ISR 2010

Introduction to Term Rewriting

lecture 6

Aart Middeldorp and Femke van Raamsdonk

Institute of Computer Science
University of Innsbruck

Department of Computer Science
VU Amsterdam

Sunday

introduction, examples, abstract rewriting, equational reasoning, term rewriting

Monday

termination, completion

Tuesday

completion, termination

Wednesday

confluence, modularity, strategies

Thursday

exam, advanced topics

Outline

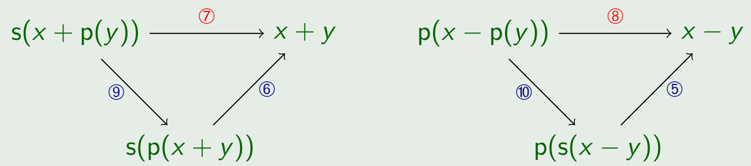
- Efficient Completion
- Cola Gene Puzzle
- Abstract Completion
- Proof Orders
- Critical Pair Criteria
- Further Reading

Example

TRS $\mathcal{R} = \{①, ②, ③, ④, ⑤, ⑥\}$ TRS $\mathcal{S} = \{①, ②, ③, ④, ⑤, ⑥, ⑦, ⑧, ⑨, ⑩\}$

①	$x + 0 \to x$	②	$x - 0 \to x$
③	$x + \mathsf{s}(y) \to \mathsf{s}(x + y)$	④	$x - \mathsf{s}(y) \to \mathsf{p}(x - y)$
⑤	$\mathsf{p}(\mathsf{s}(x)) \to x$	⑥	$\mathsf{s}(\mathsf{p}(x)) \to x$
⑦	$\mathsf{s}(x + \mathsf{p}(y)) \to x + y$	⑧	$\mathsf{p}(x - \mathsf{p}(y)) \to x - y$
⑨	$x + \mathsf{p}(y) \to \mathsf{p}(x + y)$	⑩	$x - \mathsf{p}(y) \to \mathsf{s}(x - y)$

rewrite rules ⑦ and ⑧ are redundant:



Observation

- less rewrite rules $\Longrightarrow$ less critical pairs
- TRS without redundancy $=$ reduced TRS

Definition

TRS $\mathcal{R}$ is reduced if for all $\ell \to r \in \mathcal{R}$

1. r is normal form with respect to $\mathcal{R}$
2. ℓ is normal form with respect to $\mathcal{R} \setminus \{\ell \to r\}$

Example

TRS $\mathcal{R} = \{①, ②, ③, ④, ⑤, ⑥\}$ TRS $\mathcal{S} = \{①, ②, ③, ④, ⑤, ⑥, ⑦, ⑧, ⑨, ⑩\}$

①	$x + 0 \to x$	②	$x - 0 \to x$
③	$x + \mathsf{s}(y) \to \mathsf{s}(x + y)$	④	$x - \mathsf{s}(y) \to \mathsf{p}(x - y)$
⑤	$\mathsf{p}(\mathsf{s}(x)) \to x$	⑥	$\mathsf{s}(\mathsf{p}(x)) \to x$
⑦	$\mathsf{s}(x + \mathsf{p}(y)) \to x + y$	⑧	$\mathsf{p}(x - \mathsf{p}(y)) \to x - y$
⑨	$x + \mathsf{p}(y) \to \mathsf{p}(x + y)$	⑩	$x - \mathsf{p}(y) \to \mathsf{s}(x - y)$

- $\mathcal{R}$ is reduced
- $\mathcal{S}$ is not reduced

simplification after completion

Theorem

$\forall$ complete TRS $\mathcal{R}$ $\exists$ complete reduced TRS $\mathcal{S}$ such that $\xleftrightarrow[\mathcal{R}]{*} \;=\; \xleftrightarrow[\mathcal{S}]{*}$

Proof Sketch (construction)

1. $\mathcal{R}' = \{\, \ell \to r{\downarrow_{\mathcal{R}}} \mid \ell \to r \in \mathcal{R} \,\}$
2. $\mathcal{S} = \{\, \ell \to r \in \mathcal{R}' \mid \ell \in \mathsf{NF}(\,\mathcal{R}' \setminus \{\ell \to r\}\,) \,\}$

more efficient: simplification during completion

Knuth-Bendix Completion Procedure (More Efficient Version)

input ES $\mathcal{E}$ and reduction order $>$
output complete reduced TRS $\mathcal{R}$ such that $\xleftrightarrow[\mathcal{E}]{*} \;=\; \xleftrightarrow[\mathcal{R}]{*}$

```
R := ∅    C := E
while C ≠ ∅ do
    choose s ≈ t ∈ C    C := C \ {s ≈ t}    s' := s↓R    t' := t↓R
    if s' ≠ t' then
        if s' > t' then         α := s'    β := t'
        else if t' > s' then    α := t'    β := s'
        else                    failure
        R' := R ∪ {α → β}
        for all ℓ → r ∈ R do
            R' := R' \ {ℓ → r}    ℓ' := ℓ↓R'    r' := r↓R'
            if ℓ = ℓ' then R' := R' ∪ {ℓ' → r'} else C := C ∪ {ℓ' ≈ r'}
        R := R'
        C := C ∪ {e ∈ CP(R) | α → β was used to generate e}
```

Example

$$\begin{aligned} g(b) &\approx g(b) \\ f(b) &\approx g(f(a)) \end{aligned} \qquad \begin{aligned} f(f(x)) &\to g(x) \\ g(a) &\to b \\ f(g(x)) &\to g(f(x)) \\ f(b) &\to g(f(a)) \end{aligned}$$

- LPO with precedence $f > g > b > a$
- complete and reduced TRS

Example

$$\begin{aligned} f(f(x)) &\approx g(x) \\ g(a) &\approx b \end{aligned} \qquad \begin{aligned} g(x) &\to f(f(x)) \\ b &\to f(f(a)) \end{aligned}$$

- LPO with precedence $b > g > f > a$
- complete and reduced TRS

Example

$$\begin{aligned} f(f(a)) &\approx b \\ g(a) &\approx b \end{aligned} \qquad \begin{aligned} g(x) &\to f(f(x)) \\ f(f(a)) &\to b \end{aligned}$$

- LPO with precedence $g > f > b > a$
- complete and reduced TRS

Theorem

if complete reduced TRSs $\mathcal{R}$ and $\mathcal{S}$ satisfy

1. $\overset{*}{\underset{\mathcal{R}}{\longleftrightarrow}} = \overset{*}{\underset{\mathcal{S}}{\longleftrightarrow}}$
2. *$\mathcal{R}$ and $\mathcal{S}$ are compatible with same reduction order*

then $\mathcal{R} = \mathcal{S}$ (modulo variable renaming)

Outline

- Efficient Completion
- **Cola Gene Puzzle**
- Abstract Completion
- Proof Orders
- Critical Pair Criteria
- Further Reading

Example (Cola Gene Puzzle)

ES $\mathcal{E}$

$$\mathsf{TCAT} \approx \mathsf{T} \quad \mathsf{GAG} \approx \mathsf{AG} \quad \mathsf{CTC} \approx \mathsf{TC} \quad \mathsf{AGTA} \approx \mathsf{A} \quad \mathsf{TAT} \approx \mathsf{CT}$$

TRS $\mathcal{R}$

$$\mathsf{GA} \to \mathsf{A} \quad \mathsf{AGT} \to \mathsf{AT} \quad \mathsf{ATA} \to \mathsf{A} \quad \mathsf{CT} \to \mathsf{T} \quad \mathsf{TAT} \to \mathsf{T} \quad \mathsf{TCA} \to \mathsf{TA}$$

- $\mathcal{R}$ is reduced and complete
- $\xleftrightarrow[\mathcal{E}]{*} = \xleftrightarrow[\mathcal{R}]{*}$
- (milk gene) TAGCTAGCTAGCT $\xleftrightarrow[\mathcal{E}]{*}$ CTGACTGACT (cola gene)

$$\mathsf{TAGCTAGCTAGCT} \xrightarrow[\mathcal{R}]{!} \mathsf{T} \xleftarrow[\mathcal{R}]{!} \mathsf{CTGACTGACT}$$

- (milk gene) TAGCTAGCTAGCT $\not\xleftrightarrow[\mathcal{E}]{*}$ CTGCTACTGACT (mad cow retrovirus)

$$\mathsf{TAGCTAGCTAGCT} \xrightarrow[\mathcal{R}]{!} \mathsf{T} \neq \mathsf{TGT} \xleftarrow[\mathcal{R}]{!} \mathsf{CTGCTACTGACT}$$

Outline

- Efficient Completion
- Cola Gene Puzzle
- **Abstract Completion**
- Proof Orders
- Critical Pair Criteria
- Further Reading

Definition

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

inference system $\mathcal{SC}$ (standard completion) consists of six rules

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

Definitions

- $\trianglerighteq$ encompassment

  $s \trianglerighteq t \iff \exists$ position p $\exists$ substitution σ: $s|_p = t\sigma$

- $\triangleright$ strict encompassment

  $s \triangleright t \iff s \trianglerighteq t \wedge \neg(t \trianglerighteq s)$

Example

$\mathsf{s}(x) + \mathsf{s}(y+0) \triangleright \mathsf{s}(x) + y \qquad x + x \triangleright x + y \qquad x + y \not\triangleright x + x$

Definitions

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

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

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

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

- completion procedure is correct if every run that does not fail succeeds

Question

how to guarantee correctness ?

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

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

Lemmata

- *if* $(\mathcal{E}, \mathcal{R}) \vdash_{\mathcal{SC}} (\mathcal{E}', \mathcal{R}')$ *and* $\mathcal{R} \subseteq >$ *then* $\mathcal{R}' \subseteq >$
- *if* $(\mathcal{E}, \mathcal{R}) \vdash_{\mathcal{SC}} (\mathcal{E}', \mathcal{R}')$ *then* $\xleftrightarrow[\mathcal{E} \cup \mathcal{R}]{*} = \xleftrightarrow[\mathcal{E}' \cup \mathcal{R}']{*}$

Definition

$\mathcal{E}_\infty = \bigcup_{i \geqslant 0} \mathcal{E}_i$ and $\mathcal{R}_\infty = \bigcup_{i \geqslant 0} \mathcal{R}_i$

Lemmata

- $\mathcal{R}_\omega \subseteq \mathcal{R}_\infty \subseteq >$
- $\xleftrightarrow[\mathcal{E}]{*} = \xleftrightarrow[\mathcal{E}_\infty \cup \mathcal{R}_\infty]{*}$

Two Questions

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

1. is $\mathcal{R}_\omega$ confluent ?
2. $\xleftrightarrow[\mathcal{E}_\infty \cup \mathcal{R}_\infty]{*} = \xleftrightarrow[\mathcal{R}_\omega]{*}$?

Definitions

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

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

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

Theorem

every fair completion procedure is correct

Remark

strict encompassment condition in collapse rule cannot be dropped

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

Example

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

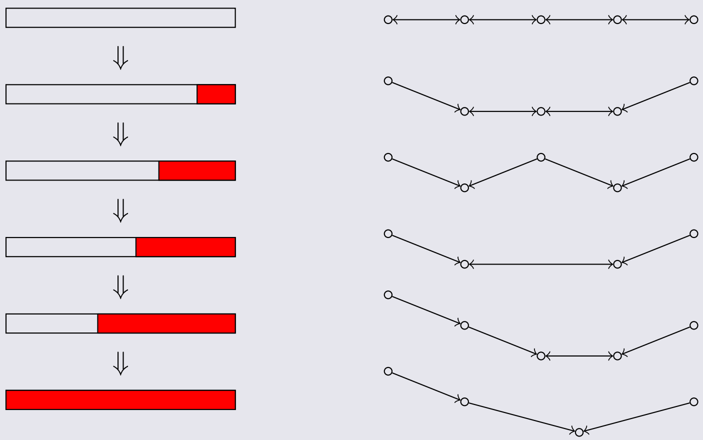
- LPO with precedence $\mathsf{f} > \mathsf{a} > \mathsf{g} > \mathsf{c} > \mathsf{b}$

Outline

- Efficient Completion
- Cola Gene Puzzle
- Abstract Completion
- Proof Orders
- Critical Pair Criteria
- Further Reading

Completion is Proof Normalization

$(\mathcal{E}, \mathcal{R})$ proof in $(\mathcal{E}, \mathcal{R})$



fair derivation rewrite proof

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

Definitions

- proof of $s \approx t$ is sequence $(u_1, \dots, u_n)$ of terms such that
  - $s = u_1$
  - $t = u_n$
  - for all $1 \leqslant i < n$ $u_i \to_{\mathcal{R}} u_{i+1}$ or $u_i \leftarrow_{\mathcal{R}} u_{i+1}$ or $u_i \leftrightarrow_{\mathcal{E}} u_{i+1}$
- rewrite proof is proof $(u_1, \dots, u_n)$ such that
  - $u_i \to_{\mathcal{R}} u_{i+1}$ for all $1 \leqslant i < j$
  - $u_i \leftarrow_{\mathcal{R}} u_{i+1}$ for all $j \leqslant i < n$

  for some $1 \leqslant j \leqslant n$
- two proofs $(s_1, \dots, s_n)$ and $(t_1, \dots, t_n)$ are equivalent if $s_1 = t_1$ and $s_n = t_n$

Definitions

- complexity of proof $(u_1, \dots, u_n)$ is multiset $\{c(u_1, u_2), \dots, c(u_{n-1}, u_n)\}$
- complexity of proof step (u_i, u_{i+1}) is triple

$$c(u_i, u_{i+1}) = \begin{cases} (\{u_i, u_{i+1}\}, -, -) & \text{if } u_i \leftrightarrow_{\mathcal{E}} u_{i+1} \\ (\{u_i\}, \ell, r) & \text{if } u_i \rightarrow_{\mathcal{R}} u_{i+1} \text{ using rule } \ell \rightarrow r \\ (\{u_{i+1}\}, \ell, r) & \text{if } u_i \leftarrow_{\mathcal{R}} u_{i+1} \text{ using rule } \ell \rightarrow r \end{cases}$$

- order $\gg$ on proof steps: lexicographic combination of
  - $>_{\mathrm{mul}}$ multiset extension of $>$
  - $\rhd$ strict encompassment
  - $>$

Lemma

$\gg_{mul}$ is a well-founded order on proofs

non-failing and fair run $\mathcal{S}$: $(\mathcal{E}_0, \mathcal{R}_0) \vdash_{SC} (\mathcal{E}_1, \mathcal{R}_1) \vdash_{SC} (\mathcal{E}_2, \mathcal{R}_2) \vdash_{SC} \cdots$

Lemma

$\forall$ proof P in $\mathcal{E}_\infty \cup \mathcal{R}_\infty$ that is no rewrite proof in $\mathcal{R}_\omega$
$\exists$ equivalent proof Q in $\mathcal{E}_\infty \cup \mathcal{R}_\infty$ such that $P \gg_{mul} Q$

Proof Sketch

three cases:

1. P contains step using equation $\ell \approx r \in \mathcal{E}_\infty$

   $\ell \approx r \notin \mathcal{E}_\omega$: consider how equation $\ell \approx r$ is removed in $\mathcal{S}$
2. P contains step using rule $\ell \rightarrow r \in \mathcal{R}_\infty \setminus \mathcal{R}_\omega$

   $\ell \rightarrow r \notin \mathcal{R}_\omega$: consider how rule $\ell \rightarrow r$ is removed in $\mathcal{S}$
3. P contains peak using rules from $\mathcal{R}_\omega$

   use critical pair lemma

Theorem

$\forall$ non-failing and fair run $(\mathcal{E}_0, \mathcal{R}_0) \vdash_{SC} (\mathcal{E}_1, \mathcal{R}_1) \vdash_{SC} (\mathcal{E}_2, \mathcal{R}_2) \vdash_{SC} \cdots$

- $\xleftrightarrow[\mathcal{E}_\infty \cup \mathcal{R}_\infty]{*} = \xleftrightarrow[\mathcal{R}_\omega]{*}$
- *$\mathcal{R}_\omega$ is complete*

Corollary

every fair completion procedure is correct

Outline

- Efficient Completion
- Cola Gene Puzzle
- Abstract Completion
- Proof Orders
- **Critical Pair Criteria**
- Further Reading

Fact

$CP(\mathcal{R}_\omega) \subseteq \mathcal{E}_\infty$ ensures correcteness

Question

are all critical pairs in $CP(\mathcal{R}_\omega)$ needed ?

Definitions

- critical pair criterion is mapping CPC on sets of equations such that $CPC(\mathcal{E}) \subseteq CP(\mathcal{E})$
- run $(\mathcal{E}_0, \mathcal{R}_0) \vdash_{\mathcal{SC}} (\mathcal{E}_1, \mathcal{R}_1) \vdash_{\mathcal{SC}} (\mathcal{E}_2, \mathcal{R}_2) \vdash_{\mathcal{SC}} \cdots$ is fair with respect to critical pair criterion CPC if $CP(\mathcal{R}_\omega) \setminus CPC(\mathcal{E}_\infty \cup \mathcal{R}_\infty) \subseteq \mathcal{E}_\infty$
- critical pair criterion CPC is correct if $\mathcal{R}_\omega$ is confluent and terminating for every non-failing run $(\mathcal{E}_0, \mathcal{R}_0) \vdash_{\mathcal{SC}} (\mathcal{E}_1, \mathcal{R}_1) \vdash_{\mathcal{SC}} (\mathcal{E}_2, \mathcal{R}_2) \vdash_{\mathcal{SC}} \cdots$ that is fair with respect to critical pair criterion CPC

Definitions

- peak $P\colon s \leftarrow_{\mathcal{R}} u \rightarrow_{\mathcal{R}} t$ is composite if there exist proofs

$$Q_1 : u_1 \overset{*}{\longleftrightarrow} u_2 \quad \cdots \quad Q_{n-1} : u_{n-1} \overset{*}{\longleftrightarrow} u_n$$

  such that

  - $s = u_1$
  - $t = u_n$
  - $\forall\, 1 \leqslant i \leqslant n \quad u > u_i$
  - $\forall\, 1 \leqslant i < n \quad P \gg_{\text{mul}} Q_i$

- critical pair $s \leftarrow\!\rtimes\!\rightarrow t$ is composite if corresponding peak $s \leftarrow \cdot \rightarrow t$ is composite

Definition

composite critical pair criterion: $CCP(\mathcal{E}) = \{ s \approx t \in CP(\mathcal{E}) \mid s \approx t \text{ is composite} \}$

Lemma

critical pair criterion CCP is correct

Question

how to check compositeness ?

Definition

- critical pair $s \leftarrow\!\rtimes\!\rightarrow t$ originating from overlap $\langle \ell_1 \to r_1, p, \ell_2 \to r_2 \rangle$ with mgu σ is unblocked if $x\sigma$ is reducible for some $x \in \mathcal{V}\text{ar}(\ell_1) \cup \mathcal{V}\text{ar}(\ell_2)$
- critical pair $s \leftarrow\!\rtimes\!\rightarrow t$ originating from overlap $\langle \ell_1 \to r_1, p, \ell_2 \to r_2 \rangle$ with mgu σ is reducible if proper subterm of $\ell_1\sigma$ is reducible

Lemma

- *every unblocked critical pair is composite*
- *every reducible critical pair is composite*

Example

TRS

$$\begin{aligned} e^- &\to e & x/e &\to x \\ x^{--} &\to x & e/x &\to x \\ x \cdot (x^- \cdot y) &\to y & (x/y^-)/y &\to x \\ x^- &\to e/x & z/(z^-/y)^- &\to y^- \end{aligned}$$

critical pair

$$y/e^- \leftarrow\!\rtimes\!\rightarrow y$$

originating from overlap

$$\langle\, x/e \to x,\ \epsilon,\ (y/z^-)/z \to y \,\rangle$$

is reducible because $(y/e^-)/e$ is reducible at position 12

Outline

- Efficient Completion
- Cola Gene Puzzle
- Abstract Completion
- Proof Orders
- Critical Pair Criteria
- Further Reading

Canonical Equational Proofs
Leo Bachmair
Progress in Theoretical Computer Science, Birkhäuser, 1991

Equational Inference, Canonical Proofs, and Proof Orderings
Leo Bachmair and Nachum Dershowitz
J.ACM 41(2), pp. 236–276, 1994

Completion Tools

- Waldmeister
- Slothrop
- mkbTT
- KBCV