Automated Theorem Proving

Lecture 10: Termination

Prof. Dr. Jasmin Blanchette

based on slides by Dr. Uwe Waldmann

Winter Term 2025/26

4.5 Termination

Termination problems:

Given a finite TRS R and a term t, are all R-reductions starting from t terminating?

Given a finite TRS R, are all R-reductions terminating?

Termination

Proposition 4.5.1:
Both termination problems for TRSs are undecidable in general.

Proof:
Encode Turing machines using rewrite rules and reduce the (uniform) halting problems for TMs to the termination problems for TRSs. □

Consequence:

Decidable criteria for termination are not complete.

Two Scenarios

Depending on the application, the TRS whose termination we want to show can be

(i) fixed and known in advance, or

(ii) evolving (e.g., generated by some saturation process).

Methods for case (ii) are also usable for case (i).
Many methods for case (i) are not usable for case (ii).

We will focus on case (ii).

Reduction Orderings

Goal:

Given a finite TRS R, show termination of R by looking at finitely many rules $l \to r \in R$, rather than at infinitely many possible replacement steps $s \to_R s'$.

Reduction Orderings

A binary relation $\sqsupset$ over $\mathrm{T}_\Sigma(X)$ is called compatible with Σ-operations if $s \sqsupset s'$ implies $f(t_1, \dots, s, \dots, t_n) \sqsupset f(t_1, \dots, s', \dots, t_n)$ for all $f \in \Omega$ and $s, s', t_i \in \mathrm{T}_\Sigma(X)$.

Lemma 4.5.2:
The relation $\sqsupset$ is compatible with Σ-operations if and only if $s \sqsupset s'$ implies $t[s]_p \sqsupset t[s']_p$ for all $s, s', t \in \mathrm{T}_\Sigma(X)$ and $p \in \mathrm{pos}(t)$.

Note: compatible with Σ-operations $=$ compatible with contexts.

Reduction Orderings

A binary relation $\sqsupset$ over $\mathrm{T}_\Sigma(X)$ is called stable under substitutions if $s \sqsupset s'$ implies $s\sigma \sqsupset s'\sigma$ for all $s, s' \in \mathrm{T}_\Sigma(X)$ and substitutions σ.

Reduction Orderings

A binary relation $\sqsupset$ is called a rewrite relation if it is compatible with Σ-operations and stable under substitutions.

Example: If R is a TRS, then $\rightarrow_R$ is a rewrite relation.

A strict partial ordering over $\mathrm{T}_\Sigma(X)$ that is a rewrite relation is called rewrite ordering.

A well-founded rewrite ordering is called reduction ordering.

Reduction Orderings

Theorem 4.5.3:
A TRS R terminates if and only if there exists a reduction ordering $\succ$ such that $l \succ r$ for every rule $l \to r \in R$.

The Interpretation Method

Proving termination by interpretation:

Let $\mathcal{A}$ be a Σ-algebra;
let $\succ$ be a well-founded strict partial ordering on its universe.

Define the ordering $\succ_{\mathcal{A}}$ over $\mathrm{T}_{\Sigma}(X)$ by $s \succ_{\mathcal{A}} t$ if and only if $\mathcal{A}(\beta)(s) \succ \mathcal{A}(\beta)(t)$ for all assignments $\beta : X \to U_{\mathcal{A}}$.

Is $\succ_{\mathcal{A}}$ a reduction ordering?

The Interpretation Method

Lemma 4.5.4:

$\succ_{\mathcal{A}}$ is stable under substitutions.

The Interpretation Method

A function $\phi : U_{\mathcal{A}}^n \to U_{\mathcal{A}}$ is called monotone (w.r.t. $\succ$)
if $a \succ a'$ implies $\phi(b_1, \ldots, a, \ldots, b_n) \succ \phi(b_1, \ldots, a', \ldots, b_n)$
for all $a, a', b_i \in U_{\mathcal{A}}$.

Lemma 4.5.5:
If the interpretation $f_{\mathcal{A}}$ of every function symbol f is monotone w.r.t. $\succ$,
then $\succ_{\mathcal{A}}$ is compatible with Σ-operations.

Theorem 4.5.6:
If the interpretation $f_{\mathcal{A}}$ of every function symbol f is monotone w.r.t. $\succ$,
then $\succ_{\mathcal{A}}$ is a reduction ordering.

Polynomial Orderings

Polynomial orderings:

Instance of the interpretation method:

The carrier set $U_{\mathcal{A}}$ is $\mathbb{N}$ or some subset of $\mathbb{N}$.

With every function symbol f/n we associate a polynomial $P_f(X_1, \ldots, X_n) \in \mathbb{N}[X_1, \ldots, X_n]$ with coefficients in $\mathbb{N}$ and indeterminates $X_1, \ldots, X_n$. Then we define $f_{\mathcal{A}}(a_1, \ldots, a_n) = P_f(a_1, \ldots, a_n)$ for $a_i \in U_{\mathcal{A}}$.

Polynomial Orderings

Requirement 1:

If $a_1, \ldots, a_n \in U_\mathcal{A}$, then $f_\mathcal{A}(a_1, \ldots, a_n) \in U_\mathcal{A}$.
(Otherwise, $\mathcal{A}$ would not be a Σ-algebra.)

Polynomial Orderings

Requirement 2:

$f_{\mathcal{A}}$ must be monotone (w.r.t. $\succ$).

From now on:

$U_{\mathcal{A}} = \{n \in \mathbb{N} \mid n \geq 1\}$.

If arity$(f) = 0$, then P_f is a constant ≥ 1.

If arity$(f) = n \geq 1$, then P_f is a polynomial $P(X_1, \ldots, X_n)$ such that every X_i occurs in some monomial $m \cdot X_1^{j_1} \cdots X_k^{j_k}$ with exponent at least 1 and nonzero coefficient $m \in \mathbb{N}$.

$\Rightarrow$ Requirements 1 and 2 are satisfied.

Polynomial Orderings

The mapping from function symbols can be extended to terms:
A term t containing the variables $x_1, \ldots, x_n$
yields a polynomial P_t with indeterminates $X_1, \ldots, X_n$
(where X_i corresponds to $\beta(x_i)$).

Example:

$$\Omega = \{b/0,\ f/1,\ g/3\}$$
$$P_b = 3, \quad P_f(X_1) = X_1^2, \quad P_g(X_1, X_2, X_3) = X_1 + X_2X_3.$$

Let $t = g(f(b), f(x), y)$, then $P_t(X, Y) = 9 + X^2Y$.

Polynomial Orderings

Given polynomials P, Q in $\mathbb{N}[X_1, \ldots, X_n]$, we write $P > Q$ if $P(a_1, \ldots, a_n) > Q(a_1, \ldots, a_n)$ for all $a_1, \ldots, a_n \in U_{\mathcal{A}}$.

Clearly, $s \succ_{\mathcal{A}} t$ if and only if $P_s > P_t$ if and only if $P_s - P_t > 0$.

Question: Can we check $P_s - P_t > 0$ automatically?

Polynomial Orderings

Hilbert's 10th Problem:

Given a polynomial $P \in \mathbb{Z}[X_1, \ldots, X_n]$ with integer coefficients,
is $P = 0$ for some n-tuple of natural numbers?

Theorem 4.5.7:
Hilbert's 10th Problem is undecidable.

Proposition 4.5.8:
Given a polynomial interpretation and two terms s, t, it is undecidable whether $P_s > P_t$.

Proof:
By reduction of Hilbert's 10th Problem. $\square$

Polynomial Orderings

One easy case:

If we restrict to linear polynomials, deciding whether $P_s - P_t > 0$ is trivial:

$\sum k_i a_i + k > 0$ for all $a_1, \ldots, a_n \geq 1$ if and only if

$k_i \geq 0$ for all $i \in \{1, \ldots, n\}$,

and $\sum k_i + k > 0$

Polynomial Orderings

Another possible solution:

Test whether $P_s(a_1, \ldots, a_n) > P_t(a_1, \ldots, a_n)$
for all $a_1, \ldots, a_n \in \{x \in \mathbb{R} \mid x \geq 1\}$.

This is decidable (but hard).
Since $U_{\mathcal{A}} \subseteq \{x \in \mathbb{R} \mid x \geq 1\}$, it implies $P_s > P_t$.

Alternatively:

Use fast overapproximations.

Simplification Orderings

The proper subterm ordering $\rhd$ is defined by $s \rhd t$ if and only if $s|_p = t$ for some position $p \neq \varepsilon$ of s.

Simplification Orderings

A rewrite ordering $\succ$ over $T_\Sigma(X)$ is called simplification ordering if it has the subterm property:
$s \rhd t$ implies $s \succ t$ for all $s, t \in T_\Sigma(X)$.

Example:

Let R_{emb} be the rewrite system
$R_{\text{emb}} = \{f(x_1, \dots, x_n) \to x_i \mid f/n \in \Omega, 1 \leq i \leq n\}$.

Define $\rhd_{\text{emb}} = \to^{+}_{R_{\text{emb}}}$ and $\unrhd_{\text{emb}} = \to^{*}_{R_{\text{emb}}}$
("homeomorphic embedding relation").

$\rhd_{\text{emb}}$ is a simplification ordering.

Simplification Orderings

Lemma 4.5.9:
If $\succ$ is a simplification ordering, then $s \rhd_{\text{emb}} t$ implies $s \succ t$ and $s \trianglerighteq_{\text{emb}} t$ implies $s \succeq t$.

Simplification Orderings

Goal:

Show that every simplification ordering is well-founded (and therefore a reduction ordering).

Note: This works only for finite signatures.

To fix this for infinite signatures, the definition of simplification orderings and the definition of embedding have to be modified.

Simplification Orderings

Theorem 4.5.10 (“Kruskal’s Theorem”):
Let Σ be a finite signature, and let X be a finite set of variables. Then for every infinite sequence $t_1, t_2, t_3, \ldots$ there are indices $j > i$ such that $t_j \trianglerighteq_{\text{emb}} t_i$.
($\trianglerighteq_{\text{emb}}$ is called a well-partial-ordering (wpo).)

Proof:
See Baader and Nipkow, pages 113–115. $\square$

Simplification Orderings

Theorem 4.5.11 (Dershowitz):
If Σ is a finite signature, then every simplification ordering $\succ$ on $\mathrm{T}_\Sigma(X)$ is well-founded (and therefore a reduction ordering).

Simplification Orderings

There are reduction orderings that are not simplification orderings and terminating TRSs that are not contained in any simplification ordering.

Example:

Let $R = \{f(f(x)) \to f(g(f(x)))\}$.

R terminates and $\to_R^+$ is therefore a reduction ordering.

Assume that $\to_R$ were contained in a simplification ordering $\succ$. Then $f(f(x)) \to_R f(g(f(x)))$ implies $f(f(x)) \succ f(g(f(x)))$, and $f(g(f(x))) \trianglerighteq_{\text{emb}} f(f(x))$ implies $f(g(f(x))) \succeq f(f(x))$, hence $f(f(x)) \succ f(f(x))$.

Path Orderings

Let $\Sigma = (\Omega, \Pi)$ be a finite signature, let $\succ$ be a strict partial ordering ("precedence") on Ω.

The lexicographic path ordering $\succ_{\mathrm{lpo}}$ on $T_\Sigma(X)$ induced by $\succ$ is defined by: $s \succ_{\mathrm{lpo}} t$ if

(1) $t \in \mathrm{var}(s)$ and $t \neq s$, or

(2) $s = f(s_1, \dots, s_m)$, $t = g(t_1, \dots, t_n)$, and

- (a) $s_i \succeq_{\mathrm{lpo}} t$ for some i, or
- (b) $f \succ g$ and $s \succ_{\mathrm{lpo}} t_j$ for all j, or
- (c) $f = g$, $s \succ_{\mathrm{lpo}} t_j$ for all j, and $(s_1, \dots, s_m)\ (\succ_{\mathrm{lpo}})_{\mathrm{lex}}\ (t_1, \dots, t_n)$.

Path Orderings

Lemma 4.5.12:
$s \succ_{\text{lpo}} t$ implies $\text{var}(s) \supseteq \text{var}(t)$.

Theorem 4.5.13:
$\succ_{\text{lpo}}$ is a simplification ordering on $\mathrm{T}_\Sigma(X)$.

Theorem 4.5.14:
If the precedence $\succ$ is total, then the lexicographic path ordering $\succ_{\text{lpo}}$ is total on ground terms, i.e., for all $s, t \in \mathrm{T}_\Sigma(\emptyset)$:
$s \succ_{\text{lpo}} t \vee t \succ_{\text{lpo}} s \vee s = t$.

Path Orderings

Recapitulation:

Let $\Sigma = (\Omega, \Pi)$ be a finite signature, let $\succ$ be a strict partial ordering ("precedence") on Ω. The lexicographic path ordering $\succ_{\mathsf{lpo}}$ on $\mathrm{T}_\Sigma(X)$ induced by $\succ$ is defined by: $s \succ_{\mathsf{lpo}} t$ if

(1) $t \in \mathrm{var}(s)$ and $t \neq s$, or

(2) $s = f(s_1, \dots, s_m)$, $t = g(t_1, \dots, t_n)$, and

- (a) $s_i \succeq_{\mathsf{lpo}} t$ for some i, or
- (b) $f \succ g$ and $s \succ_{\mathsf{lpo}} t_j$ for all j, or
- (c) $f = g$, $s \succ_{\mathsf{lpo}} t_j$ for all j, and $(s_1, \dots, s_m)\ (\succ_{\mathsf{lpo}})_{\mathsf{lex}}\ (t_1, \dots, t_n)$.

Path Orderings

There are several possibilities to compare subterms in (2)(c):

- compare list of subterms lexicographically left-to-right
  ("lexicographic path ordering (lpo)," Kamin and Lévy)
- compare list of subterms lexicographically right-to-left
  (or according to some permutation π)
- compare multiset of subterms using the multiset extension
  ("multiset path ordering (mpo)," Dershowitz)
- with each function symbol $f/n \in \Omega$ with $n \geq 1$ associate a status $\in \{\text{mul}\} \cup \{\text{lex}_\pi \mid \pi : \{1, \ldots, n\} \to \{1, \ldots, n\}\}$
  and compare according to that status
  ("recursive path ordering (rpo) with status")

Path Orderings

Example 4.5.15:
Consider the following set of equations:

$$
\begin{aligned}
f(h(h(x))) &\approx h(f(f(x))) \\
g(g(x)) &\approx f(h(f(h(h(f(x)))))) \\
f(h(x)) &\approx f(f(x))
\end{aligned}
$$

Using the lpo with the precedence $g \succ h \succ f$, the left-hand side of each equation is greater than the corresponding right-hand side.

The Knuth–Bendix Ordering

Let $\Sigma = (\Omega, \Pi)$ be a finite signature,
let $\succ$ be a strict partial ordering ("precedence") on Ω,
let $w : \Omega \cup X \to \mathbb{R}_0^+$ be a weight function
such that the following admissibility conditions are satisfied:

$w(x) = w_0 \in \mathbb{R}^+$ for all variables $x \in X$;
$w(c) \geq w_0$ for all constants $c \in \Omega$.

If $w(f) = 0$ for some $f/1 \in \Omega$, then $f \succ g$ for all $g/n \in \Omega$ with $f \neq g$.

The Knuth–Bendix Ordering

The weight function w can be extended to terms recursively:

$$w(f(t_1, \ldots, t_n)) = w(f) + \sum_{1 \leq i \leq n} w(t_i)$$

or alternatively

$$w(t) = \sum_{x \in \mathrm{var}(t)} w(x) \cdot \#(x, t) + \sum_{f \in \Omega} w(f) \cdot \#(f, t)$$

where $\#(a, t)$ is the number of occurrences of a in t.

The Knuth–Bendix Ordering

The Knuth–Bendix ordering $\succ_{\mathrm{kbo}}$ on $\mathrm{T}_\Sigma(X)$ induced by $\succ$ and w is defined by: $s \succ_{\mathrm{kbo}} t$ if

(1) $\#(x, s) \geq \#(x, t)$ for all variables x and $w(s) > w(t)$, or

(2) $\#(x, s) \geq \#(x, t)$ for all variables x, $w(s) = w(t)$, and

- (a) $t = x$, $s = f^n(x)$ for some $n \geq 1$, or
- (b) $s = f(s_1, \dots, s_m)$, $t = g(t_1, \dots, t_n)$, and $f \succ g$, or
- (c) $s = f(s_1, \dots, s_m)$, $t = f(t_1, \dots, t_m)$, and $(s_1, \dots, s_m)\ (\succ_{\mathrm{kbo}})_{\mathrm{lex}}\ (t_1, \dots, t_m)$.

The Knuth–Bendix Ordering

Theorem 4.5.16:

The Knuth–Bendix ordering induced by $\succ$ and w is a simplification ordering on $\mathrm{T}_\Sigma(X)$.

Proof:

See Baader and Nipkow, pages 125–129. $\square$

The Knuth–Bendix Ordering

Example 4.5.17:

Consider the following set of equations:

$$f(h(h(x))) \approx h(f(f(x)))$$
$$g(g(x)) \approx f(h(f(h(h(f(x))))))$$
$$f(h(x)) \approx f(f(x))$$

Using the kbo with weight 100 for g, weight 10 for h, weight 1 for f and variables, and an arbitrary precedence, the left-hand side of each equation is greater than the corresponding right-hand side.

Remark

If $\Pi \neq \emptyset$, then all the term orderings described in this section can also be used to compare nonequational atoms by treating predicate symbols like function symbols.