\mathcal{R}$-sequence starting at some $t \in \mathcal{T}^\infty_\mathcal{R}$, eventually, an $\mathcal{R}$-step at the root takes place



Minimal Counterexamples

Observations

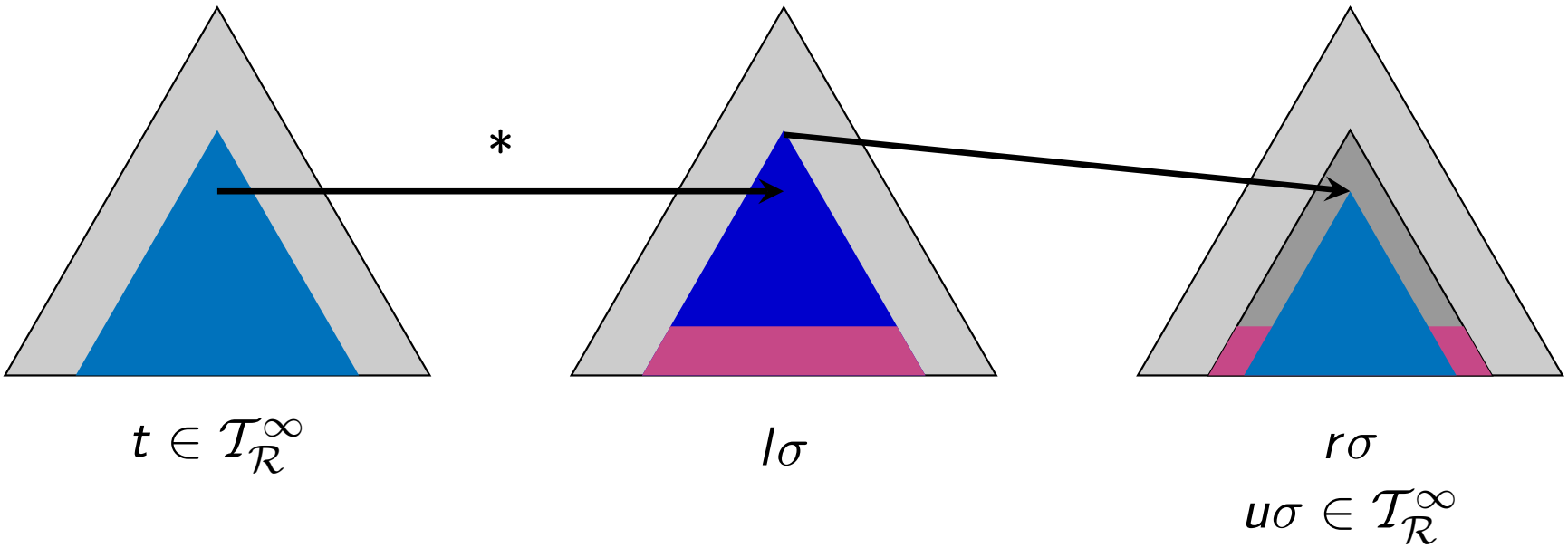
- minimal nontermination
  $t \in \mathcal{T}^\infty_\mathcal{R} \equiv \neg\,\mathsf{SN}_\mathcal{R}(t) \wedge (\forall s \lhd t.\, \mathsf{SN}_\mathcal{R}(s))$
- every nonterminating term has subterm in $\mathcal{T}^\infty_\mathcal{R}$
- in every infinite $\mathcal{R}$-sequence starting at some $t \in \mathcal{T}^\infty_\mathcal{R}$, eventually, an $\mathcal{R}$-step at the root takes place



Minimal Counterexamples

Observations

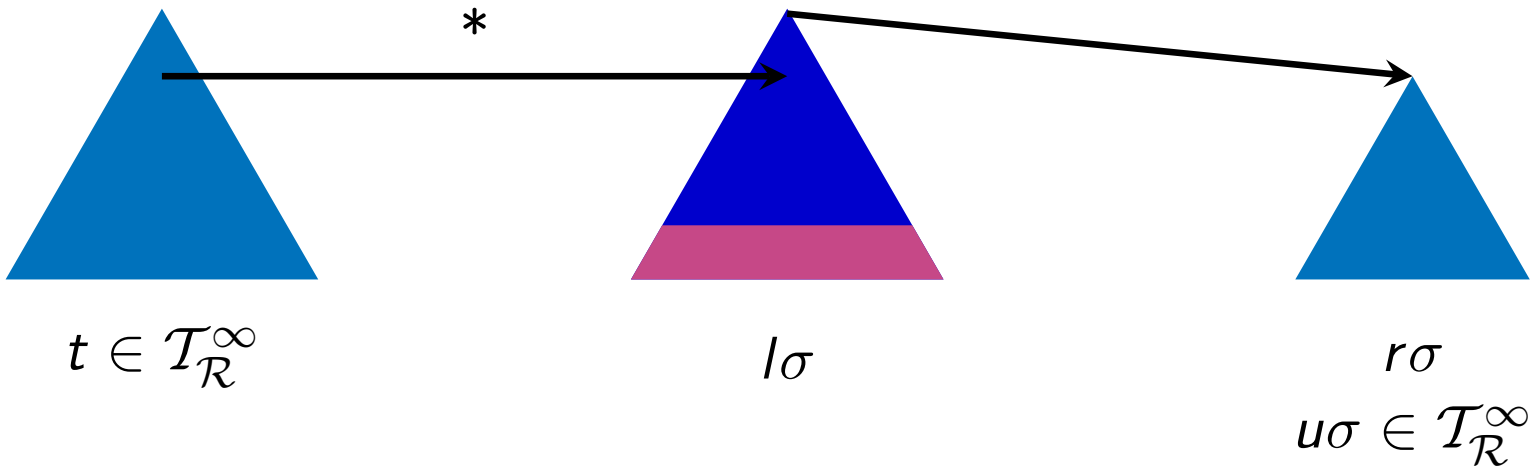
- minimal nontermination
  $t \in \mathcal{T}^\infty_\mathcal{R} \equiv \neg\,\mathsf{SN}_\mathcal{R}(t) \wedge (\forall s \lhd t.\, \mathsf{SN}_\mathcal{R}(s))$
- every nonterminating term has subterm in $\mathcal{T}^\infty_\mathcal{R}$
- in every infinite $\mathcal{R}$-sequence starting at some $t \in \mathcal{T}^\infty_\mathcal{R}$, eventually, an $\mathcal{R}$-step at the root takes place



Minimal Counterexamples

Observations

- minimal nontermination
  $t \in \mathcal{T}^\infty_\mathcal{R} \equiv \neg\,\mathsf{SN}_\mathcal{R}(t) \wedge (\forall s \lhd t.\, \mathsf{SN}_\mathcal{R}(s))$
- every nonterminating term has subterm in $\mathcal{T}^\infty_\mathcal{R}$
- in every infinite $\mathcal{R}$-sequence starting at some $t \in \mathcal{T}^\infty_\mathcal{R}$, eventually, an $\mathcal{R}$-step at the root takes place



Dependency Pairs

Idea

- possible (mutually) recursive dependencies
- ignore call-context (i.e., code around recursive call)
- if there is no infinite chain of DPs, then termination

Dependency Pairs

Idea

- possible (mutually) recursive dependencies
- ignore call-context (i.e., code around recursive call)
- if there is no infinite chain of DPs, then termination

Definition

- DP problem is pair of TRSs $(\mathcal{P}, \mathcal{R})$
- $(\mathcal{P}, \mathcal{R})$ is finite iff no infinite sequence

$$s_1\sigma_1 \xrightarrow{\varepsilon}_{\mathcal{P}} t_1\sigma_1 \to^*_{\mathcal{R}} s_2\sigma_2 \xrightarrow{\varepsilon}_{\mathcal{P}} \cdots \quad (\forall i.\, \mathsf{SN}_{\mathcal{R}}(t_i\sigma_i))$$

Dependency Pairs

Idea

- possible (mutually) recursive dependencies
- ignore call-context (i.e., code around recursive call)
- if there is no infinite chain of DPs, then termination

Definition

- DP problem is pair of TRSs $(\mathcal{P}, \mathcal{R})$
- $(\mathcal{P}, \mathcal{R})$ is finite iff no infinite sequence

$$s_1\sigma_1 \xrightarrow{\varepsilon}_{\mathcal{P}} t_1\sigma_1 \to^*_{\mathcal{R}} s_2\sigma_2 \xrightarrow{\varepsilon}_{\mathcal{P}} \cdots \quad (\forall i.\, \mathsf{SN}_{\mathcal{R}}(t_i\sigma_i))$$

Dependency Pairs

Idea

- possible (mutually) recursive dependencies
- ignore call-context (i.e., code around recursive call)
- if there is no infinite chain of DPs, then termination

Definition

- DP problem is pair of TRSs $(\mathcal{P}, \mathcal{R})$
- $(\mathcal{P}, \mathcal{R})$ is finite iff no infinite sequence

$$s_1\sigma_1 \xrightarrow{\varepsilon}_{\mathcal{P}} t_1\sigma_1 \to^*_{\mathcal{R}} s_2\sigma_2 \xrightarrow{\varepsilon}_{\mathcal{P}} \cdots \quad (\forall i.\, \mathsf{SN}_{\mathcal{R}}(t_i\sigma_i))$$

Theorem

finiteness of $(DP(\mathcal{R}), \mathcal{R})$ *implies* $SN(\mathcal{R})$

Example

Recall

$$\mathsf{sum}(0) \to 0$$
$$\mathsf{sum}(\mathsf{s}(n)) \to \mathsf{s}(n) + \mathsf{sum}(n)$$

Example

Recall

$$\mathsf{sum}(0) \to 0$$
$$\mathsf{sum}(\mathsf{s}(n)) \to \mathsf{s}(n) + \mathsf{sum}(n)$$

DPs

$$\mathsf{SUM}(\mathsf{s}(n)) \to \mathsf{SUM}(n)$$

DPs in IsaFoR

Remarks

- typed setting (Isabelle), hence DP($\mathcal{R}$) is not compatible to $\mathcal{R}$
- IsaFoR's definition of finiteness

$$\begin{aligned}&\mathsf{finite}(\mathcal{P},\mathcal{R}) \equiv \nexists s\ t\ \sigma.\forall i.\\ &\quad (s_i,t_i)\in\mathcal{P}\wedge t_i\sigma_i \to^*_{\mathcal{R}} s_{i+1}\sigma_{i+1}\wedge \mathsf{SN}_{\mathcal{R}}(t_i\sigma_i)\end{aligned}$$

Interlude - Signature Extensions

Definition (Signatures in IsaFoR)

- function symbols of a term

$$\mathcal{F}\mathrm{un}(\mathit{Var}\ x) = \{\}$$

$$\mathcal{F}\mathrm{un}(\mathit{Fun}\ f\ \ ts) = \{(f, \mathit{length}\ ts)\} \cup \bigcup \mathit{set}(\mathit{map}\ \mathcal{F}\mathrm{un}(\cdot)\ ts)$$

- signature of TRS $\mathcal{R}$

$$\mathcal{F}\mathrm{un}(\mathcal{R}) = \{\mathcal{F}\mathrm{un}(l), \mathcal{F}\mathrm{un}(r) \mid (l, r) \in \mathcal{R}\}$$

Interlude - Signature Extensions

Definition (Signatures in IsaFoR)

- function symbols of a term

$$\mathcal{F}\mathrm{un}(\mathit{Var}\ x) = \{\}$$
$$\mathcal{F}\mathrm{un}(\mathit{Fun}\ f\ \ ts) = \{(f, \mathit{length}\ ts)\} \cup \bigcup \mathit{set}(\mathit{map}\ \mathcal{F}\mathrm{un}(\cdot)\ ts)$$

- signature of TRS $\mathcal{R}$

$$\mathcal{F}\mathrm{un}(\mathcal{R}) = \{\mathcal{F}\mathrm{un}(l), \mathcal{F}\mathrm{un}(r) \mid (l, r) \in \mathcal{R}\}$$

in IsaFoR

Interlude - Signature Extensions

Definition (Signatures in IsaFoR)

- function symbols of a term

$$\mathcal{F}\mathrm{un}(\mathit{Var}\ x) = \{\}$$
$$\mathcal{F}\mathrm{un}(\mathit{Fun}\ f\ \ ts) = \{(f, \mathit{length}\ ts)\} \cup \bigcup \mathit{set}(\mathit{map}\ \mathcal{F}\mathrm{un}(\cdot)\ ts)$$

- signature of TRS $\mathcal{R}$

$$\mathcal{F}\mathrm{un}(\mathcal{R}) = \{\mathcal{F}\mathrm{un}(l), \mathcal{F}\mathrm{un}(r) \mid (l, r) \in \mathcal{R}\}$$

in IsaFoR

- signature extensions preserve termination

Interlude - Signature Extensions

Definition (Signatures in IsaFoR)

- function symbols of a term

$$\mathcal{F}\mathrm{un}(\mathit{Var}\ x) = \{\}$$
$$\mathcal{F}\mathrm{un}(\mathit{Fun}\ f\ \ ts) = \{(f, \mathit{length}\ ts)\} \cup \bigcup \mathit{set}(\mathit{map}\ \mathcal{F}\mathrm{un}(\cdot)\ ts)$$

- signature of TRS $\mathcal{R}$

$$\mathcal{F}\mathrm{un}(\mathcal{R}) = \{\mathcal{F}\mathrm{un}(l), \mathcal{F}\mathrm{un}(r) \mid (l, r) \in \mathcal{R}\}$$

in IsaFoR

- signature extensions preserve termination
- signature extensions preserve finiteness

Interlude - Signature Extensions

Definition (Signatures in IsaFoR)

- function symbols of a term

$$\mathcal{F}\mathrm{un}(\mathit{Var}\ x) = \{\}$$
$$\mathcal{F}\mathrm{un}(\mathit{Fun}\ f\ \ ts) = \{(f, \mathit{length}\ ts)\} \cup \bigcup \mathit{set}(\mathit{map}\ \mathcal{F}\mathrm{un}(\cdot)\ ts)$$

- signature of TRS $\mathcal{R}$

$$\mathcal{F}\mathrm{un}(\mathcal{R}) = \{\mathcal{F}\mathrm{un}(l), \mathcal{F}\mathrm{un}(r) \mid (l, r) \in \mathcal{R}\}$$

in IsaFoR

- signature extensions preserve termination
- signature extensions preserve finiteness for left-linear $\mathcal{R}$

Interlude - Signature Extensions

Definition (Signatures in IsaFoR)

- function symbols of a term

$$\mathcal{F}\mathrm{un}(\mathit{Var}\ x) = \{\}$$
$$\mathcal{F}\mathrm{un}(\mathit{Fun}\ f\ \ ts) = \{(f, \mathit{length}\ ts)\} \cup \bigcup \mathit{set}(\mathit{map}\ \mathcal{F}\mathrm{un}(\cdot)\ ts)$$

- signature of TRS $\mathcal{R}$

$$\mathcal{F}\mathrm{un}(\mathcal{R}) = \{\mathcal{F}\mathrm{un}(l), \mathcal{F}\mathrm{un}(r) \mid (l, r) \in \mathcal{R}\}$$

in IsaFoR

- signature extensions preserve termination
- signature extensions preserve finiteness for left-linear $\mathcal{R}$ or when dropping minimality

Subterm Criterion

Idea

- well-founded decrease in structural size of arguments
- subsumes primitive recursion

Subterm Criterion

Idea

- well-founded decrease in structural size of arguments
- subsumes primitive recursion

Theorem (DP Processor)

- *DP problem* $(\mathcal{P}, \mathcal{R})$
- *simple projection* π*, s.t.,* $\mathcal{P} \subseteq \trianglerighteq_\pi$
- *finiteness of* $(\mathcal{P} \setminus \triangleright_\pi, \mathcal{R})$ *implies finiteness of* $(\mathcal{P}, \mathcal{R})$

Subterm Criterion

Idea

- well-founded decrease in structural size of arguments
- subsumes primitive

$$\forall (f, n) \in \mathcal{F}.\, \pi(f) \in \{1, \ldots, n\}$$

Theorem (DP Processor)

- *DP problem* $(\mathcal{P}, \mathcal{R})$
- *simple projection* π, *s.t.*, $\mathcal{P} \subseteq \trianglerighteq_\pi$
- *finiteness of* $(\mathcal{P} \setminus \triangleright_\pi, \mathcal{R})$ *implies finiteness of* $(\mathcal{P}, \mathcal{R})$

Subterm Criterion

Idea

- well-founded decrease in structural size of arguments
- subsumes primitive

$$\forall (f, n) \in \mathcal{F}.\, \pi(f) \in \{1, \ldots, n\}$$

Theorem (DP Processor)

$$\pi(f(t_1, \ldots, t_n)) = t_{\pi(f)}$$

- *DP problem* $(\mathcal{P}, \mathcal{R})$
- *simple projection* π*, s.t.,* $\mathcal{P} \subseteq \trianglerighteq_\pi$
- *finiteness of* $(\mathcal{P} \setminus \triangleright_\pi, \mathcal{R})$ *implies finiteness of* $(\mathcal{P}, \mathcal{R})$

Example

DP Problem

- $\mathsf{DP}(\mathcal{R}) = \{\mathsf{SUM}(\mathsf{s}(n)) \to \mathsf{SUM}(n)\}$
- $\mathcal{R} = \left\{ \begin{array}{c} \mathsf{sum}(0) \to 0 \\ \mathsf{sum}(\mathsf{s}(n)) \to \mathsf{s}(n) + \mathsf{sum}(n) \end{array} \right\}$

Example

DP Problem

- $\mathrm{DP}(\mathcal{R}) = \{\mathsf{SUM}(\mathsf{s}(n)) \to \mathsf{SUM}(n)\}$
- $\mathcal{R} = \left\{ \begin{array}{c} \mathsf{sum}(0) \to 0 \\ \mathsf{sum}(\mathsf{s}(n)) \to \mathsf{s}(n) + \mathsf{sum}(n) \end{array} \right\}$

Subterm Criterion

- $\pi(\mathsf{SUM}) = 1$
- $\pi(\mathsf{SUM}(\mathsf{s}(n))) = \mathsf{s}(n) \rhd n = \pi(\mathsf{SUM}(n))$
- remaining DP problem $(\varnothing, \mathcal{R})$

The Subterm Criterion in IsaFoR

Processor

1. $\forall (s,t) \in \mathcal{P} \setminus \mathcal{P}'.\, \pi(s) \rhd \pi(t)$
2. $\forall (s,t) \in \mathcal{P}'.\, \pi(s) = \pi(t)$

then finiteness of $(\mathcal{P}', \mathcal{R})$ implies finiteness of $(\mathcal{P}, \mathcal{R})$

The Subterm Criterion in IsaFoR

Cheating! Dropped further conditions

Processor

1. $\forall (s,t) \in \mathcal{P} \setminus \mathcal{P}'.\, \pi(s) \rhd \pi(t)$
2. $\forall (s,t) \in \mathcal{P}'.\, \pi(s) = \pi(t)$

then finiteness of $(\mathcal{P}', \mathcal{R})$ implies finiteness of $(\mathcal{P}, \mathcal{R})$

The Subterm Criterion in IsaFoR

Processor

1. $\forall (s,t) \in \mathcal{P} \setminus \mathcal{P}'.\, \pi(s) \rhd \pi(t)$
2. $\forall (s,t) \in \mathcal{P}'.\, \pi(s) = \pi(t)$

then finiteness of $(\mathcal{P}', \mathcal{R})$ implies finiteness of $(\mathcal{P}, \mathcal{R})$

Proof Sketch (assuming $(\mathcal{P}, \mathcal{R})$ is not finite).

The Subterm Criterion in IsaFoR

Processor

1. $\forall (s,t) \in \mathcal{P} \setminus \mathcal{P}'.\, \pi(s) \rhd \pi(t)$
2. $\forall (s,t) \in \mathcal{P}'.\, \pi(s) = \pi(t)$

then finiteness of $(\mathcal{P}', \mathcal{R})$ implies finiteness of $(\mathcal{P}, \mathcal{R})$

Proof Sketch (assuming $(\mathcal{P}, \mathcal{R})$ is not finite).

- $\exists j.\, \forall i > j.\, (s_i, t_i) \in \mathcal{P}'$: then $(\mathcal{P}', \mathcal{R})$ not finite

The Subterm Criterion in IsaFoR

Processor

1. $\forall (s,t) \in \mathcal{P} \setminus \mathcal{P}'.\, \pi(s) \rhd \pi(t)$
2. $\forall (s,t) \in \mathcal{P}'.\, \pi(s) = \pi(t)$

then finiteness of $(\mathcal{P}', \mathcal{R})$ implies finiteness of $(\mathcal{P}, \mathcal{R})$

Proof Sketch (assuming $(\mathcal{P}, \mathcal{R})$ is not finite).

- $\exists j.\, \forall i > j.\, (s_i, t_i) \in \mathcal{P}'$: then $(\mathcal{P}', \mathcal{R})$ not finite
- choice function f, s.t., $f(i) > i$ and $(s_{f(i)}, t_{f(i)}) \in \mathcal{P} \setminus \mathcal{P}'$
  let $[i] = f^{i+1}(0)$ and $u_i = \pi(s_{[i]})\sigma_{[i]}$

The Subterm Criterion in IsaFoR

Processor

1. $\forall (s,t) \in \mathcal{P} \setminus \mathcal{P}'.\, \pi(s) \rhd \pi(t)$
2. $\forall (s,t) \in \mathcal{P}'.\, \pi(s) = \pi(t)$

then finiteness of $(\mathcal{P}', \mathcal{R})$ implies finiteness of $(\mathcal{P}, \mathcal{R})$

Proof Sketch (assuming $(\mathcal{P}, \mathcal{R})$ is not finite).

- $\exists j.\, \forall i > j.\, (s_i, t_i) \in \mathcal{P}'$: then $(\mathcal{P}', \mathcal{R})$ not finite
- choice function f, s.t., $f(i) > i$ and $(s_{f(i)}, t_{f(i)}) \in \mathcal{P} \setminus \mathcal{P}'$
  let $[i] = f^{i+1}(0)$ and $u_i = \pi(s_{[i]})\sigma_{[i]}$
  - $u_i \; (\rhd \cup \to_{\mathcal{R}})^+ \; \pi(t_{[i]})\sigma_{[i]}$

The Subterm Criterion in IsaFoR

Processor

1. $\forall (s,t) \in \mathcal{P} \setminus \mathcal{P}'.\, \pi(s) \rhd \pi(t)$
2. $\forall (s,t) \in \mathcal{P}'.\, \pi(s) = \pi(t)$

then finiteness of $(\mathcal{P}', \mathcal{R})$ implies finiteness of $(\mathcal{P}, \mathcal{R})$

Proof Sketch (assuming $(\mathcal{P}, \mathcal{R})$ is not finite).

- $\exists j.\, \forall i > j.\, (s_i, t_i) \in \mathcal{P}$. $(\mathcal{P}, \mathcal{R})$ not finite
- choice function f, s.t. $f(i) > i$ and $(s_{f(i)}, t_{f(i)}) \in \mathcal{P} \setminus \mathcal{P}'$
  let $[i] = f^{i+1}(0)$ and $u_i = \pi(s_{[i]})\sigma_{[i]}$
  - $u_i \; (\rhd \cup \to_{\mathcal{R}})^+ \; \pi(t_{[i]})\sigma_{[i]}$

$(s_{[i]}, t_{[i]}) \in \mathcal{P} \setminus \mathcal{P}'$

The Subterm Criterion in IsaFoR

Processor

1. $\forall (s,t) \in \mathcal{P} \setminus \mathcal{P}'.\, \pi(s) \rhd \pi(t)$
2. $\forall (s,t) \in \mathcal{P}'.\, \pi(s) = \pi(t)$

then finiteness of $(\mathcal{P}', \mathcal{R})$ implies finiteness of $(\mathcal{P}, \mathcal{R})$

Proof Sketch (assuming $(\mathcal{P}, \mathcal{R})$ is not finite).

- $\exists j.\, \forall i > j.\, (s_i, t_i) \in \mathcal{P}'$: then $(\mathcal{P}', \mathcal{R})$ not finite
- choice function f, s.t., $f(i) > i$ and $(s_{f(i)}, t_{f(i)}) \in \mathcal{P} \setminus \mathcal{P}'$
  let $[i] = f^{i+1}(0)$ and $u_i = \pi(s_{[i]})\sigma_{[i]}$
  - $u_i \ (\rhd \cup \to_{\mathcal{R}})^+ \ \pi(t_{[i]})\sigma_{[i]}$
  - $\pi(t_{[i]})\sigma_{[i]} \to_{\mathcal{R}}^* \pi(s_{[i]+1})\sigma_{[i]+1} \ \ (\rhd \cup \to_{\mathcal{R}})^* \ u_{i+1}$

The Subterm Criterion in IsaFoR

Processor

1. $\forall (s,t) \in \mathcal{P} \setminus \mathcal{P}'.\, \pi(s) \rhd \pi(t)$
2. $\forall (s,t) \in \mathcal{P}'.\, \pi(s) = \pi(t)$

then finiteness of $(\mathcal{P}', \mathcal{R})$ implies finiteness of $(\mathcal{P}, \mathcal{R})$

Proof Sketch (assuming $(\mathcal{P}, \mathcal{R})$ is not finite).

- $\exists j.\, \forall i > j.\, (s_i, t_i) \in \mathcal{P}'$: then $(\mathcal{P}', \mathcal{R})$ not finite
- choice function f, s.t., $f(i) > i$ and $(s_{f(i)}, t_{f(i)}) \in \mathcal{P} \setminus \mathcal{P}'$

  let $[i] =$ infinite chain $\pi(s_{[i]})\sigma_{[i]}$
  - $u_i \ (\rhd \cup \to_{\mathcal{R}}) \ \pi(t_{[i]})\sigma_{[i]}$
  - $\pi(t_{[i]})\sigma_{[i]} \to_{\mathcal{R}}^* \pi(s_{[i]+1})\sigma_{[i]+1} \ \ (\rhd \cup \to_{\mathcal{R}})^* \ u_{i+1}$

The Subterm Criterion in IsaFoR

Processor

1. $\forall (s,t) \in \mathcal{P} \setminus \mathcal{P}'.\, \pi(s) \rhd \pi(t)$
2. $\forall (s,t) \in \mathcal{P}'.\, \pi(s) = \pi(t)$

then finiteness of $(\mathcal{P}', \mathcal{R})$ implies finiteness of $(\mathcal{P}, \mathcal{R})$

Proof Sketch (assuming $(\mathcal{P}, \mathcal{R})$ is not finite).

- $\exists j.\, \forall i > j.\, (s_i, t_i) \in \mathcal{P}'$: then $(\mathcal{P}', \mathcal{R})$ not finite
- choice function f, s.t., $f(i) > i$ and $(s_{f(i)}, t_{f(i)}) \in \mathcal{P} \setminus \mathcal{P}'$

  let $[i] =$ infinite chain $\pi($ all $\pi(s_i)\sigma_i$ are connected
  - $u_i \ (\rhd \cup \to_{\mathcal{R}}) \ \pi(t_{[i]})\sigma_{[i]}$
  - $\pi(t_{[i]})\sigma_{[i]} \to_{\mathcal{R}}^* \pi(s_{[i]+1})\sigma_{[i]+1} \ \ (\rhd \cup \to_{\mathcal{R}})^* \ u_{i+1}$

The Subterm Criterion in IsaFoR

Processor

1. $\forall (s,t) \in \mathcal{P} \setminus \mathcal{P}'.\, \pi(s) \rhd \pi(t)$
2. $\forall (s,t) \in \mathcal{P}'.\, \pi(s) = \pi(t)$

then finiteness of $(\mathcal{P}', \mathcal{R})$ implies finiteness of $(\mathcal{P}, \mathcal{R})$

Proof Sketch (assuming $(\mathcal{P}, \mathcal{R})$ is not finite).

- $\exists j.\, \forall i > j.\, (s_i, t_i) \in \mathcal{P}'$: then $(\mathcal{P}', \mathcal{R})$ not finite
- choice function f, s.t., $f(i) > i$ and $(s_{f(i)}, t_{f(i)}) \in \mathcal{P} \setminus \mathcal{P}'$
  let $[i] = f^{i+1}(0)$ and $u_i = \pi(s_{[i]})\sigma_{[i]}$
  - $u_i \ (\rhd \cup \to_{\mathcal{R}})^+ \ \pi(t_{[i]})\sigma_{[i]}$
  - $\pi(t_{[i]})\sigma_{[i]} \to_{\mathcal{R}}^* \pi(s_{[i]+1})\sigma_{[i]+1} \quad (\rhd \cup \to_{\mathcal{R}})^* \ u_{i+1}$
  - $\neg\, \mathsf{SN}_{(\rhd \cup \to_{\mathcal{R}})^+}(u_i)$

The Subterm Criterion in IsaFoR

Processor

1. $\forall (s,t) \in \mathcal{P} \setminus \mathcal{P}'.\, \pi(s) \rhd \pi(t)$
2. $\forall (s,t) \in \mathcal{P}'.\, \pi(s) = \pi(t)$

then finiteness of $(\mathcal{P}', \mathcal{R})$ implies finiteness of $(\mathcal{P}, \mathcal{R})$

Proof Sketch (assuming $(\mathcal{P}, \mathcal{R})$ is not finite).

- $\exists j.\, \forall i > j.\, (s_i, t_i) \in \mathcal{P}'$: then $(\mathcal{P}', \mathcal{R})$ not finite
- choice fun… $(s_{f(i)}, t_{f(i)}) \in \mathcal{P} \setminus \mathcal{P}'$
  let $[i] = f^{i+1}(0)$ an… $= \pi(s_{[i]})\sigma_{[i]}$
  - $u_i\ (\rhd \cup \to_{\mathcal{R}})^{+}\ \pi(t_{[i]})\sigma_{[i]}$
  - $\pi(t_{[i]})\sigma_{[i]} \to_{\mathcal{R}}^{*} \pi(s_{[i]+1})\sigma_{[i]+1}\ \ (\rhd \cup \to_{\mathcal{R}})^{*}\ u_{i+1}$
  - $\neg\, \mathsf{SN}_{(\rhd \cup \to_{\mathcal{R}})^{+}}(u_i)$

$\forall i.\, u_i\ (\rhd \cup \to_{\mathcal{R}})^{+}\ u_{i+1}$

The Subterm Criterion in IsaFoR

Processor

1. $\forall (s,t) \in \mathcal{P} \setminus \mathcal{P}'.\, \pi(s) \rhd \pi(t)$
2. $\forall (s,t) \in \mathcal{P}'.\, \pi(s) = \pi(t)$

then finiteness of $(\mathcal{P}', \mathcal{R})$ implies finiteness of $(\mathcal{P}, \mathcal{R})$

Proof Sketch (assuming $(\mathcal{P}, \mathcal{R})$ is not finite).

- $\exists j.\, \forall i > j.\, (s_i, t_i) \in \mathcal{P}'$: then $(\mathcal{P}', \mathcal{R})$ not finite
- choice function f, s.t., $f(i) > i$ and $(s_{f(i)}, t_{f(i)}) \in \mathcal{P} \setminus \mathcal{P}'$
  let $[i] = f^{i+1}(0)$ and $u_i = \pi(s_{[i]})\sigma_{[i]}$
  - $u_i \; (\rhd \cup \to_{\mathcal{R}})^+ \; \pi(t_{[i]})\sigma_{[i]}$
  - $\pi(t_{[i]})\sigma_{[i]} \to_{\mathcal{R}}^* \pi(s_{[i]+1})\sigma_{[i]+1} \quad (\rhd \cup \to_{\mathcal{R}})^* \; u_{i+1}$
  - $\neg\, \mathsf{SN}_{(\rhd \cup \to_{\mathcal{R}})^+}(u_i)$
  - contradicts minimality

□

The Subterm Criterion in IsaFoR

Processor

1. $\forall (s,t) \in \mathcal{P} \setminus \mathcal{P}'.\, \pi(s)\ (\rhd \cup \to_{\mathcal{R}})^+\ \pi(t)$
2. $\forall (s,t) \in \mathcal{P}'.\, \pi(s) = \pi(t)$

then finiteness of $(\mathcal{P}', \mathcal{R})$ implies finiteness of $(\mathcal{P}, \mathcal{R})$

Proof Sketch (assuming $(\mathcal{P}, \mathcal{R})$ is not finite).

- $\exists j.\, \forall i > j.\, (s_i, t_i) \in \mathcal{P}'$: then $(\mathcal{P}', \mathcal{R})$ not finite
- choice function f, s.t., $f(i) > i$ and $(s_{f(i)}, t_{f(i)}) \in \mathcal{P} \setminus \mathcal{P}'$
  let $[i] = f^{i+1}(0)$ and $u_i = \pi(s_{[i]})\sigma_{[i]}$
  - $u_i\ (\rhd \cup \to_{\mathcal{R}})^+\ \pi(t_{[i]})\sigma_{[i]}$
  - $\pi(t_{[i]})\sigma_{[i]} \to_{\mathcal{R}}^*\ \pi(s_{[i]+1})\sigma_{[i]+1}\ \ (\rhd \cup \to_{\mathcal{R}})^*\ u_{i+1}$
  - $\neg\, \mathsf{SN}_{(\rhd \cup \to_{\mathcal{R}})^+}(u_i)$
  - contradicts minimality

□

Semantic Labeling - Algebras and Models

Definition (Algebra $\mathcal{A}$ over signature $\mathcal{F}$)

- carrier A
- $\forall (f, n) \in \mathcal{F}$ interpretation $f_{\mathcal{A}} \colon A^n \to A$

Semantic Labeling - Algebras and Models

Definition (Algebra $\mathcal{A}$ over signature $\mathcal{F}$)

- carrier A
- $\forall (f, n) \in \mathcal{F}$ interpretation $f_{\mathcal{A}} \colon A^n \to A$

Definition (Model for TRS $\mathcal{R}$)

$\mathcal{F}$-algebra $\mathcal{A}$ is model of $\mathcal{R}$ if for all $l \to r \in \mathcal{R}$ and for every assignment α

$$[\alpha]_{\mathcal{A}}(l) = [\alpha]_{\mathcal{A}}(r)$$

Semantic Labeling - Labeling for TRS $\mathcal{R}$ over Signature $\mathcal{F}$

- set of labels L_f for every $(f, n) \in \mathcal{F}$

Semantic Labeling - Labeling for TRS $\mathcal{R}$ over Signature $\mathcal{F}$

- set of labels L_f for every $(f, n) \in \mathcal{F}$
- labeling function $\ell_f : A^n \to L_f$ for every $(f, n) \in \mathcal{F}$ with $L_f \neq \varnothing$

Semantic Labeling - Labeling for TRS $\mathcal{R}$ over Signature $\mathcal{F}$

- set of labels L_f for every $(f, n) \in \mathcal{F}$
- labeling function $\ell_f \colon A^n \to L_f$ for every $(f, n) \in \mathcal{F}$ with $L_f \neq \varnothing$
- value of terms (for every assignment $\alpha \colon \mathcal{V} \to A$)

$$[\alpha]_{\mathcal{A}}(t) = \begin{cases} \alpha(t) & \text{if } t \in \mathcal{V} \\ f_{\mathcal{A}}([\alpha]_{\mathcal{A}}(t_1), \ldots, [\alpha]_{\mathcal{A}}(t_n)) & \text{if } t = f(t_1, \ldots, t_n) \end{cases}$$

Semantic Labeling - Labeling for TRS $\mathcal{R}$ over Signature $\mathcal{F}$

- set of labels L_f for every $(f, n) \in \mathcal{F}$
- labeling function $\ell_f \colon A^n \to L_f$ for every $(f, n) \in \mathcal{F}$ with $L_f \neq \varnothing$
- labeling of terms (for every assignment $\alpha \colon \mathcal{V} \to A$)

$$\mathsf{lab}_\alpha(t) = \begin{cases} t & \text{if } t \in \mathcal{V} \\ f(\mathsf{lab}_\alpha(t_1), \ldots, \mathsf{lab}_\alpha(t_n)) & \text{if } t = f(t_1, \ldots, t_n) \text{ and } L_f = \varnothing \\ f_a(\mathsf{lab}_\alpha(t_1), \ldots, \mathsf{lab}_\alpha(t_n)) & \text{if } t = f(t_1, \ldots, t_n) \text{ and } L_f \neq \varnothing \end{cases}$$

with $a = \ell_f([\alpha]_{\mathcal{A}}(t_1), \ldots, [\alpha]_{\mathcal{A}}(t_n))$

Semantic Labeling - Labeling for TRS $\mathcal{R}$ over Signature $\mathcal{F}$

- set of labels L_f for every $(f, n) \in \mathcal{F}$
- labeling function $\ell_f \colon A^n \to L_f$ for every $(f, n) \in \mathcal{F}$ with $L_f \neq \varnothing$
- labeling of terms (for every assignment $\alpha \colon \mathcal{V} \to A$)

$$\mathsf{lab}_\alpha(t) = \begin{cases} t & \text{if } t \in \mathcal{V} \\ f(\mathsf{lab}_\alpha(t_1), \ldots, \mathsf{lab}_\alpha(t_n)) & \text{if } t = f(t_1, \ldots, t_n) \text{ and } L_f = \varnothing \\ f_a(\mathsf{lab}_\alpha(t_1), \ldots, \mathsf{lab}_\alpha(t_n)) & \text{if } t = f(t_1, \ldots, t_n) \text{ and } L_f \neq \varnothing \end{cases}$$

with $a = \ell_f([\alpha]_{\mathcal{A}}(t_1), \ldots, [\alpha]_{\mathcal{A}}(t_n))$

- labeled TRS

$$\mathcal{R}_{\mathsf{lab}} = \{\, \mathsf{lab}_\alpha(l) \to \mathsf{lab}_\alpha(r) \mid l \to r \in \mathcal{R} \text{ and } \alpha \colon \mathcal{V} \to A \,\}$$

Semantic Labeling - For Termination

Theorem (Zantema 1995)

TRS $\mathcal{R}$ over signature $\mathcal{F}$ is terminating if

$\exists$ *algebra* $\mathcal{A} = (A, \{f_{\mathcal{A}}\}_{f \in \mathcal{F}})$

$\exists$ *labeling* ℓ

such that

1. $\mathcal{A}$ *is (non-empty) model of* $\mathcal{R}$
2. $\mathcal{R}_{lab}$ *is terminating*

Example - Toyama 1987

- TRS $\mathcal{R} = \{ \mathsf{f}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}(x, x, x) \}$

Example - Toyama 1987

- TRS $\mathcal{R} = \{ \mathsf{f}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}(x, x, x) \}$
- model $\mathcal{A}$: $A = \{0, 1\}$ $\mathsf{a}_{\mathcal{A}} = 0$ $\mathsf{b}_{\mathcal{A}} = 1$ $\mathsf{f}_{\mathcal{A}}(x, y, z) = 0$

Example - Toyama 1987

- TRS $\mathcal{R} = \{\, \mathsf{f}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}(x, x, x) \,\}$
- model $\mathcal{A}$: $A = \{0, 1\}$ $\mathsf{a}_{\mathcal{A}} = 0$ $\mathsf{b}_{\mathcal{A}} = 1$ $\mathsf{f}_{\mathcal{A}}(x, y, z) = 0$
- labeling

  ℓ: $L_{\mathsf{a}} = L_{\mathsf{b}} = \varnothing$ $L_{\mathsf{f}} = A$ $\ell_{\mathsf{f}}(x, y, z) = \begin{cases} 0 & \text{if } x = y \\ 1 & \text{if } x \neq y \end{cases}$

Example - Toyama 1987

- TRS $\mathcal{R} = \{\, \mathsf{f}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}(x, x, x) \,\}$
- model $\mathcal{A}$: $A = \{0, 1\}$ $\mathsf{a}_{\mathcal{A}} = 0$ $\mathsf{b}_{\mathcal{A}} = 1$ $\mathsf{f}_{\mathcal{A}}(x, y, z) = 0$
- labeling

  ℓ: $L_{\mathsf{a}} = L_{\mathsf{b}} = \varnothing$ $L_{\mathsf{f}} = A$ $\ell_{\mathsf{f}}(x, y, z) = \begin{cases} 0 & \text{if } x = y \\ 1 & \text{if } x \neq y \end{cases}$
- TRS $\mathcal{R}_{\mathsf{lab}} = \{\, \mathsf{f}_1(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}_0(x, x, x) \,\}$

Example - Toyama 1987

- TRS $\mathcal{R} = \{\, \mathsf{f}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}(x, x, x) \,\}$
- model $\mathcal{A}$: $A = \{0, 1\}$ $\mathsf{a}_{\mathcal{A}} = 0$ $\mathsf{b}_{\mathcal{A}} = 1$ $\mathsf{f}_{\mathcal{A}}(x, y, z) = 0$
- labeling

  ℓ: $L_{\mathsf{a}} = L_{\mathsf{b}} = \varnothing$ $L_{\mathsf{f}} = A$ $\ell_{\mathsf{f}}(x, y, z) = \begin{cases} 0 & \text{if } x = y \\ 1 & \text{if } x \neq y \end{cases}$
- TRS $\mathcal{R}_{\mathsf{lab}} = \{\, \mathsf{f}_1(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}_0(x, x, x) \,\}$
- no dependency pairs

Root-Labeling - Basic Version

Definition

- $\mathcal{F}$-algebra $\mathcal{A}_\mathcal{F}$
  - $A_\mathcal{F} = \mathcal{F}$
  - $f_{\mathcal{A}_\mathcal{F}}(x_1, \ldots, x_n) = f \qquad \forall (f, n) \in \mathcal{F}$

Root-Labeling - Basic Version

Definition

- $\mathcal{F}$-algebra $\mathcal{A}_\mathcal{F}$
  - $A_\mathcal{F} = \mathcal{F}$
  - $f_{\mathcal{A}_\mathcal{F}}(x_1, \ldots, x_n) = f \qquad \forall (f, n) \in \mathcal{F}$
- labeling
  - $L_f = \mathcal{F}^n$
  - $\ell_f(x_1, \ldots, x_n) = (x_1, \ldots, x_n)$

Root-Labeling - Basic Version

Definition

- $\mathcal{F}$-algebra $\mathcal{A}_\mathcal{F}$
  - $A_\mathcal{F} = \mathcal{F}$
  - $f_{\mathcal{A}_\mathcal{F}}(x_1, \ldots, x_n) = f \qquad \forall (f, n) \in \mathcal{F}$
- labeling
  - $L_f = \mathcal{F}^n$
  - $\ell_f(x_1, \ldots, x_n) = (x_1, \ldots, x_n)$
- notation $\mathcal{R}_{\mathsf{lab}} \Rightarrow \mathcal{R}_{\mathsf{rl}}$

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{ \mathsf{f}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}(x, x, x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \}$

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{ \mathsf{f}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}(x, x, x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \}$
- $\mathcal{R}_{\mathsf{rl}}$ $\quad \mathsf{f}_{(a,b,a)}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}_{(a,a,a)}(x, x, x) \quad \alpha(x) = a$

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{ \mathsf{f}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}(x, x, x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \}$
- $\mathcal{R}_{\mathsf{rl}}$
$$\begin{aligned} \mathsf{f}_{(a,b,a)}(\mathsf{a}, \mathsf{b}, x) &\to \mathsf{f}_{(a,a,a)}(x, x, x) \\ \mathsf{f}_{(a,b,b)}(\mathsf{a}, \mathsf{b}, x) &\to \mathsf{f}_{(b,b,b)}(x, x, x) \qquad \alpha(x) = b \end{aligned}$$

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{ \mathsf{f}(\mathsf{a},\mathsf{b},x) \to \mathsf{f}(x,x,x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \}$
- $\mathcal{R}_{\mathsf{rl}}$

$$\begin{aligned} \mathsf{f}_{(a,b,a)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(a,a,a)}(x,x,x) \\ \mathsf{f}_{(a,b,b)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(b,b,b)}(x,x,x) \\ \mathsf{f}_{(a,b,c)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(c,c,c)}(x,x,x) \qquad \alpha(x) = c \end{aligned}$$

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{ \mathsf{f}(\mathsf{a},\mathsf{b},x) \to \mathsf{f}(x,x,x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \}$
- $\mathcal{R}_{\mathsf{rl}}$

$$\begin{aligned}
\mathsf{f}_{(a,b,a)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(a,a,a)}(x,x,x) \\
\mathsf{f}_{(a,b,b)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(b,b,b)}(x,x,x) \\
\mathsf{f}_{(a,b,c)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(c,c,c)}(x,x,x) \\
\mathsf{f}_{(a,b,f)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(f,f,f)}(x,x,x) \qquad \alpha(x) = f
\end{aligned}$$

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{ \mathsf{f}(\mathsf{a},\mathsf{b},x) \to \mathsf{f}(x,x,x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \}$
- $\mathcal{R}_{\mathsf{rl}}$

$$
\begin{aligned}
\mathsf{f}_{(a,b,a)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(a,a,a)}(x,x,x)\\
\mathsf{f}_{(a,b,b)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(b,b,b)}(x,x,x)\\
\mathsf{f}_{(a,b,c)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(c,c,c)}(x,x,x)\\
\mathsf{f}_{(a,b,f)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(f,f,f)}(x,x,x)\\
\mathsf{c} &\to \mathsf{a}\\
\mathsf{c} &\to \mathsf{b}
\end{aligned}
$$

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{ \mathsf{f}(\mathsf{a},\mathsf{b},x) \to \mathsf{f}(x,x,x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \}$
- $\mathcal{R}_{\mathsf{rl}}$

$$
\begin{aligned}
\mathsf{f}_{(a,b,a)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(a,a,a)}(x,x,x)\\
\mathsf{f}_{(a,b,b)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(b,b,b)}(x,x,x)\\
\mathsf{f}_{(a,b,c)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(c,c,c)}(x,x,x)\\
\mathsf{f}_{(a,b,f)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(f,f,f)}(x,x,x)\\
\mathsf{c} &\to \mathsf{a}\\
\mathsf{c} &\to \mathsf{b}
\end{aligned}
$$

Problem

- $DP(\mathcal{R}_{\mathsf{rl}}) = \varnothing \quad \Longrightarrow \quad \mathcal{R}_{\mathsf{rl}}$ terminating

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{ \mathsf{f}(\mathsf{a},\mathsf{b},x) \to \mathsf{f}(x,x,x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \}$
- $\mathcal{R}_{\mathsf{rl}}$

$$
\begin{aligned}
\mathsf{f}_{(a,b,a)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(a,a,a)}(x,x,x)\\
\mathsf{f}_{(a,b,b)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(b,b,b)}(x,x,x)\\
\mathsf{f}_{(a,b,c)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(c,c,c)}(x,x,x)\\
\mathsf{f}_{(a,b,f)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(f,f,f)}(x,x,x)\\
\mathsf{c} &\to \mathsf{a}\\
\mathsf{c} &\to \mathsf{b}
\end{aligned}
$$

Problem

- $DP(\mathcal{R}_{\mathsf{rl}}) = \varnothing \implies \mathcal{R}_{\mathsf{rl}}$ terminating
- $\mathcal{R}$ not terminating

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{ \mathsf{f}(\mathsf{a},\mathsf{b},x) \to \mathsf{f}(x,x,x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \}$
- $\mathcal{R}_{\mathsf{rl}}$

$$
\begin{aligned}
\mathsf{f}_{(a,b,a)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(a,a,a)}(x,x,x)\\
\mathsf{f}_{(a,b,b)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(b,b,b)}(x,x,x)\\
\mathsf{f}_{(a,b,c)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(c,c,c)}(x,x,x)\\
\mathsf{f}_{(a,b,f)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(f,f,f)}(x,x,x)\\
\mathsf{c} &\to \mathsf{a}\\
\mathsf{c} &\to \mathsf{b}
\end{aligned}
$$

Problem

- $DP(\mathcal{R}_{\mathsf{rl}}) = \varnothing \quad \Longrightarrow \quad \mathcal{R}_{\mathsf{rl}}$ terminating
- $\mathcal{R}$ not terminating
- reason: $\mathcal{A}_{\mathcal{F}}$ not model of $\mathcal{R}$
  $([\alpha]_{\mathcal{A}_{\mathcal{F}}}(\mathsf{c}) = \mathsf{c} \neq \mathsf{a} = [\alpha]_{\mathcal{A}_{\mathcal{F}}}(\mathsf{a}))$

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{ \mathsf{f}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}(x, x, x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \}$
- $\mathcal{R}_{\mathsf{rl}}$

$$
\begin{aligned}
\mathsf{f}_{(a,b,a)}(\mathsf{a}, \mathsf{b}, x) &\to \mathsf{f}_{(a,a,a)}(x, x, x) \\
\mathsf{f}_{(a,b,b)}(\mathsf{a}, \mathsf{b}, x) &\to \mathsf{f}_{(b,b,b)}(x, x, x) \\
\mathsf{f}_{(a,b,c)}(\mathsf{a}, \mathsf{b}, x) &\to \mathsf{f}_{(c,c,c)}(x, x, x) \\
\mathsf{f}_{(a,b,f)}(\mathsf{a}, \mathsf{b}, x) &\to \mathsf{f}_{(f,f,f)}(x, x, x) \\
\mathsf{c} &\to \mathsf{a} \\
\mathsf{c} &\to \mathsf{b}
\end{aligned}
$$

Problem

- $DP(\mathcal{R}_{\mathsf{rl}}) = \varnothing \implies \mathcal{R}_{\mathsf{rl}}$ terminating
- $\mathcal{R}$ not terminating
- reason: $\mathcal{A}_{\mathcal{F}}$ not model of $\mathcal{R}$
  $([\alpha]_{\mathcal{A}_{\mathcal{F}}}(\mathsf{c}) = \mathsf{c} \neq \mathsf{a} = [\alpha]_{\mathcal{A}_{\mathcal{F}}}(\mathsf{a}))$
- solution: preprocess TRS

TRS $\mathcal{R}$ over Signature $\mathcal{F}$

Definition

- root-preserving rules
  $\mathcal{R}_{\mathsf{p}} = \{ l \to r \in \mathcal{R} \mid \mathsf{root}(l) = \mathsf{root}(r) \}$

TRS $\mathcal{R}$ over Signature $\mathcal{F}$

Definition

- root-preserving rules
  $\mathcal{R}_{\mathsf{p}} = \{ l \to r \in \mathcal{R} \mid \mathsf{root}(l) = \mathsf{root}(r) \}$
- root-altering rules
  $\mathcal{R}_{\mathsf{a}} = \mathcal{R} \setminus \mathcal{R}_{\mathsf{p}}$

TRS $\mathcal{R}$ over Signature $\mathcal{F}$

Definition

- root-preserving rules
  $\mathcal{R}_\mathsf{p} = \{ l \to r \in \mathcal{R} \mid \mathsf{root}(l) = \mathsf{root}(r) \}$
- root-altering rules
  $\mathcal{R}_\mathsf{a} = \mathcal{R} \setminus \mathcal{R}_\mathsf{p}$
- flat contexts
  $\mathcal{FC}(\mathcal{F}) = \{ f(x_1, \ldots, \Box, \ldots, x_n) \mid (f, n) \in \mathcal{F} \}$

TRS $\mathcal{R}$ over Signature $\mathcal{F}$

Definition

- root-preserving rules
  $\mathcal{R}_\mathsf{p} = \{ l \to r \in \mathcal{R} \mid \mathsf{root}(l) = \mathsf{root}(r) \}$
- root-altering rules
  $\mathcal{R}_\mathsf{a} = \mathcal{R} \setminus \mathcal{R}_\mathsf{p}$
- flat contexts
  $\mathcal{FC}(\mathcal{F}) = \{ f(x_1, \ldots, \Box, \ldots, x_n) \mid (f, n) \in \mathcal{F} \}$
- $\mathcal{FC}(\mathcal{R}) = \mathcal{R}_\mathsf{p} \cup \{ C[l] \to C[r] \mid l \to r \in \mathcal{R}_\mathsf{a} \text{ and } C \in \mathcal{FC}(\mathcal{F}) \}$

TRS $\mathcal{R}$ over Signature $\mathcal{F}$

Definition

- root-preserving rules
  $\mathcal{R}_\mathsf{p} = \{ l \to r \in \mathcal{R} \mid \mathrm{root}(l) = \mathrm{root}(r) \}$
- root-altering rules
  $\mathcal{R}_\mathsf{a} = \mathcal{R} \setminus \mathcal{R}_\mathsf{p}$
- flat contexts
  $\mathcal{FC}(\mathcal{F}) = \{ f(x_1, \ldots, \Box, \ldots, x_n) \mid (f, n) \in \mathcal{F} \}$
- $\mathcal{FC}(\mathcal{R}) = \mathcal{R}_\mathsf{p} \cup \{ C[l] \to C[r] \mid l \to r \in \mathcal{R}_\mathsf{a} \text{ and } C \in \mathcal{FC}(\mathcal{F}) \}$

Lemma

- *$\mathcal{R}$ terminating if and only if $\mathcal{FC}(\mathcal{R})$ terminating*
- *$\mathcal{A}_\mathcal{F}$ model of $\mathcal{FC}(\mathcal{R})$*

Example - Toyama 1987 extended by $c \longrightarrow a$ and $c \longrightarrow b$

- $\mathcal{R} = \{ \mathsf{f}(\mathsf{a}, \mathsf{b}, x) \longrightarrow \mathsf{f}(x, x, x), \mathsf{c} \longrightarrow \mathsf{a}, \mathsf{c} \longrightarrow \mathsf{b} \}$

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{ \mathsf{f}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}(x, x, x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \}$
- $\mathcal{FC}(\mathcal{R})$ $\quad \mathsf{f}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}(x, x, x)$

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{ \mathsf{f}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}(x, x, x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \}$
- $\mathcal{FC}(\mathcal{R})$

$$\begin{array}{lcll} \mathsf{f}(\mathsf{a}, \mathsf{b}, x) & \to & \mathsf{f}(x, x, x) & \\ \mathsf{f}(\mathsf{c}, x, y) & \to & \mathsf{f}(\mathsf{a}, x, y) & f(\Box, x, y) \end{array}$$

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{ \mathsf{f}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}(x, x, x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \}$
- $\mathcal{FC}(\mathcal{R})$

$$\begin{array}{lcll} \mathsf{f}(\mathsf{a}, \mathsf{b}, x) & \to & \mathsf{f}(x, x, x) & \\ \mathsf{f}(\mathsf{c}, x, y) & \to & \mathsf{f}(\mathsf{a}, x, y) & \\ \mathsf{f}(x, \mathsf{c}, y) & \to & \mathsf{f}(x, \mathsf{a}, y) & f(x, \Box, y) \end{array}$$

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{ \mathsf{f}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}(x, x, x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \}$
- $\mathcal{FC}(\mathcal{R})$

$$
\begin{array}{lcll}
\mathsf{f}(\mathsf{a}, \mathsf{b}, x) & \to & \mathsf{f}(x, x, x) & \\
\mathsf{f}(\mathsf{c}, x, y) & \to & \mathsf{f}(\mathsf{a}, x, y) & \\
\mathsf{f}(x, \mathsf{c}, y) & \to & \mathsf{f}(x, \mathsf{a}, y) & \\
\mathsf{f}(x, y, \mathsf{c}) & \to & \mathsf{f}(x, y, \mathsf{a}) & f(x, y, \Box)
\end{array}
$$

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{ \mathsf{f}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}(x, x, x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \}$
- $\mathcal{FC}(\mathcal{R})$

$$
\begin{array}{lcll}
\mathsf{f}(\mathsf{a}, \mathsf{b}, x) & \to & \mathsf{f}(x, x, x) & \\
\mathsf{f}(\mathsf{c}, x, y) & \to & \mathsf{f}(\mathsf{a}, x, y) & \\
\mathsf{f}(x, \mathsf{c}, y) & \to & \mathsf{f}(x, \mathsf{a}, y) & \\
\mathsf{f}(x, y, \mathsf{c}) & \to & \mathsf{f}(x, y, \mathsf{a}) & \\
\mathsf{f}(\mathsf{c}, x, y) & \to & \mathsf{f}(\mathsf{b}, x, y) & f(\Box, x, y)
\end{array}
$$

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{ \mathsf{f}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}(x, x, x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \}$
- $\mathcal{FC}(\mathcal{R})$

$$
\begin{array}{lcll}
\mathsf{f}(\mathsf{a}, \mathsf{b}, x) & \to & \mathsf{f}(x, x, x) & \\
\mathsf{f}(\mathsf{c}, x, y) & \to & \mathsf{f}(\mathsf{a}, x, y) & \\
\mathsf{f}(x, \mathsf{c}, y) & \to & \mathsf{f}(x, \mathsf{a}, y) & \\
\mathsf{f}(x, y, \mathsf{c}) & \to & \mathsf{f}(x, y, \mathsf{a}) & \\
\mathsf{f}(\mathsf{c}, x, y) & \to & \mathsf{f}(\mathsf{b}, x, y) & \\
\mathsf{f}(x, \mathsf{c}, y) & \to & \mathsf{f}(x, \mathsf{b}, y) & f(x, \Box, y)
\end{array}
$$

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{ \mathsf{f}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}(x, x, x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \}$
- $\mathcal{FC}(\mathcal{R})$

$$
\begin{array}{lcll}
\mathsf{f}(\mathsf{a}, \mathsf{b}, x) & \to & \mathsf{f}(x, x, x) & \\
\mathsf{f}(\mathsf{c}, x, y) & \to & \mathsf{f}(\mathsf{a}, x, y) & \\
\mathsf{f}(x, \mathsf{c}, y) & \to & \mathsf{f}(x, \mathsf{a}, y) & \\
\mathsf{f}(x, y, \mathsf{c}) & \to & \mathsf{f}(x, y, \mathsf{a}) & \\
\mathsf{f}(\mathsf{c}, x, y) & \to & \mathsf{f}(\mathsf{b}, x, y) & \\
\mathsf{f}(x, \mathsf{c}, y) & \to & \mathsf{f}(x, \mathsf{b}, y) & \\
\mathsf{f}(x, y, \mathsf{c}) & \to & \mathsf{f}(x, y, \mathsf{b}) & f(x, y, \Box)
\end{array}
$$

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{ \mathsf{f}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}(x, x, x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \}$
- $\mathcal{FC}(\mathcal{R})$

$$
\begin{aligned}
\mathsf{f}(\mathsf{a}, \mathsf{b}, x) &\to \mathsf{f}(x, x, x) \\
\mathsf{f}(\mathsf{c}, x, y) &\to \mathsf{f}(\mathsf{a}, x, y) \\
\mathsf{f}(x, \mathsf{c}, y) &\to \mathsf{f}(x, \mathsf{a}, y) \\
\mathsf{f}(x, y, \mathsf{c}) &\to \mathsf{f}(x, y, \mathsf{a}) \\
\mathsf{f}(\mathsf{c}, x, y) &\to \mathsf{f}(\mathsf{b}, x, y) \\
\mathsf{f}(x, \mathsf{c}, y) &\to \mathsf{f}(x, \mathsf{b}, y) \\
\mathsf{f}(x, y, \mathsf{c}) &\to \mathsf{f}(x, y, \mathsf{b})
\end{aligned}
$$

- $\mathcal{FC}(\mathcal{R})_{\mathsf{rl}}$ $\quad \mathsf{f}_{(a,b,a)}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}_{(a,a,a)}(x, x, x) \quad \alpha(x) = a$

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{ \mathsf{f}(\mathsf{a}, \mathsf{b}, x) \to \mathsf{f}(x, x, x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \}$
- $\mathcal{FC}(\mathcal{R})$

$$\begin{aligned}
\mathsf{f}(\mathsf{a}, \mathsf{b}, x) &\to \mathsf{f}(x, x, x) \\
\mathsf{f}(\mathsf{c}, x, y) &\to \mathsf{f}(\mathsf{a}, x, y) \\
\mathsf{f}(x, \mathsf{c}, y) &\to \mathsf{f}(x, \mathsf{a}, y) \\
\mathsf{f}(x, y, \mathsf{c}) &\to \mathsf{f}(x, y, \mathsf{a}) \\
\mathsf{f}(\mathsf{c}, x, y) &\to \mathsf{f}(\mathsf{b}, x, y) \\
\mathsf{f}(x, \mathsf{c}, y) &\to \mathsf{f}(x, \mathsf{b}, y) \\
\mathsf{f}(x, y, \mathsf{c}) &\to \mathsf{f}(x, y, \mathsf{b})
\end{aligned}$$

- $\mathcal{FC}(\mathcal{R})_{\mathsf{rl}}$

$$\begin{aligned}
\mathsf{f}_{(a,b,a)}(\mathsf{a}, \mathsf{b}, x) &\to \mathsf{f}_{(a,a,a)}(x, x, x) \\
\mathsf{f}_{(a,b,b)}(\mathsf{a}, \mathsf{b}, x) &\to \mathsf{f}_{(b,b,b)}(x, x, x) \qquad \alpha(x) = b
\end{aligned}$$

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{ \mathsf{f}(\mathsf{a},\mathsf{b},x) \to \mathsf{f}(x,x,x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \}$
- $\mathcal{FC}(\mathcal{R})$

$$\begin{aligned}
\mathsf{f}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}(x,x,x)\\
\mathsf{f}(\mathsf{c},x,y) &\to \mathsf{f}(\mathsf{a},x,y)\\
\mathsf{f}(x,\mathsf{c},y) &\to \mathsf{f}(x,\mathsf{a},y)\\
\mathsf{f}(x,y,\mathsf{c}) &\to \mathsf{f}(x,y,\mathsf{a})\\
\mathsf{f}(\mathsf{c},x,y) &\to \mathsf{f}(\mathsf{b},x,y)\\
\mathsf{f}(x,\mathsf{c},y) &\to \mathsf{f}(x,\mathsf{b},y)\\
\mathsf{f}(x,y,\mathsf{c}) &\to \mathsf{f}(x,y,\mathsf{b})
\end{aligned}$$

- $\mathcal{FC}(\mathcal{R})_{\mathsf{rl}}$

$$\begin{aligned}
\mathsf{f}_{(a,b,a)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(a,a,a)}(x,x,x)\\
\mathsf{f}_{(a,b,b)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(b,b,b)}(x,x,x)\\
\mathsf{f}_{(a,b,c)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(c,c,c)}(x,x,x) \qquad \alpha(x) = c
\end{aligned}$$

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{\, \mathsf{f}(\mathsf{a},\mathsf{b},x) \to \mathsf{f}(x,x,x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \,\}$
- $\mathcal{FC}(\mathcal{R})$

$$\begin{aligned}
\mathsf{f}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}(x,x,x)\\
\mathsf{f}(\mathsf{c},x,y) &\to \mathsf{f}(\mathsf{a},x,y)\\
\mathsf{f}(x,\mathsf{c},y) &\to \mathsf{f}(x,\mathsf{a},y)\\
\mathsf{f}(x,y,\mathsf{c}) &\to \mathsf{f}(x,y,\mathsf{a})\\
\mathsf{f}(\mathsf{c},x,y) &\to \mathsf{f}(\mathsf{b},x,y)\\
\mathsf{f}(x,\mathsf{c},y) &\to \mathsf{f}(x,\mathsf{b},y)\\
\mathsf{f}(x,y,\mathsf{c}) &\to \mathsf{f}(x,y,\mathsf{b})
\end{aligned}$$

- $\mathcal{FC}(\mathcal{R})_{\mathsf{rl}}$

$$\begin{aligned}
\mathsf{f}_{(a,b,a)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(a,a,a)}(x,x,x)\\
\mathsf{f}_{(a,b,b)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(b,b,b)}(x,x,x)\\
\mathsf{f}_{(a,b,c)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(c,c,c)}(x,x,x)\\
\mathsf{f}_{(a,b,f)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(f,f,f)}(x,x,x) \qquad \alpha(x) = f\\
&\vdots \qquad \text{(has 96 more rules)}
\end{aligned}$$

Example - Toyama 1987 extended by $c \to a$ and $c \to b$

- $\mathcal{R} = \{\, \mathsf{f}(\mathsf{a},\mathsf{b},x) \to \mathsf{f}(x,x,x), \mathsf{c} \to \mathsf{a}, \mathsf{c} \to \mathsf{b} \,\}$
- $\mathcal{FC}(\mathcal{R})$

$$\begin{aligned}
\mathsf{f}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}(x,x,x)\\
\mathsf{f}(\mathsf{c},x,y) &\to \mathsf{f}(\mathsf{a},x,y)\\
\mathsf{f}(x,\mathsf{c},y) &\to \mathsf{f}(x,\mathsf{a},y)\\
\mathsf{f}(x,y,\mathsf{c}) &\to \mathsf{f}(x,y,\mathsf{a})\\
\mathsf{f}(\mathsf{c},x,y) &\to \mathsf{f}(\mathsf{b},x,y)\\
\mathsf{f}(x,\mathsf{c},y) &\to \mathsf{f}(x,\mathsf{b},y)\\
\mathsf{f}(x,y,\mathsf{c}) &\to \mathsf{f}(x,y,\mathsf{b})
\end{aligned}$$

- $\mathcal{FC}(\mathcal{R})_{\mathsf{rl}}$

$$\begin{aligned}
\mathsf{f}_{(a,b,a)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(a,a,a)}(x,x,x)\\
\mathsf{f}_{(a,b,b)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(b,b,b)}(x,x,x)\\
\mathsf{f}_{(a,b,c)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(c,c,c)}(x,x,x)\\
\mathsf{f}_{(a,b,f)}(\mathsf{a},\mathsf{b},x) &\to \mathsf{f}_{(f,f,f)}(x,x,x) \qquad \alpha(x) = f\\
&\vdots \qquad \text{(has 96 more rules)}
\end{aligned}$$

- $\mathcal{FC}(\mathcal{R})_{\mathsf{rl}}$ is non-terminating

Root-Labeling - For the DP Framework

Definition (DP Problem $(\mathcal{P}, \mathcal{R})$ over Signature $\mathcal{F} \cup \mathcal{F}^\sharp$)

- $\mathcal{FC}(\mathcal{G})$ consists of all flat contexts over signature $\mathcal{G}$

Root-Labeling - For the DP Framework

Definition (DP Problem $(\mathcal{P}, \mathcal{R})$ over Signature $\mathcal{F} \cup \mathcal{F}^\sharp$)

- $\mathcal{FC}(\mathcal{G})$ consists of all flat contexts over signature $\mathcal{G}$
- $\mathcal{FC}_\mathcal{G}(\mathcal{R})$ closure of $\mathcal{R}_\mathsf{a}$ under $\mathcal{FC}(\mathcal{G})$ together with $\mathcal{R}_\mathsf{p}$

Root-Labeling - For the DP Framework

Definition (DP Problem $(\mathcal{P}, \mathcal{R})$ over Signature $\mathcal{F} \cup \mathcal{F}^\sharp$)

- $\mathcal{FC}(\mathcal{G})$ consists of all flat contexts over signature $\mathcal{G}$
- $\mathcal{FC}_\mathcal{G}(\mathcal{R})$ closure of $\mathcal{R}_\mathsf{a}$ under $\mathcal{FC}(\mathcal{G})$ together with $\mathcal{R}_\mathsf{p}$
- fresh unary function symbol $\triangle$

Root-Labeling - For the DP Framework

Definition (DP Problem $(\mathcal{P}, \mathcal{R})$ over Signature $\mathcal{F} \cup \mathcal{F}^\sharp$)

- $\mathcal{FC}(\mathcal{G})$ consists of all flat contexts over signature $\mathcal{G}$
- $\mathcal{FC}_\mathcal{G}(\mathcal{R})$ closure of $\mathcal{R}_\mathsf{a}$ under $\mathcal{FC}(\mathcal{G})$ together with $\mathcal{R}_\mathsf{p}$
- fresh unary function symbol $\triangle$
- $\mathsf{block}(f(t_1, \ldots, t_n)) = f(\triangle(t_1), \ldots, \triangle(t_n))$ ($\mathsf{block}(t) = t$ otherwise)

Root-Labeling - For the DP Framework

Definition (DP Problem $(\mathcal{P}, \mathcal{R})$ over Signature $\mathcal{F} \cup \mathcal{F}^\sharp$)

- $\mathcal{FC}(\mathcal{G})$ consists of all flat contexts over signature $\mathcal{G}$
- $\mathcal{FC}_\mathcal{G}(\mathcal{R})$ closure of $\mathcal{R}_\mathsf{a}$ under $\mathcal{FC}(\mathcal{G})$ together with $\mathcal{R}_\mathsf{p}$
- fresh unary function symbol $\triangle$
- $\mathsf{block}(f(t_1, \ldots, t_n)) = f(\triangle(t_1), \ldots, \triangle(t_n))$ ($\mathsf{block}(t) = t$ otherwise)
- $\mathcal{FC}_1(\mathcal{P}, \mathcal{R}) = (\{\mathsf{block}(l) \to \mathsf{block}(r) \mid l \to r \in \mathcal{P}\}, \mathcal{FC}_{\mathcal{F} \cup \{\triangle\}}(\mathcal{R}))$

Root-Labeling - For the DP Framework

Definition (DP Problem $(\mathcal{P}, \mathcal{R})$ over Signature $\mathcal{F} \cup \mathcal{F}^\sharp$)

- $\mathcal{FC}(\mathcal{G})$ consists of all flat contexts over signature $\mathcal{G}$
- $\mathcal{FC}_\mathcal{G}(\mathcal{R})$ closure of $\mathcal{R}_\mathsf{a}$ under $\mathcal{FC}(\mathcal{G})$ together with $\mathcal{R}_\mathsf{p}$
- fresh unary function symbol $\triangle$
- $\mathsf{block}(f(t_1, \ldots, t_n)) = f(\triangle(t_1), \ldots, \triangle(t_n))$ ($\mathsf{block}(t) = t$ otherwise)
- $\mathcal{FC}_1(\mathcal{P}, \mathcal{R}) = (\{\mathsf{block}(l) \to \mathsf{block}(r) \mid l \to r \in \mathcal{P}\}, \mathcal{FC}_{\mathcal{F} \cup \{\triangle\}}(\mathcal{R}))$

Theorem (Soundness)

$(\mathcal{P}, \mathcal{R})$ *finite* $\quad \Longleftarrow \quad$ $\mathcal{FC}_1(\mathcal{P}, \mathcal{R})$ *finite*

Root-Labeling - For the DP Framework

Definition (DP Problem $(\mathcal{P}, \mathcal{R})$ over Signature $\mathcal{F} \cup \mathcal{F}^\sharp$)

- $\mathcal{FC}(\mathcal{G})$ consists of all flat contexts over signature $\mathcal{G}$
- $\mathcal{FC}_\mathcal{G}(\mathcal{R})$ closure of $\mathcal{R}_\mathsf{a}$ under $\mathcal{FC}(\mathcal{G})$ together with $\mathcal{R}_\mathsf{p}$
- fresh unary function symbol $\triangle$
- $\mathsf{block}(f(t_1, \ldots, t_n)) = f(\triangle(t_1), \ldots, \triangle(t_n))$ ($\mathsf{block}(t) = t$ otherwise)
- $\mathcal{FC}_1(\mathcal{P}, \mathcal{R}) = (\{\mathsf{block}(l) \to \mathsf{block}(r) \mid l \to r \in \mathcal{P}\}, \mathcal{FC}_{\mathcal{F} \cup \{\triangle\}}(\mathcal{R}))$

Theorem (Soundness)

$(\mathcal{P}, \mathcal{R})$ *finite* $\impliedby$ $\mathcal{FC}_1(\mathcal{P}, \mathcal{R})$ *finite and* $\mathcal{R}$ *is left-linear*

Hélène Touzet 1998

Definition (SRS $\mathcal{T}$)

$$\mathsf{bu} \longrightarrow \mathsf{bs} \qquad \mathsf{sbs} \longrightarrow \mathsf{bt} \qquad \mathsf{tb} \longrightarrow \mathsf{bs} \qquad \mathsf{ts} \longrightarrow \mathsf{tt}$$

$$\mathsf{sb} \longrightarrow \mathsf{bsss} \qquad \mathsf{su} \longrightarrow \mathsf{ss} \qquad \mathsf{tbs} \longrightarrow \mathsf{utb} \qquad \mathsf{tu} \longrightarrow \mathsf{ut}$$

Hélène Touzet 1998

Definition (SRS $\mathcal{T}$)

$$\mathsf{bu} \to \mathsf{bs} \qquad \mathsf{sbs} \to \mathsf{bt} \qquad \mathsf{tb} \to \mathsf{bs} \qquad \mathsf{ts} \to \mathsf{tt}$$
$$\mathsf{sb} \to \mathsf{bsss} \qquad \mathsf{su} \to \mathsf{ss} \qquad \mathsf{tbs} \to \mathsf{utb} \qquad \mathsf{tu} \to \mathsf{ut}$$

Theorem

- *derivational complexity of* $\mathcal{T}$ *is not primitive recursive*
- $\mathcal{T}$ *is simply terminating*

Hélène Touzet 1998

Definition (SRS $\mathcal{T}$)

$$\begin{array}{llll} bu \to bs & sbs \to bt & tb \to bs & ts \to tt \\ sb \to bsss & su \to ss & tbs \to utb & tu \to ut \end{array}$$

Theorem

- *derivational complexity of* $\mathcal{T}$ *is not primitive recursive*
- $\mathcal{T}$ *is simply terminating*

Remark

- no automatic termination proof before termination competition 2007

Hélène Touzet 1998

Definition (SRS $\mathcal{T}$)

$$\begin{array}{llll} bu \to bs & sbs \to bt & tb \to bs & ts \to tt \\ sb \to bsss & su \to ss & tbs \to utb & tu \to ut \end{array}$$

Theorem

- *derivational complexity of $\mathcal{T}$ is not primitive recursive*
- *$\mathcal{T}$ is simply terminating*

Remark

- no automatic termination proof before termination competition 2007
- with root-labeling $\mathsf{T}_\mathsf{T}\mathsf{T}_2$ proves termination of $\mathcal{T}$

Hélène Touzet 1998

Definition (SRS $\mathcal{T}$)

$$\begin{array}{llll} \mathsf{bu} \to \mathsf{bs} & \mathsf{sbs} \to \mathsf{bt} & \mathsf{tb} \to \mathsf{bs} & \mathsf{ts} \to \mathsf{tt} \\ \mathsf{sb} \to \mathsf{bsss} & \mathsf{su} \to \mathsf{ss} & \mathsf{tbs} \to \mathsf{utb} & \mathsf{tu} \to \mathsf{ut} \end{array}$$

Theorem

- *derivational complexity of $\mathcal{T}$ is not primitive recursive*
- *$\mathcal{T}$ is simply terminating*

Remark

- no automatic termination proof before termination competition 2007
- with root-labeling $\mathsf{T_T T_2}$ proves termination of $\mathcal{T}$
- CeTA successfully certifies this proof

Overview

- Motivation
- Preliminaries
  - Isabelle/HOL
  - Code-Extraction
  - IsaFoR
- Termination Techniques
  - Dependency Pairs
  - Subterm Criterion
  - Root-Labeling
- CeTA

The Certifier CeTA

demo

Some Statistics

Formalization

	PhD-Thesis	IsaFoR
lines	15.900	44.700
lemmas	888	1.859
definitions	107	383
functions	48	272

Some Statistics

Formalization

	PhD-Thesis	IsaFoR
lines	15.900	44.700
lemmas	888	1.859
definitions	107	383
functions	48	272

Certification

	noncert	CeTA
AProVE	365	266
$T_T T_2$	274	262

Some Statistics

Formalization

	PhD-Thesis	IsaFoR
lines	15.900	44.700
lemmas	888	1.859
definitions	107	383
functions		

more than 80 %!

Certification

	noncert	CeTA
AProVE	365	266
T_TT_2	274	262

Conclusion

Summary

Conclusion

Summary

- Isabelle library on rewriting (IsaFoR)

Conclusion

Summary

- Isabelle library on rewriting (IsaFoR)
- formalized termination techniques: DP transformation, subterm criterion, root-labeling (closure under flat contexts)

Conclusion

Summary

- Isabelle library on rewriting (IsaFoR)
- formalized termination techniques: DP transformation, subterm criterion, root-labeling (closure under flat contexts)
- first mechanized proof of termination preservation for signature extensions (+ additional results for finiteness of DP problems)

Conclusion

Summary

- Isabelle library on rewriting (IsaFoR)
- formalized termination techniques: DP transformation, subterm criterion, root-labeling (closure under flat contexts)
- first mechanized proof of termination preservation for signature extensions (+ additional results for finiteness of DP problems)

Future Work

Conclusion

Summary

- Isabelle library on rewriting (IsaFoR)
- formalized termination techniques: DP transformation, subterm criterion, root-labeling (closure under flat contexts)
- first mechanized proof of termination preservation for signature extensions (+ additional results for finiteness of DP problems)

Future Work

- formalize more techniques

Conclusion

Summary

- Isabelle library on rewriting (IsaFoR)
- formalized termination techniques: DP transformation, subterm criterion, root-labeling (closure under flat contexts)
- first mechanized proof of termination preservation for signature extensions (+ additional results for finiteness of DP problems)

Future Work

- formalize more techniques
- allow termination tools to prove termination of Isabelle/HOL functions