



# Term Rewriting

Philipp Dablander and Aart Middeldorp

## Definitions

- ▶ **one-step (many-step) rewrite strategy**  $\mathcal{S}$  is relation  $\xrightarrow{\mathcal{S}} \subseteq \xrightarrow{\mathcal{R}}^{(+)}$  such that  $\text{NF}(\mathcal{S}) = \text{NF}(\mathcal{R})$
- ▶ rewrite strategy  $\mathcal{S}$  is **deterministic** if  $\xleftarrow{\mathcal{S}} \cdot \xrightarrow{\mathcal{S}} \subseteq =$
- ▶ rewrite strategy  $\mathcal{S}$  **normalizes** term  $t$  if all  $\mathcal{S}$ -rewrite sequences starting from  $t$  are finite
- ▶ rewrite strategy  $\mathcal{S}$  is **normalizing** if it normalizes every normalizing term
- ▶ **relative rewriting**:  $\xrightarrow{\mathcal{S}/\mathcal{R}} = \xrightarrow{\mathcal{R}}^* \cdot \xrightarrow{\mathcal{S}} \cdot \xrightarrow{\mathcal{R}}^*$
- ▶ rewrite strategy  $\mathcal{S}$  is **hyper-normalizing** if every  $\mathcal{R}$ -normalizing term is  $\mathcal{S}/\mathcal{R}$ -terminating
- ▶ rewrite strategy  $\mathcal{S}$  is **perpetual** if every maximal  $\mathcal{S}$ -rewrite sequence starting from any non-terminating term is infinite
- ▶ rewrite strategy  $\mathcal{S}$  is **cofinal** for TRS  $\mathcal{R}$  if for every  $s \xrightarrow{\mathcal{R}}^* t$  and every maximal sequence  $s = s_0 \xrightarrow{\mathcal{S}} s_1 \xrightarrow{\mathcal{S}} s_2 \xrightarrow{\mathcal{S}} \dots$  there exists  $k \geq 0$  such that  $t \xrightarrow{\mathcal{R}}^* s_k$

## Outline

1. Summary of Lecture 10
2. Optimal Strategies
3. Strategy Annotations
4. Simple Termination
5. Exercises
6. Further Reading

## Lemma

cofinal strategies are normalizing

## Definitions

- ▶ **leftmost outermost** strategy contracts leftmost of outermost redexes
- ▶ **leftmost innermost** strategy contracts leftmost of innermost redexes
- ▶ **maximal outermost** strategy contracts all outermost redexes
- ▶ **maximal innermost** strategy contracts all innermost redexes
- ▶ **maximal** strategy performs maximal multi-step

## Definition

TRS is **left-normal** if variables do not precede function symbols in left-hand sides

### Theorem

- ▶ maximal strategy is cofinal for orthogonal TRSs
- ▶ maximal outermost strategy is hyper-normalizing for orthogonal TRSs
- ▶ leftmost outermost strategy is normalizing for orthogonal left-normal TRSs

### Notation

$a \xrightarrow{\langle l, r \rangle}^* b$  if conversion  $a \leftrightarrow^* b$  has  $l$  left ( $\leftarrow$ ) steps and  $r$  right ( $\rightarrow$ ) steps

### Definition

ARS  $\mathcal{A} = \langle A, \rightarrow \rangle$  has **random descent** if  $a \rightarrow^{r-l} b$  whenever  $a \xrightarrow{\langle l, r \rangle}^* b \in \text{NF}(\mathcal{A})$

### Theorem

if ARS  $\mathcal{A}$  has random descent and  $a \leftrightarrow^* b$  with normal form  $b$  then  $a$  is complete and all rewrite sequences from  $a$  to  $b$  have same length

## Outline

1. Summary of Lecture 10
2. Optimal Strategies
3. Strategy Annotations
4. Simple Termination
5. Exercises
6. Further Reading

### Theorem

- ▶ innermost rewriting has random descent for orthogonal TRSs
- ▶ any innermost strategy is perpetual for orthogonal TRSs

### Observation

maximal outermost and maximal strategies are not **optimal**

### Example

- ▶ TRS  $\mathcal{R}$ 

$$\begin{array}{ll} 0 + y \rightarrow y & 0 \times y \rightarrow 0 \\ s(x) + y \rightarrow s(x + y) & s(x) \times y \rightarrow (x \times y) + y \end{array}$$
- ▶ rewrite sequence  $(0 \times s(0)) \times (0 + s(0)) \mapsto 0 \times s(0) \rightarrow 0$
- ▶ redex  $0 + s(0)$  is not **needed**:  $(0 \times s(0)) \times \bullet \rightarrow 0 \times \bullet \rightarrow 0$

### Definition (Needed Redex)

redex occurrence  $\Delta$  in term  $C[\Delta]$  with respect to TRS  $\mathcal{R}$  is **needed** if  $C[\bullet]$  has no normal form in  $\mathcal{R} \cup \{\bullet \rightarrow \bullet\}$

## Theorem

for orthogonal TRSs

- ▶ every reducible term has needed redex
- ▶ needed reduction is hyper-normalizing

## Remark

for orthogonal TRSs it is **undecidable** whether redex is needed

## Remark

decidable approximations based on tree automata techniques exist

## Theorem

for **left-normal** orthogonal TRSs **leftmost outermost** redex is needed

## Example

▶ rewrite rules

$$\text{and}(x, T) \xrightarrow{\alpha} x \quad \text{and}(x, F) \xrightarrow{\beta} F \quad \text{or}(T, x) \xrightarrow{\gamma} T \quad \text{or}(F, x) \xrightarrow{\delta} x \quad \infty \xrightarrow{\epsilon} \infty$$

▶ strategy annotation

$$A(\text{and}) = [2, \alpha, \beta, 1] \quad A(\text{or}) = [1, \gamma, \delta, 2] \quad A(\infty) = [\epsilon]$$

▶ evaluation

$$\text{or}(\text{and}(\infty, F), \text{or}(T, \infty)) \rightarrow \text{or}(F, \text{or}(T, \infty)) \rightarrow \text{or}(T, \infty) \rightarrow T$$

## Outline

1. Summary of Lecture 10
2. Optimal Strategies
3. Strategy Annotations
4. Simple Termination
5. Exercises
6. Further Reading

## Definitions (Strategy Annotation)

▶ **strategy annotation** for function symbol  $f$  is finite list  $A(f)$  containing

- ▶ argument positions of  $f$
- ▶ (labels of) rewrite rules for  $f$

▶ strategy annotation  $A(f)$  for function symbol  $f$  is **full** if  $A(f)$  contains all argument positions of  $f$  and all rewrite rules for  $f$

▶ strategy annotation  $A(f)$  for function symbol  $f$  is **in-time** if argument positions are listed in  $A(f)$  before rewrite rules that **need** them

▶ rewrite rule  $f(s_1, \dots, s_n) \rightarrow t$  **needs** argument position  $i$  if

- ▶  $s_i$  is non-variable, or
- ▶  $s_i$  is variable that appears in  $s_1, \dots, s_{i-1}, s_{i+1}, \dots, s_n$

## Example

### ► rewrite rules

$$\text{if}(T, x, y) \xrightarrow{\alpha} x \quad \text{if}(F, x, y) \xrightarrow{\beta} y \quad \text{if}(z, x, x) \xrightarrow{\gamma} x$$

### ► strategy annotations

	full	in-time
$A_1(\text{if}) = [1, \alpha, \beta, 2, 3, \gamma]$	😊	😊
$A_2(\text{if}) = [\alpha, \beta, 1, 2, 3, \gamma]$	😊	😞
$A_3(\text{if}) = [1, \alpha, \beta, \gamma, 2, 3]$	😊	😞
$A_4(\text{if}) = [1, \alpha, \beta]$	😞	😊

## Lemma

if argument position  $i$  is not needed for  $\ell$  and  $t \geq \ell$  then  $t[u]_i \geq \ell$  for any term  $u$

## Definition

$$\begin{aligned} \text{redex}'_A(t, []) &= \perp & \text{redex}_A(t) &= \text{redex}'_A(t, A(\text{root}(t))) \\ \text{redex}'_A(t, [\ell \rightarrow r \mid L]) &= \begin{cases} (\epsilon, \ell \rightarrow r) & \text{if } t \geq \ell \\ \text{redex}'_A(t, L) & \text{otherwise} \end{cases} \\ \text{redex}'_A(t, [i \mid L]) &= \begin{cases} (ip, \ell \rightarrow r) & \text{if } \text{redex}_A(t|_i) = (p, \ell \rightarrow r) \\ \text{redex}'_A(t, L) & \text{otherwise} \end{cases} \end{aligned}$$

## Theorem

$\forall$  full strategy annotation  $A \quad \forall$  term  $t \quad \text{redex}_A(t) = \perp \iff t$  is normal form

## Example

### ► rewrite rules

$$\text{and}(x, T) \xrightarrow{\alpha} x \quad \text{and}(x, F) \xrightarrow{\beta} F \quad \text{or}(T, x) \xrightarrow{\gamma} T \quad \text{or}(F, x) \xrightarrow{\delta} x \quad \infty \xrightarrow{\epsilon} \infty$$

### ► strategy annotation

$$A(\text{and}) = [2, \alpha, \beta, 1] \quad A(\text{or}) = [1, \gamma, \delta, 2] \quad A(\infty) = [\epsilon]$$

### ► evaluation

$$\begin{aligned} &\text{redex}_A(\text{or}(\text{and}(\infty, F), \text{or}(T, \infty))) \\ &= \text{redex}'_A(\text{or}(\text{and}(\infty, F), \text{or}(T, \infty)), [1, \gamma, \delta, 2]) = (1, \beta) \\ &\text{redex}_A(\text{and}(\infty, F)) \\ &= \text{redex}'_A(\text{and}(\infty, F), [2, \alpha, \beta, 1]) \\ &= \text{redex}'_A(\text{and}(\infty, F), [\alpha, \beta, 1]) \\ &= \text{redex}'_A(\text{and}(\infty, F), [\beta, 1]) = (\epsilon, \beta) \\ &\text{redex}_A(F) = \perp \end{aligned}$$

## Example

### ► rewrite rules

$$\text{if}(T, x, y) \xrightarrow{\alpha} x \quad \text{if}(F, x, y) \xrightarrow{\beta} y \quad \text{if}(z, x, x) \xrightarrow{\gamma} x$$

### ► strategy annotation $A$ with $A(\text{if}) = [1, \alpha, 2, 3, \gamma]$ is not full

$$\text{redex}_A(\text{if}(F, T, F)) = \perp \quad \text{if}(F, T, F) \rightarrow F$$

## Definition

$s \xrightarrow{S_A} t$  if  $\text{redex}_A(s) = (p, \ell \rightarrow r)$  and  $s \rightarrow_{p|\ell \rightarrow r} t$

## Corollary

$S_A$  is rewrite strategy for every full strategy annotation  $A$

## Definition

$$\text{normalize}_A(t) = \text{normalize}'_A(t, A(\text{root}(t)))$$

$$\text{normalize}'_A(t, []) = t$$

$$\text{normalize}'_A(t, [\ell \rightarrow r \mid L]) = \begin{cases} \text{normalize}_A(r\sigma) & \text{if } t = \ell\sigma \text{ for some substitution } \sigma \\ \text{normalize}'_A(t, L) & \text{otherwise} \end{cases}$$

$$\text{normalize}'_A(t, [i \mid L]) = \text{normalize}'_A(t[\text{normalize}_A(t|_i)]_i, L)$$

## Remark

normalize and normalize' are evaluated in call-by-value manner

## Theorem

$\forall$  full and in-time strategy annotation  $A$

1  $S_A$  normalizes term  $t \iff \text{normalize}_A(t)$  is defined

2  $t \downarrow_{S_A} = \text{normalize}_A(t)$  for all normalizing terms  $t$

## Theorem

$\forall$  full and in-time strategy annotation  $A \ \forall$  term  $t$

leftmost innermost strategy normalizes  $t \implies S_A$  normalizes  $t$

## Example

► rewrite rules

$$\text{and}(x, T) \xrightarrow{\alpha} x \quad \text{and}(x, F) \xrightarrow{\beta} F \quad \text{or}(T, x) \xrightarrow{\gamma} T \quad \text{or}(F, x) \xrightarrow{\delta} x \quad \infty \xrightarrow{\epsilon} \infty$$

► strategy annotation

$$A(\text{and}) = [2, \alpha, \beta, 1]$$

$$A(\text{or}) = [1, \gamma, \delta, 2]$$

$$A(\infty) = [\epsilon]$$

► evaluation

$$\text{normalize}_A(\text{or}(\text{and}(\infty, F), \text{or}(T, \infty)))$$

$$= \text{normalize}'_A(\text{or}(\text{and}(\infty, F), \text{or}(T, \infty)), [1, \gamma, \delta, 2])$$

$$= \text{normalize}'_A(\text{or}(\text{normalize}_A(\text{and}(\infty, F)), \text{or}(T, \infty)), [\gamma, \delta, 2])$$

$$= \dots = \text{normalize}'_A(\text{or}(F, \text{or}(T, \infty)), [\gamma, \delta, 2])$$

$$= \text{normalize}'_A(\text{or}(F, \text{or}(T, \infty)), [\delta, 2])$$

$$= \text{normalize}_A(\text{or}(T, \infty)) = \text{normalize}'_A(\text{or}(T, \infty), [1, \gamma, \delta, 2])$$

$$= \text{normalize}'_A(\text{or}(\text{normalize}_A(T), \infty), [\gamma, \delta, 2])$$

$$= \text{normalize}'_A(\text{or}(T, \infty), [\gamma, \delta, 2]) = \text{normalize}_A(T) = T$$

## Outline

1. Summary of Lecture 10

2. Optimal Strategies

3. Strategy Annotations

4. Simple Termination

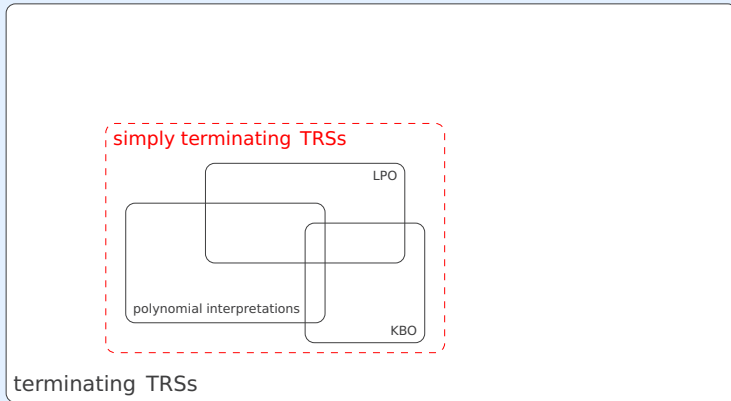
Embedding

Kruskal's Tree Theorem

Simple Monotone Algebras

5. Exercises

6. Further Reading



## Outline

1. Summary of Lecture 10
2. Optimal Strategies
3. Strategy Annotations
4. **Simple Termination**
  - Embedding      Kruskal's Tree Theorem      Simple Monotone Algebras
5. Exercises
6. Further Reading

### Definitions (Subterm Property)

- ▶ term relation  $>$  has **subterm property** if  $C[t] > t$  for all non-empty contexts  $C$  and terms  $t$
- ▶ TRS  $\mathcal{E}_{mb} = \{f(x_1, \dots, x_n) \rightarrow x_i \mid f \text{ is } n\text{-ary function symbol, } 1 \leq i \leq n\}$
- ▶  $\triangleleft_{emb} = \leftarrow_{\mathcal{E}_{mb}}^*$  (**embedding**)
- ▶  $\triangleright_{emb} = \rightarrow_{\mathcal{E}_{mb}}^+$

### Example

term  $f(g(a), b)$  is (properly) embedded in  $f(h(g(a), a), g(b))$  since

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

in TRS

$$\mathcal{E}_{mb}(\{a, b, f, g, h\}) = \left\{ \begin{array}{lll} f(x, y) \rightarrow x & g(x) \rightarrow x & h(x, y) \rightarrow x \\ f(x, y) \rightarrow y & & h(x, y) \rightarrow y \end{array} \right\}$$

**Lemma**

$\triangleright_{emb}$  is smallest rewrite order with subterm property

**Definitions (Simple Termination)**

- ▶ reduction order is well-founded rewrite order
- ▶ **simplification order** is rewrite order with subterm property
- ▶ TRS is **simply terminating** if it is compatible with simplification order

**Theorem**

- ▶  $>_{lpo}$  is simplification order
- ▶  $>_{kbo}$  is simplification order

assumption: signatures are **finite**

# Outline

1. Summary of Lecture 10
2. Optimal Strategies
3. Strategy Annotations
- 4. Simple Termination**
  - Embedding
  - Kruskal's Tree Theorem
  - Simple Monotone Algebras
5. Exercises
6. Further Reading

## Kruskal's Tree Theorem

$\forall$  infinite sequence of ground terms  $t_1, t_2, t_3, \dots \exists i < j$  such that  $t_i \preceq_{\text{emb}} t_j$

## Lemma

simplification orders are **well-founded**

## Proof (by contradiction)

- ▶ consider simplification order  $> \implies \triangleright_{\text{emb}} \subseteq >$
- ▶  $\text{Var}(t) \subseteq \text{Var}(s)$  whenever  $s > t$
- ▶ consider infinite sequence  $t_1 > t_2 > t_3 > \dots$
- ▶ define substitution  $\tau = \{x \mapsto c \mid x \in \text{Var}(t_1)\}$  where  $c$  is (fresh) constant
- ▶ infinite sequence  $t_1\tau > t_2\tau > t_3\tau > \dots$  of ground terms
- ▶  $\exists i < j$  such that  $t_i\tau \preceq_{\text{emb}} t_j\tau \implies t_i\tau \leq t_j\tau \quad \text{⚡}$

## Corollary

simply terminating TRSs are **terminating**

## Lemma

TRS  $\mathcal{R}$  is simply terminating if and only if  $\mathcal{R} \cup \mathcal{E}_{\text{mb}}$  is terminating

## Examples

- ▶ SRS  $aa \rightarrow aba$  is not simply terminating:

$$aa \rightarrow aba \rightarrow aa$$

- ▶ TRS  $f(a, b, x) \rightarrow f(x, x, x)$  is not simply terminating:

$$\begin{aligned} f(a, b, f(a, b, b)) &\rightarrow f(f(a, b, b), f(a, b, b), f(a, b, b)) \\ &\rightarrow f(a, f(a, b, b), f(a, b, b)) \\ &\rightarrow f(a, b, f(a, b, b)) \end{aligned}$$

# Outline

1. Summary of Lecture 10
2. Optimal Strategies
3. Strategy Annotations
- 4. Simple Termination**
  - Embedding
  - Kruskal's Tree Theorem
  - Simple Monotone Algebras
5. Exercises
6. Further Reading

### Definition (Simple Monotone Algebra)

**simple monotone**  $\mathcal{F}$ -algebra  $(\mathcal{A}, >)$  consists of non-empty algebra  $\mathcal{A} = (A, \{f_{\mathcal{A}}\}_{f \in \mathcal{F}})$  with well-founded order  $>$  on  $A$  such that every  $f_{\mathcal{A}}$  is simple and weakly monotone:

$$f_{\mathcal{A}}(a_1, \dots, a_i, \dots, a_n) \geq a_i$$

for all  $a_1, \dots, a_n \in A$  and  $i \in \{1, \dots, n\}$

$$f_{\mathcal{A}}(a_1, \dots, a_i, \dots, a_n) \geq f_{\mathcal{A}}(a_1, \dots, b, \dots, a_n)$$

for all  $a_1, \dots, a_n, b \in A$  and  $i \in \{1, \dots, n\}$  with  $a_i > b$

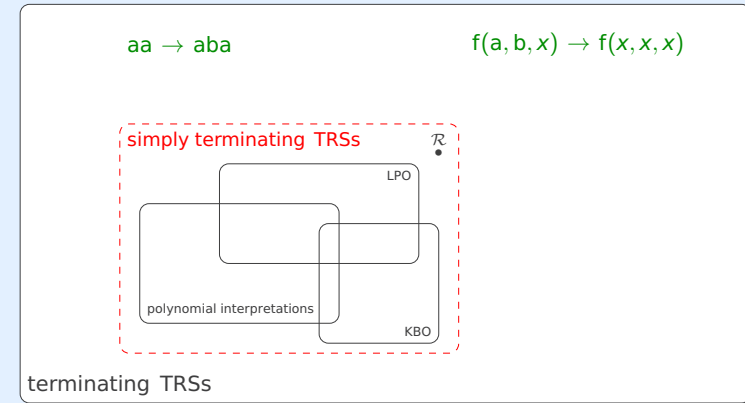
### Theorem

TRS  $\mathcal{R}$  is simply terminating  $\iff \mathcal{R}$  is compatible with simple monotone algebra

### Corollary

polynomially terminating TRSs are simply terminating

### Termination Methods: Realistic View



$\mathcal{R}$ :    f(a)  $\rightarrow$  f(b)    g(b)  $\rightarrow$  g(a)

## Outline

1. Summary of Lecture 10
2. Optimal Strategies
3. Strategy Annotations
4. Simple Termination
5. Exercises
6. Further Reading

### Homework Exercises for June 8

- ① Exercise 7.12. 1
- ② Exercise 7.17. 2
- ③ Exercise 7.19. 2
- ④ Exercise 4.24. 2

next lecture (June 8): online evaluation in presence  $\implies$  bring device

# Outline

1. Summary of Lecture 10
2. Optimal Strategies
3. Strategy Annotations
4. Simple Termination
5. Exercises
- 6. Further Reading**

## Lecture Notes

- ▶ Section 7.3
- ▶ Section 7.4
- ▶ Section 4.2
- ▶ Appendix B.1
- ▶ Appendix B.2

## Important Concepts

- ▶ embedding
- ▶ full strategy annotation
- ▶ in-time strategy annotation
- ▶ Kruskal's Tree Theorem
- ▶ needed
- ▶ simple monotone algebra
- ▶ simple termination
- ▶ simplification order
- ▶ subterm property
- ▶ strategy annotation