3.3a Term Rewriting Systems (Part 1)

Donnerstag, 5. November 2015 12:30

For ground identities, the word problem is decidable by congruence closure. Now we want to regard the general case where $\mathcal{E}, s, t$ can contain variables.

By Birkhoff's Theorem: To check $s \equiv_{\mathcal{E}} t$, one has to investigate $s \leftrightarrow^*_{\mathcal{E}} t$. This has an enormous search space.

To reduce the search space: restrict orientation of equations such that they can only be applied from left to right.

<u>Def 3.3.1</u> (Term Rewriting System, TRS)

For $\ell, r \in \mathcal{T}(\Sigma, \mathcal{V})$, we say that $\ell \to r$ is a <u>rule</u> over Σ and $\mathcal{V}$ iff

- $\mathcal{V}(r) \subseteq \mathcal{V}(\ell)$
- $\ell \notin \mathcal{V}$

A set $\mathcal{R}$ of rules is called a <u>term rewriting system</u> (TRS).

For a TRS $\mathcal{R}$, the <u>rewrite relation</u> $\to_{\mathcal{R}} \subseteq \mathcal{T}(\Sigma, \mathcal{V}) \times \mathcal{T}(\Sigma, \mathcal{V})$ is defined as:

$$s \to_{\mathcal{R}} t \quad \text{iff} \quad s|_\pi = \ell\sigma \text{ and } t = s[r\sigma]_\pi$$

for some $\pi \in Occ(s)$, some $\ell \to r \in \mathcal{R}$, some $\sigma \in SUB(\Sigma, \mathcal{V})$.

The subterm $s|_\pi$ is called <u>redex</u> ("<u>red</u>ucible <u>ex</u>pression")

Sometimes we write "$s \to t$" instead of "$s \to_{\mathcal{R}} t$" if $\mathcal{R}$ is clear from the context.

Reason for "$\mathcal{V}(r) \subseteq \mathcal{V}(\ell)$": this restricts the search space. Now the matcher of ℓ uniquely determines the instantiation of $\mathcal{R}$. $\rightsquigarrow$ For any term s, there are only finitely many terms t with $s \to_{\mathcal{R}} t$ (if $\mathcal{R}$ is finite). $\Rightarrow$ Breadth of search tree for the word problem is finite.

Reason for "$\ell \notin \mathcal{V}$": also restricts the search space.

A rule $x \to t$ would always be applicable, since a variable x matches <u>any</u> term.

<u>Ex. 3.3.2</u> We now want to use TRSs $\mathcal{R}$ instead of sets of equations $\mathcal{E}$.

For groups:

$$f(x, f(y,z)) \to f(f(x,y), z)$$
$$f(x, e) \to x$$
$$f(x, i(x)) \to e$$

Similarly for addition.

$$\underline{plus(s(s(0)), s(0))} \to_{\mathcal{R}} s(\underline{plus(s(0), s(0))})$$
$$\to_{\mathcal{R}} s(s(\underline{plus(0, s(0))}))$$
$$\to_{\mathcal{R}} s(s(s(0)))$$

Term Rewriting is very simple, but it is already a

Turing-Complete programming language (i.e., every computable function can be computed by a TRS).

Clearly: $\to_{\mathcal{R}}$ is stable and monotonic

$\to_{\mathcal{R}}^{*}$ is $\underbrace{\text{reflexive and transitive (+ stable + monotonic)}}$

$\leftrightarrow_{\mathcal{R}}^{*}$ is symmetric and "

To solve the word problem for $\mathcal{E}$: $\leftarrow$ $s \equiv_{\mathcal{E}} t$ iff $s \leftrightarrow_{\mathcal{E}}^{*} t$

- transform set of equations $\mathcal{E}$ into an equivalent TRS $\mathcal{R}$
- use $\mathcal{R}$ to solve the word problem $\leftarrow$ $s \leftrightarrow_{\mathcal{R}}^{*} t$

Def 3.3.3 (Equivalence of Set of Equations and TRS)

Let $\mathcal{E}$ be a set of equations and $\mathcal{R}$ be a TRS.
Then $\mathcal{R}$ is equivalent to $\mathcal{E}$ iff $\leftrightarrow_{\mathcal{E}}^{*} = \leftrightarrow_{\mathcal{R}}^{*}$.

Reason: $s \equiv_{\mathcal{E}} t$ iff $s \leftrightarrow_{\mathcal{E}}^{*} t$ iff $s \leftrightarrow_{\mathcal{R}}^{*} t$.

Birkhoff's Thm — if $\mathcal{R}$ and $\mathcal{E}$ are equivalent

To check whether $\mathcal{R}$ is equivalent to $\mathcal{E}$, it suffices to regard $\mathcal{E}$ and $\mathcal{R}$ instead of $\leftrightarrow^*_{\mathcal{E}}$ and $\leftrightarrow^*_{\mathcal{R}}$:

<u>Thm 334</u> (Connection between Sets of Equations and TRSs)

Let $\mathcal{E}$ be a set of equations and $\mathcal{R}$ be a TRS.
$\mathcal{R}$ is equivalent to $\mathcal{E}$ iff

- $l \leftrightarrow^*_{\mathcal{E}} r$ for all rules $l \to r \in \mathcal{R}$ ("$\mathcal{R}$ is <u>sound</u> for $\mathcal{E}$")
- $s \leftrightarrow^*_{\mathcal{R}} t$ for all equations $s \equiv t \in \mathcal{E}$ ("$\mathcal{R}$ is <u>adequate</u> for $\mathcal{E}$")

<u>Proof</u>: "$\Rightarrow$": Equivalence trivially implies that $\mathcal{R}$ is sound and adequate for $\mathcal{E}$.

"$\Leftarrow$": Equivalence follows from Soundness and Adequateness: Follows directly from the fact that $\leftrightarrow^*_{\mathcal{E}}$ and $\leftrightarrow^*_{\mathcal{R}}$ are stable, monotonic, reflexive, transitive, and symmetric. $\square$

If one simply replaces "$\equiv$" by "$\to$" in $\mathcal{E}$, then one clearly obtains an equivalent TRS. But these TRSs are not always advantageous, i.e., we sometimes want other TRSs that are still equivalent to $\mathcal{E}$

Ex. 5.35 $\mathcal{E} = \{b \equiv c,\ b \equiv a,\ f(a) \equiv f(f(a))\}$

$(\Sigma = \{a, b, c, f\})$.

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

$\mathcal{R}$ is sound for $\mathcal{E}$, because:

- $c \leftarrow_{\mathcal{E}} b$
- $a \leftarrow_{\mathcal{E}} b$
- $f(f(b)) \to_{\mathcal{E}} f(f(a)) \leftarrow_{\mathcal{E}} f(a) \leftarrow_{\mathcal{E}} f(b)$

$\mathcal{R}$ is adequate for $\mathcal{E}$, because:

- $b \leftarrow_{\mathcal{R}} c$
- $b \leftarrow_{\mathcal{R}} a$
- $f(a) \to_{\mathcal{R}} f(b) \leftarrow_{\mathcal{R}} f(f(b)) \leftarrow_{\mathcal{R}} f(f(a))$

Thus: $\mathcal{R}$ is equivalent to $\mathcal{E}$.

Why don't we just take the equations $\mathcal{E}$ and replace "$\equiv$" by "$\to$" in order to obtain an equivalent TRS?

Reason: The rules of $\mathcal{R}$ should not be applied in both directions (in order to reduce the search space).

Goal: To solve $s \equiv_{\mathcal{E}} t$, choose an equivalent

"Suitable" TRS $\mathcal{R}$.

To check $s \leftrightarrow^*_{\mathcal{R}} t$, we want to proceed as follows:

- Rewrite both s and t as long as possible:

$$s \to_{\mathcal{R}} s_1 \to_{\mathcal{R}} s_2 \to_{\mathcal{R}} \dots \to_{\mathcal{R}} s_n$$

$$t \to_{\mathcal{R}} t_1 \to_{\mathcal{R}} t_2 \to_{\mathcal{R}} \dots \to_{\mathcal{R}} t_m$$

- Check whether the resulting terms s_n and t_m are syntactically equal.

  If yes: return "true" (i.e., $s \equiv_{\mathcal{E}} t$ holds)

  If no: return "false" (i.e., $s \equiv_{\mathcal{E}} t$ does not hold).

Ex 336 Use the plus-TRS to check

$$plus(s(s(\mathcal{O})), x) \equiv_{\mathcal{E}} plus(s(\mathcal{O}), s(x))$$

$$plus(s(s(\mathcal{O})), x) \to_{\mathcal{R}} s(plus(s(\mathcal{O}), x)) \to_{\mathcal{R}} s(s(plus(\mathcal{O}, x))) \to_{\mathcal{R}} s(s(x))$$

$$plus(s(\mathcal{O}), s(x)) \to_{\mathcal{R}} s(plus(\mathcal{O}, s(x))) \to_{\mathcal{R}} s(s(x))$$

Alg. returns "True".

If the algorithm returns "True", then we really have

$s \equiv_\mathcal{E} t$. But the algorithm has 2 problems:

1. It can happen that the reduction of s or t does not <u>terminate</u> (this can also happen if $s \equiv_\mathcal{E} t$, i.e., the algorithm is not even a semi-decision procedure for the word problem)

   Solution: require that $\mathcal{R}$ is <u>terminating</u>

2. It can happen that $s \leftrightarrow^*_\mathcal{R} t$ holds, but there is no term q such that $s \to^*_\mathcal{R} q \leftarrow^*_\mathcal{R} t$.
   So the alg. could return "False" although $s \equiv_\mathcal{E} t$ holds.

   Solution: require that $\mathcal{R}$ is <u>confluent</u>

<u>Ex 337</u>

$\mathcal{E} = \{b \equiv c,\ b \equiv a,\ f(a) \equiv f(f(a))\}$

$\mathcal{R}_1 = \{b \to c,\ b \to a,\ f(a) \to f(f(a))\}$ is clearly equivalent to $\mathcal{E}$.

Check whether $f(a) \equiv_\mathcal{E} f(c)$ holds.

$$f(a) \to_{\mathcal{R}_1} f(f(a)) \to_{\mathcal{R}} f(f(f(a))) \to_{\mathcal{R}_1} f^4(a) \to_{\mathcal{R}_1} \dots$$

$\Rightarrow$ Alg. does not terminate.

Indeed, $\mathcal{R}_1$ is not terminating.

$\mathcal{R}_2 = \{b \to c,\ b \to a,\ f(f(a)) \to f(a)\}$ is equivalent to $\mathcal{E}$

Check whether $f(a) \equiv_E f(c)$ and terminating

Neither $f(a)$ nor $f(c)$ can be reduced further.

$\Rightarrow$ Alg. returns "False" since $f(a)$ and $f(c)$ are not syntactically equal. (This is the wrong answer, since $f(a) \equiv_E f(c)$ holds.)

Indeed, $\mathcal{R}_2$ is not confluent.