UPPSALA UNIVERSITET
Matematiska institutionen
Erik Palmgren

Lecture notes
Applied Logic
(Tillämpad logik)
2007-11-06

Equational logic, unification and term rewriting

1 Equational logic

Below we shall largely follow Klop (1992) in the presentation of equational logic and unification.

1.1 Some notions from universal algebra

In *universal algebra* properties of general algebraic systems are studied. These systems include the usual, groups, semigroups, monoids, rings, but also systems with operations of arbitrary number of arguments. In algebraic specification theory these operations may describe programs or hardware components. (See Meinke and Tucker 1992, Goguen and Malcolm 1996 and Wechler 1992.)

A *signature* Σ is a set of function symbols, where each $F \in \Sigma$ takes a fixed number $n(F)$ (the *arity*) of arguments. 0-ary function symbols are considered as constant symbols. (Thus a signature is like a description of a first order language but without relation symbols.) A Σ*-algebra* $\mathcal{A}$ consists of an underlying nonempty set A, and for each function symbol $F \in \Sigma$, an operation

$$F^{\mathcal{A}} : A^{n(F)} \to A,$$

for $n(F) > 0$. If $n(F) = 0$, $F^{\mathcal{A}} \in A$.

Homomorphisms, mappings which preserves the operations of an algebra are of central importance. Let $\mathcal{A}$ and $\mathcal{B}$ be Σ-algebras. A *(*Σ*-algebra) homomorphism* $\varphi : \mathcal{A} \to \mathcal{B}$ is function between the underlying sets $\varphi : A \to B$ which is such that for every function symbol $F \in \Sigma$ of arity n we have for all $a_1, \ldots, a_n \in A$:

$$\varphi(F^{\mathcal{A}}(a_1, \ldots, a_n)) = F^{\mathcal{B}}(\varphi(a_1), \ldots, \varphi(a_n)).$$

If $n = 0$, this reads $\varphi(F^{\mathcal{A}}) = F^{\mathcal{B}}$.

Example 1.1 The embedding $\mathbb{Z} \hookrightarrow \mathbb{Q}$ and the quotient mapping $x \mapsto x \bmod n : \mathbb{Z} \to \mathbb{Z}/n\mathbb{Z}$ are basic examples of homomorphisms with respect to the signature $\Sigma = \{0, 1, +, \cdot\}$.

There is always a trivial homomorphism $\mathcal{A} \to \mathcal{A}$, the identity homomorphism $\mathrm{id}_{\mathcal{A}}$ defined by $\mathrm{id}_{\mathcal{A}}(x) = x$. A homomorphism $\varphi : \mathcal{A} \to \mathcal{B}$ is an *isomorphism* if there is a homomorphism $\psi : \mathcal{B} \to \mathcal{A}$ such that $\psi \circ \varphi = \mathrm{id}_{\mathcal{A}}$ and $\varphi \circ \psi = \mathrm{id}_{\mathcal{B}}$. We leave the verification of the following result to the reader:

Proposition 1.2 *A homomorphism $\varphi : \mathcal{A} \to \mathcal{B}$ is an isomorphism iff $\varphi : A \to B$ is a bijection.* □

Let $\mathrm{Ter}(\Sigma)$ be the set of terms that can be formed from the function symbols in Σ and variables from a fixed set of variable symbols $\mathbb{X} = \{x_1, x_2, x_3, \ldots\}$. The set $\mathrm{Ter}(\Sigma)$ is inductively defined by the following clauses

(T1) If $x \in \mathbb{X}$, then $x \in \mathrm{Ter}(\Sigma)$.

(T2) If $F \in \Sigma$ and $n(F) = 0$, then $F \in \mathrm{Ter}(\Sigma)$.

(T3) If $F \in \Sigma$, $n = n(F) > 0$ and $t_1, \ldots, t_n \in \mathrm{Ter}(\Sigma)$, then $F(t_1, \ldots, t_n) \in \mathrm{Ter}(\Sigma)$.

Since the set $\mathrm{Ter}(\Sigma)$ is inductively defined, we may prove properties of terms by structural induction. We may also define functions on terms by structural recursion. A *substitution* is a function $\sigma : \mathbb{X} \to \mathrm{Ter}(\Sigma)$, assigning to each variable symbol a term. The effect t^σ of a substitution σ on a term t is defined recursively

$$\begin{array}{rcll} x_i^\sigma & = & \sigma(x_i) & \\ F^\sigma & = & F & (n(F) = 0) \\ F(t_1, \ldots, t_n)^\sigma & = & F(t_1^\sigma, \ldots, t_n^\sigma) & (n = n(F)) \end{array}$$

Thus we may extend σ to a function $\mathrm{Ter}(\Sigma) \to \mathrm{Ter}(\Sigma)$ by $\sigma(t) = t^\sigma$. Denote by

$$\{x_{i_1} := t_1, \ldots, x_{i_k} := t_k\},$$

where $i_1 < i_2 < \cdots < i_k$, the substitution σ where $\sigma(x_{i_j}) = t_j$ for $j = 1, \ldots, k$ and $\sigma(x_i) = x_i$ for $i \notin \{i_1, i_2, \ldots, i_k\}$.

Example 1.3 Let $\Sigma = \{0, f, g\}$ where the arities are $n(0) = 0$, $n(f) = 1$ and $n(g) = 2$. Then 0, $f(0)$, $g(x_1, f(x_3))$ are examples of terms over Σ. For the substitution $\sigma = \{x_1 := g(x_1, x_3), x_2 := f(0), x_3 := x_2\}$ we have

$$g(x_1, f(x_3))^\sigma = g(x_1^\sigma, f(x_3^\sigma)) = g(g(x_1, x_3), f(x_2)). \ \square$$

The set $\mathrm{Ter}(\Sigma)$ can be regarded as a Σ-algebra — in a kind of trivial way — by defining for each n-ary function symbol $F \in \Sigma$, a function $F^{\mathrm{Ter}(\Sigma)}$ by

$$F^{\mathrm{Ter}(\Sigma)}(t_1, \ldots, t_n) = F(t_1, \ldots, t_n).$$

We call $\mathrm{Ter}(\Sigma)$ the *term algebra of* Σ. We may also restrict ourselves to terms without variables (in case there are constant symbols) The resulting set, $\mathrm{Ter}_0(\Sigma)$, also forms a Σ-algebra.

Note that any substitution $\sigma : \mathbb{X} \to \mathrm{Ter}(\Sigma)$ extends to a Σ-algebra homomorphism $\sigma : \mathrm{Ter}(\Sigma) \to \mathrm{Ter}(\Sigma)$. (Exercise: verify this.)

Let $\mathcal{A}$ be a Σ-algebra. A *variable assignment* or *environment* in $\mathcal{A}$ is a function $\rho : \mathbb{X} \to A$. Given such an assignment, the value $[\![t]\!]^{\mathcal{A}}_{\rho}$ of a term t in $\mathcal{A}$ is determined. Define by recursion on t:

$$\begin{array}{rcl} [\![x_i]\!]_\rho & = & \rho(x_i), \\ [\![F(t_1, \ldots, t_m)]\!]_\rho & = & F^{\mathcal{A}}([\![t_1]\!]_\rho, \ldots, [\![t_m]\!]_\rho). \end{array}$$

An equation $s = t$ *is valid in* $\mathcal{A}$ (in symbols: $\mathcal{A} \models s = t$) iff for all variable assignments ρ in $\mathcal{A}$: $[\![s]\!]^{\mathcal{A}}_{\rho} = [\![t]\!]^{\mathcal{A}}_{\rho}$.

An equational theory over Σ is given by a set E of equations $s = t$ where $s, t \in \mathrm{Ter}(\Sigma)$. The deduction rules of an equational theory essentially only tell how instances of these equations may be used to calculate inside terms. We denote by $E \vdash_{\mathrm{eq}} s = t$ that $s = t$ is derivable from E. The deduction rules are more formally

$$(\text{ax.appl.}:) \qquad E \vdash_{\mathrm{eq}} s = t \qquad \text{if } s = t \in E$$

$$\frac{E \vdash_{\mathrm{eq}} s = t}{E \vdash_{\mathrm{eq}} s^{\sigma} = t^{\sigma}}\,(\text{subst}) \qquad \text{for every substitution } \sigma : \mathbb{X} \to \mathrm{Ter}(\Sigma)$$

$$\frac{E \vdash_{\mathrm{eq}} s_1 = t_1 \quad \cdots \quad E \vdash_{\mathrm{eq}} s_n = t_n}{E \vdash_{\mathrm{eq}} F(s_1, \ldots, s_n) = F(t_1, \ldots, t_n)}\,(\text{cong}) \qquad \text{for every } F \in \Sigma \text{ with } n = n(F)$$

$$(\text{refl}:) \qquad E \vdash_{\mathrm{eq}} t = t \qquad \text{for every } t \in \mathrm{Ter}(\Sigma)$$

$$\frac{E \vdash_{\mathrm{eq}} s = t}{E \vdash_{\mathrm{eq}} t = s}\,(\text{symm})$$

$$\frac{E \vdash_{\mathrm{eq}} s = v \qquad E \vdash_{\mathrm{eq}} v = t}{E \vdash_{\mathrm{eq}} s = t}\,(\text{trans})$$

Example 1.4 *The equational theory of groups.* Let $\Sigma = \{1, \cdot, (\,)^{-1}\}$, where the arities are 0, 2 and 1 respectively. The equations E are

$$\begin{gathered} 1 \cdot x_1 = x_1, \qquad x_1 \cdot 1 = x_1, \\ x_1 \cdot (x_2 \cdot x_3) = (x_1 \cdot x_2) \cdot x_3, \\ x_1 \cdot x_1^{-1} = 1, \qquad x_1^{-1} \cdot x_1 = 1. \end{gathered}$$

We give an example of a formal derivation of $x_2 \cdot 1^{-1} = x_2$ from the axioms of E:

$$\dfrac{\dfrac{x_2 = x_2 \qquad \dfrac{\dfrac{\dfrac{x_1 \cdot 1 = x_1}{1^{-1} \cdot 1 = 1^{-1}}\,(\text{subst})}{1^{-1} = 1^{-1} \cdot 1}\,(\text{symm}) \qquad \dfrac{x_1^{-1} \cdot x_1 = 1}{1^{-1} \cdot 1 = 1}\,(\text{subst})}{1^{-1} = 1}\,(\text{trans})}{x_2 \cdot 1^{-1} = x_2 \cdot 1}\,(\text{cong}) \qquad \dfrac{x_1 \cdot 1 = x_1}{x_2 \cdot 1 = x_2}\,(\text{subst})}{x_2 \cdot 1^{-1} = x_2}\,(\text{trans})$$

□

A Σ-algebra $\mathcal{A}$ *is a model of* E (in symbols: $\mathcal{A} \models E$) iff $\mathcal{A} \models s = t$, for each $s = t \in E$. We say that $s = t$ is a (semantic) equational consequence of E (in symbols: $E \models s =_{\text{eq}} t$) if for every Σ-algebra $\mathcal{A}$:

$$\mathcal{A} \models E \Longrightarrow \mathcal{A} \models s = t.$$

We now prove Birkhoff's completeness theorem for equational theories. Let $=_E$ be the relation on $\text{Ter}(\Sigma)$ defined by

$$s =_E t \Longleftrightarrow_{\text{def}} E \vdash_{\text{eq}} s = t.$$

This relation of *E-provable equality* is an equivalence relation and a congruence with respect to the operations $F^{\text{Ter}(\Sigma)}$, according to the rules of the equational theory. We consider the set $\mathcal{T}(E) = \text{Ter}(\Sigma)/=_E$ of equivalence classes $[t]$ of terms. Thus the following is a well-defined operation

$$F^{\mathcal{T}(E)}([t_1], \ldots, [t_n]) = [F(t_1, \ldots, t_n)]$$

for any $F \in \Sigma$. Thus $\mathcal{T}(E)$ is a Σ-algebra.

Theorem 1.5 (Birkhoff) *Let Σ be a signature and let E be an equational theory over Σ. Then*

$$E \vdash_{\text{eq}} s = t \Longleftrightarrow \mathcal{T}(E) \models s = t.$$

Proof. ($\Leftarrow$) Suppose $\mathcal{T}(E) \models s = t$. Then for the "identical" variable assignment $\tau(x_i) = [x_i]$ we get $[\![s]\!]_\tau^{\mathcal{T}(E)} = [\![t]\!]_\tau^{\mathcal{T}(E)}$. Hence $[s] = [t]$, so $s =_E t$ and thus $E \vdash_{\text{eq}} s = t$.

($\Rightarrow$) Note that each variable assignment $\tau : \mathbb{X} \to \mathcal{T}(E)$ gives rise to a substitution $\sigma : \mathbb{X} \to \text{Ter}(\Sigma)$ where

$$\tau(x_i) = [\sigma(x_i)].$$

Thus from $E \vdash_{\text{eq}} s = t$ and the substitution rule follows $E \vdash_{\text{eq}} s^\sigma = t^\sigma$. Hence $[s^\sigma] = [t^\sigma]$. But $[\![s]\!]_\tau = [s^\sigma]$ and $[\![t]\!]_\tau = [t^\sigma]$, and hence $\mathcal{T}(E) \models s = t$, since τ was arbitrary. □

Corollary 1.6 *For every equational theory E and any equation $s = t$ over Σ we have*

$$E \models_{\text{eq}} s = t \Longleftrightarrow E \vdash_{\text{eq}} s = t$$

Proof. ($\Leftarrow$) This is an easy proof by induction on derivations.

($\Rightarrow$) From Theorem 1.5 ($\Rightarrow$) follows $\mathcal{T}(E) \models E$ (since $s = t \in E$ implies $E \vdash_{\text{eq}} s = t$). Suppose $E \models_{\text{eq}} s = t$. Then in particular $\mathcal{T}(E) \models s = t$. By Theorem 1.5 ($\Leftarrow$) again $E \vdash_{\text{eq}} s = t$. □

Remark 1.7 In view of Birkhoff's completeness theorem and the usual completeness theorem for first order logic, we have for equational theories E:

$$E \vdash_{\text{eq}} s = t \Longleftrightarrow \forall(E) \vdash \forall(s = t).$$

For a formula φ with free variables $x_1, \ldots, x_n$, the expression $\forall(\varphi)$ denotes $\forall x_1 \cdots \forall x_n\, \varphi$. For a set of formulas E, then $\forall(E) = \{\forall(\varphi) : \varphi \in E\}$. *Thus quantifiers and connectives are not necessary when proving an equation from equational axioms.*

Example 1.8 *The equational theory of Abelian groups.* Let $\Sigma = \{1, \cdot, (\,)^{-1}\}$, where the arities are 0, 2 and 1 respectively. The equations E are

$$\begin{aligned}
&1 \cdot x_1 = x_1, \qquad x_1 \cdot 1 = x_1,\\
&x_1 \cdot (x_2 \cdot x_3) = (x_1 \cdot x_2) \cdot x_3,\\
&x_1 \cdot x_2 = x_2 \cdot x_1,\\
&x_1 \cdot x_1^{-1} = 1, \qquad x_1^{-1} \cdot x_1 = 1.
\end{aligned}$$

The models of this theory are exactly the Abelian groups. Denote by $u^0 = 1$ and $u^{n+1} = u \cdot u^n$ for $n \in \mathbb{N}$. For $n > 0$, let $u^{-n} = (u^{-1})^n$. It is easy to show that for each $t \in \text{Ter}(\Sigma)$ there are sequences $n_1, \ldots, n_k \in \mathbb{Z} - \{0\}$, $1 \le i_1 < i_2 \cdots < i_k$, where $k \ge 0$, such that

$$t =_E x_{i_1}^{n_1} \cdot x_{i_2}^{n_2} \cdot \cdots \cdot x_{i_k}^{n_k}. \tag{1}$$

(In case $k = 0$, the product is simply 1.) Thus in the model $\mathcal{T}(E)$ the equivalence classes are represented by elements of the form $x_{i_1}^{n_1} \cdot x_{i_2}^{n_2} \cdot \cdots \cdot x_{i_k}^{n_k}$. □

One can in fact show that the sequences (n_j), (i_j) in (1) are unique. This can be used to decide when two terms are provably equal. A systematic method for obtaining such decidability results is provided by the theory of *term rewriting systems.*

For a signature Σ with at least one constant symbol, consider $\mathcal{T}_0(E)$ which is defined as $\mathcal{T}(E)$ but $\mathrm{Ter}_0(\Sigma)$ is used instead of $\mathrm{Ter}(\Sigma)$. (Exercise: What is $\mathcal{T}_0(E)$ in the case of Example 1.8? If new constants are added?)

Theorem 1.9 *Let E be an equational theory over a signature Σ, which has at least one constant symbol. Then*

(a) $\mathcal{T}_0(E) \models E$

(b) if $\mathcal{A} \models E$, there is a unique homomorphism $\varphi : \mathcal{T}_0(E) \to \mathcal{A}$.

Proof. (a): This is proved as in the direction $(\Rightarrow)$ of Theorem 1.5, but using $\mathcal{T}_0(E)$ instead of $\mathcal{T}(E)$.

(b): Define $\varphi : \mathcal{T}_0(E) \to \mathcal{A}$ by $\varphi([t]) = [\![t]\!]_\tau^{\mathcal{A}}$ where τ is some fixed variable assignment (it does not matter which since t has no variables). It is well-defined because if $[s] = [t]$, then $E \vdash_{\mathrm{eq}} s = t$. Now $\mathcal{A} \models E$, so $\mathcal{A} \models s = t$, and hence in particular $[\![s]\!]_\tau^{\mathcal{A}} = [\![t]\!]_\tau^{\mathcal{A}}$. Furthermore φ is a homomorphism, since

$$\begin{aligned} \varphi(F^{\mathcal{T}(E)}([t_1], \ldots, [t_n])) &= \varphi([F(t_1, \ldots, t_n)]) \\ &= [\![F(t_1, \ldots, t_n)]\!]_\tau^{\mathcal{A}} \\ &= F^{\mathcal{A}}([\![t_1]\!]_\tau^{\mathcal{A}}, \ldots, [\![t_n]\!]_\tau^{\mathcal{A}}) \\ &= F^{\mathcal{A}}(\varphi([t_1]), \ldots, \varphi([t_n])). \end{aligned}$$

Now, if ψ were another homomorphism, it is easily shown that $\psi([t]) = \varphi([t])$ by induction on t. □

Because of this theorem the model $\mathcal{T}_0(E)$ is called the *initial model* of the theory E.

Remark 1.10 For algebraic specification of programs one usually consider Σ-algebras with many sorts (types). For instance, we may have a sort A for an alphabet and a sort S for a stack. The constants are $\mathsf{a}, \mathsf{b}, \mathsf{c} : \mathsf{A}$ (the letters of the alphabet), $\mathsf{nil} : \mathsf{S}$ (the empty stack), the function symbols are $\mathsf{pop} : \mathsf{S} \to \mathsf{S}$ and $\mathsf{push} : \mathsf{A} \times \mathsf{S} \to \mathsf{S}$. The equations E are

$$\begin{aligned} \mathsf{pop}(\mathsf{nil}) &= \mathsf{nil}, \\ \mathsf{pop}(\mathsf{push}(x^{\mathsf{A}}, t^{\mathsf{S}})) &= t^{\mathsf{S}} \end{aligned}$$

(Here $x^{\mathsf{A}}, t^{\mathsf{S}}$ indicate variables of the different sorts.) The definitions and results above easily extend to many-sorted Σ-algebras.

Exercises

1. Let A^* be set of strings over the alfabet A. Describe this set as an algebra with a binary concatenation operator and an empty set. Let B be another alfabet. Show that each function $f : A \to B^*$ extends to a homomorphism $\varphi : A^* \to B^*$. (Hint: Letter for string substitution.)

2. The equational theory of semigroups is given by E_1:

$$1 \cdot x_1 = x_1, \qquad x_1 \cdot 1 = x_1,$$
$$x_1 \cdot (x_2 \cdot x_3) = (x_1 \cdot x_2) \cdot x_3,$$

   where $\Sigma = \{1, \cdot\}$. Determine the equivalence classes in $\mathcal{T}(E_1)$ analogously to Example 1.8.

3. Try to find simple representatives of equivalence classes in $\mathcal{T}_0(E)$ where E is as in Remark 1.10.

4. Restricting the equational logic and putting more requirements on the axioms suggests a proof search strategy. Call an equational theory E over Σ *instantiation closed* if

   (a) $s = t \in E$ implies $s^\sigma = t^\sigma \in E$ for each substitution $\sigma : \mathbb{X} \to \mathrm{Ter}(\Sigma)$.

   (b) $s = t \in E$ implies $t = s \in E$.

   Let $\vdash_{\mathrm{r}}$ denote the derivation relation which is as $\vdash_{\mathrm{eq}}$ but where derivations are restricted to using only (ax.appl.), (cong), (refl) and (trans). Let $\vdash_{\mathrm{d}}$ be the further restriction that (trans) is disallowed.

   Consider an instantiation closed theory E.

   (i) Prove by induction on the height of proofs that for all terms s, t

$$E \vdash_{\mathrm{r}} s = t \Longrightarrow E \vdash_{\mathrm{r}} t = s.$$

   (ii) Prove that for all terms s, t and all substitutions σ

$$E \vdash_{\mathrm{r}} s = t \Longrightarrow E \vdash_{\mathrm{r}} s^\sigma = t^\sigma.$$

   (iii) Conclude that

$$E \vdash_{\mathrm{eq}} s = t \Longleftrightarrow E \vdash_{\mathrm{r}} s = t.$$

(iv)* Define $E \vdash_{\mathrm{d}}^{n} s = t$ iff there are terms $s_1, \ldots, s_{n-1}$ such that $s_1 \equiv s$

$$E \vdash_{\mathrm{d}} s_1 = s_2 \quad E \vdash_{\mathrm{d}} s_2 = s_3 \quad \cdots \quad E \vdash_{\mathrm{d}} s_{n-1} = t.$$

Prove that

$$E \vdash_{\mathrm{eq}} s = t \iff \text{for some } n \geq 1\text{: } E \vdash_{\mathrm{d}}^{n} s = t.$$

Hint: transform the proofs so that transitivity applications appear at the end. Use transformations of the following kind, where $\mathcal{D}, \mathcal{D}', \mathcal{D}''$, are proof trees.

$$\dfrac{\dfrac{\mathcal{D}}{s = t} \quad \dfrac{\dfrac{\mathcal{D}'}{a = b} \quad \dfrac{\mathcal{D}''}{b = c}}{a = c}\,(\text{trans})}{f(s,a) = f(t,c)}\,(\text{cong})$$

$$\Downarrow$$

$$\dfrac{\dfrac{\dfrac{\mathcal{D}}{s = t} \quad \dfrac{\mathcal{D}'}{a = b}}{f(s,a) = f(t,b)}\,(\text{cong}) \quad \dfrac{\dfrac{}{t = t}\,(\text{refl}) \quad \dfrac{\mathcal{D}''}{b = c}}{f(t,b) = f(t,c)}\,(\text{con})}{f(s,a) = f(t,c)}\,(\text{trans})$$

1.2 Unification of terms

Unification is an important tool in term rewriting, automatic theorem proving, and is fundamental for logic programming (Prolog). Unification of terms amount to equation solving in the term algebra $\mathrm{Ter}(\Sigma)$.

Example 1.11 Let $\Sigma = \{f, g\}$ with arities 2 and 1 respectively. Find a solution in $\mathrm{Ter}(\Sigma)$ to the equation

$$f(x_1, g(f(x_2, x_1))) = f(g(x_2), x_3).$$

A solution: $x_1 := g(x_2), x_3 := g(f(x_2, g(x_2)))$.

As in ordinary equation solving we are often interested in a general solution. Over the term algebra such a solution is called a *most general unifier*. Indeed, in the example above any other solution can be gotten from the one provided, by instantiating the variables.

As explained in Section 1.1 substitutions can be regarded as Σ-algebra homomorphisms $\sigma : \mathrm{Ter}(\Sigma) \to \mathrm{Ter}(\Sigma)$ determined by their values on the set $\mathbb{X}$ of variables. A substitution that is given by a permutation of the variables is called a *renaming substitution*. Two substitutions $\tau : \mathrm{Ter}(\Sigma) \to \mathrm{Ter}(\Sigma)$ and $\sigma : \mathrm{Ter}(\Sigma) \to \mathrm{Ter}(\Sigma)$ may be composed $\sigma \circ \tau$ as follows

$$(\sigma \circ \tau)(t) = \sigma(\tau(t)) = (t^{\tau})^{\sigma}.$$

We write $\tau\sigma$ for $\sigma \circ \tau$.

Example 1.12 Let $\Sigma = \{f, g\}$ with arities 2 and 1 respectively. Consider the substitutions $\sigma = \{x_2 := g(x_1), x_3 := g(x_3)\}$ and $\tau = \{x_1 := f(x_2, x_2)\}$. Then $(\tau\sigma)(x_1) = \sigma(\tau(x_1)) = \sigma(f(x_2, x_2)) = f(\sigma(x_2), \sigma(x_2)) = f(g(x_1), g(x_1))$, $(\tau\sigma)(x_2) = g(x_1)$ and $(\tau\sigma)(x_3) = g(x_3)$. Hence

$$\tau\sigma = \{x_1 := f(g(x_1), g(x_1)), x_2 := g(x_1), x_3 := g(x_3)\}.$$

On the other hand, by a similar computation,

$$\sigma\tau = \{x_1 := f(x_2, x_2), x_2 := g(f(x_2, x_2)), x_3 := g(x_3)\}. \square$$

Generalising this example we have for $\sigma = \{x_{i_1} := t_1, \ldots, x_{i_n} := t_n\}$ and $\tau = \{x_{i_1} := s_1, \ldots, x_{i_n} := s_n, x_{j_1} := r_1, \ldots, x_{j_m} := r_m\}$, where the indices $i_1, \ldots, i_n, j_1, \ldots, j_m$ are all distinct, that

$$\sigma\tau = \{x_{i_1} := t_1^{\tau}, \ldots, x_{i_n} := t_n^{\tau}, x_{j_1} := r_1, \ldots, x_{j_m} := r_m\}$$

We say that one substitution σ is *more general* than another substitution ρ iff $\rho = \sigma\tau$ for some substitution τ. In this case we write $\sigma \leq \rho$.

Exercise 1.13

(i) Check that the relation $\leq$ is reflexive and transtive.

(ii) Prove that if $\sigma \leq \rho$ and $\rho \leq \sigma$, then there is a renaming substitution τ such that $\rho = \sigma\tau$. $\square$

A *unifier* of a set of terms $\mathcal{T} = \{t_1, \ldots, t_n\}$ is substitution σ which makes all these terms equal, i.e. $t_1^{\sigma} = \cdots = t_n^{\sigma}$. A unifier σ of $\mathcal{T}$ is a *most general unifier (mgu)*, if $\sigma \leq \rho$ for any unifier ρ of $\mathcal{T}$. By Exercise 1.13 any two mgu's σ and σ' of $\mathcal{T}$ are the same up to a renaming substitution (i.e. $\sigma = \sigma'\tau$ for some renaming substitution τ).

Note that $F(s_1, \ldots, s_n)^{\sigma} = F(t_1, \ldots, t_n)^{\sigma}$ iff $s_i^{\sigma} = t_i^{\sigma}$ for all $i = 1, \ldots, n$. Hence in order to solve one equation in the term algebra, we may have to solve a system of equations.

The unification algorithm of Martelli-Montanari. The algorithm starts with a finite set of equations $G = \{s_1 = t_1, \ldots, s_n = t_n\}$, and outputs a most general unifier σ for this set (regarded as an mgu of the set $\{F(s_1, \ldots, s_n), F(t_1, \ldots, t_n)\}$ where F is a function symbol), if there is any unifier, or reports failure otherwise. The algorithm is non-deterministic and applies certain reduction rules to the finite sets and stops at the empty set ($\emptyset$), or with a failure (denoted #). Along the way the answer substitution σ is built up. From a successful computation

$$G_1 \rightarrowtail G_2 \rightarrowtail_{\sigma_1} G_3 \rightarrowtail G_4 \rightarrowtail G_5 \rightarrowtail_{\sigma_2} G_6 \rightarrowtail \emptyset.$$

we extract $\sigma = \sigma_1\sigma_2$, the answer substitution. For a set $G = \{s_1 = t_1, \ldots, s_n = t_n\}$ we write $G^\sigma = \{s_1^\sigma = t_1^\sigma, \ldots, s_n^\sigma = t_n^\sigma\}$.

The Martelli-Montanari reduction rules are the following

1. $G \cup \{F(t_1, \ldots, t_n) = F(s_1, \ldots, s_n)\} \rightarrowtail G \cup \{t_1 = s_1, \ldots, t_n = s_n\}$ provided $F(t_1, \ldots, t_n) = F(s_1, \ldots, s_n)$ is not an element of G. (“Function decomposition”)

2. $G \cup \{t = t\} \rightarrowtail G$ provided $t = t$ is not an element of G.

3. $G \cup \{t = x\} \rightarrowtail G \cup \{x = t\}$, provided t is not a variable, and that $t = x$ is not an element of G.

4. $G \cup \{x = t\} \rightarrowtail_{\{x := t\}} G^{\{x := t\}}$, provided x is a variable, x does not occur in t and that $x = t$ is not an element of G. (“Variable elimination”)

5. $G \cup \{F(t_1, \ldots, t_n) = H(s_1, \ldots, s_m)\} \rightarrowtail$ #, if F and H are different function symbols.

6. $G \cup \{x = t\} \rightarrowtail$ #, provided $x \neq t$ and x occurs in t. (“Occur check”)

Example 1.14 We compute the mgu of $f(x_1, g(f(x_2, x_1)))$ and $f(g(x_2), x_3)$ using the algorithm.

$$\begin{array}{lll}
\{f(x_1, g(f(x_2, x_1))) = f(g(x_2), x_3)\} & \rightarrowtail & \{x_1 = g(x_2), g(f(x_2, x_1)) = x_3\} \\
 & \rightarrowtail_{\{x_1 := g(x_2)\}} & \{g(f(x_2, g(x_2))) = x_3\} \\
 & \rightarrowtail & \{x_3 = g(f(x_2, g(x_2)))\} \\
 & \rightarrowtail_{\{x_3 := g(f(x_2, g(x_2)))\}} & \emptyset
\end{array}$$

The answer substitution is $\sigma = \{x_1 := g(x_2), x_3 := g(f(x_2, g(x_2)))\}$. □

Example 1.15 The terms $f(g(x_1),x_1)$ and $f(x_2,g(x_2))$ are not unifiable.

$$\begin{array}{lll} \{f(g(x_1),x_1) = f(x_2,g(x_2))\} & \rightarrowtail & \{g(x_1) = x_2, x_1 = g(x_2)\} \\ & \rightarrowtail_{\{x_1:=g(x_2)\}} & \{g(g(x_2)) = x_2\} \\ & \rightarrowtail & \{x_2 = g(g(x_2))\} \\ & \rightarrowtail & \# \end{array}$$

This computation fails by occur check, since x_2 occurs in $g(g(x_2))$. □

We state the following important result without proof:

Theorem 1.16 (Unification Theorem) *A set of equations $G = \{s_1 = t_1, \ldots, s_n = t_n\}$ has an mgu iff it has some unifier. Moreover, if G has an mgu, the Martelli-Montanari algorithm finds it, otherwise it stops and reports failure to find a unifier.*

1.2.1 Pattern matching

Pattern matching may be regarded as a special case of unification: a *variable free* term s is matched to the *pattern term t* if there is a unifier σ with

$$s = s^\sigma = t^\sigma.$$

For a term s containing variables, we may first replace each variable x with a new constant c_x, and then match the modified term s^* to t. The new constants occuring in the resulting unifier may then be restored to variables again.

Example 1.17 The term $f(0,g(2))$ is matched to $f(u,v)$ by $\sigma = \{u := 0, v := g(2)\}$.

The term $f(x,g(u))$ is matched to $f(u,v)$ by $\tau = \{u := x, v := g(u)\}$. The intermediate step is to consider the variable free term $f(\mathrm{c}_x, g(\mathrm{c}_u))$, and the unifier $\tau^* = \{u := \mathrm{c}_x, v := g(\mathrm{c}_u)\}$. Note that τ is *not* a unifier of $f(x,g(u))$ and $f(u,v)$.

However $f(x,g(u))$ cannot be matched to $f(g(u),v)$ since c_x and $g(u)$ are not unifiable.

Exercises

1. Let $\Sigma = \{a, f, g, h, p, q\}$ where a is a constant, f, g has arity 1, h, p has arity 2 and q has arity 3. For each of the following pair of terms compute an mgu or show that no unifier exist.

   (a) $p(f(a),g(x))$, $p(y,y)$

   (b) $p(f(x),a)$, $p(y,f(w))$

   (c) $p(x,x)$, $p(y,f(y))$

(d) $q(a, x, f(g(y))), q(z, h(z, w), f(w))$

(e) $p(f(f(x)), h(g(x), f(a))), p(f(u), h(v, f(w)))$.

2. In which cases (a) – (e) in Exercise 1 does the first term match the pattern of the second term?

2 Term rewriting systems

We shall be very brief and sketchy on this subject. We refer to Baader and Nipkow (1999) or to Klop (1992) for a full account of the basic theory.

A term rewriting system (TRS) is essentially a way of assigning directions to the equations of an equational theory E, and then applying the equations only in the prescribed directions. In some circumstances a TRS can be deviced for E so that it can be decided whether $E \vdash_{\mathrm{eq}} s = t$ holds by making "mindless" applications of the directed equations to the terms s and t respectively and when no further applications are possible check whether the end results are the same. We shall explain what "mindless application" means below.

First we give an example of a TRS. Consider the equational theory of semigroups (Exercise 1.2). The language is $\Sigma = \{1, \cdot\}$. The equations

$$\begin{aligned} 1 \cdot x &= x \\ x \cdot 1 &= x \\ x \cdot (y \cdot z) &= (x \cdot y) \cdot z \end{aligned}$$

may be given the natural directions

$$\begin{aligned} 1 \cdot x &\to x \\ x \cdot 1 &\to x \\ x \cdot (y \cdot z) &\to (x \cdot y) \cdot z \end{aligned}$$

These are called *rewrite rules.*

Consider the following applications of the directed equations, so called *reductions*. We underline the subterms to which the rewrite rules have been applied

$$\underline{1 \cdot (x_2 \cdot x_3)} \xrightarrow{(3)} \underline{(1 \cdot x_2)} \cdot x_3 \xrightarrow{(1)} x_2 \cdot x_3$$

$$\underline{1 \cdot (x_2 \cdot x_3)} \xrightarrow{(1)} x_2 \cdot x_3.$$

$$\underline{(x_2 \cdot 1)} \cdot x_3 \xrightarrow{(2)} x_2 \cdot x_3.$$

It is not possible to apply any further rules to $x_2 \cdot x_3$ since x_2 and x_3 are variables and stand for arbitrary objects.

Let Σ be an arbitrary signature. A *rewrite rule* over Σ is a pair $(s,t) \in \mathrm{Ter}(\Sigma)^2$, written $s \to t$, so that s is not a variable and $FV(t) \subseteq FV(s)$. A *term rewriting system* over Σ is a finite set of rewrite rules over Σ.

Example 2.1 $\Sigma_1 = \{1, \cdot\}$ and $R_1 = \{1 \cdot x_1 \to x_1, x_1 \cdot 1 \to x_1, x_1 \cdot (x_2 \cdot x_3) \to (x_1 \cdot x_2) \cdot x_3\}$ is the TRS for semigroups above.

We allow variable names x, y, z, u, v, w as well as the official variables $x_1, x_2, x_3, \ldots$ of the set $\mathbb{X}$.

Example 2.2 A term rewriting system for addition and multiplication: $\Sigma_2 = \{0, +, \cdot, \mathsf{s}\}$ and R_2 consists of the following rules

$$\begin{array}{rcl} x+0 & \to & x \\ x+\mathsf{s}(y) & \to & \mathsf{s}(x+y) \\ x \cdot 0 & \to & 0 \\ x \cdot \mathsf{s}(y) & \to & x \cdot y + x \end{array}$$

Let R be a TRS over Σ. For two terms $t_1, t_2 \in \mathrm{Ter}(\Sigma)$ we say that t_2 *has been obtained by one-step reduction from* t_1 *using* R, in symbols

$$t_1 \to_R t_2$$

if $t_1 = C^{\{z := s^\sigma\}}$, $t_2 = C^{\{z := t^\sigma]}$ for some rule $(s, t) \in R$, some variable z, some substitution σ and some term C with exactly one occurence of z.

We write $\to^*$ for the reflexive and transitive closure of $\to$. We say that s *reduces to* t if $s \to_R^* t$.

We call a term $t \in \mathrm{Ter}(\Sigma)$ is called *normal* with respect to R if there is no term s such that $t \to_R s$.

Note that variables are normal with respect to any TRS.

Example 2.3 For TRS R_1: 1 is normal. Each expression of the form

$$(\cdots((x_{i_1} \cdot x_{i_2}) \cdot x_{i_3}) \cdot \cdots \cdot x_{i_n})$$

where $n \geq 1$, is normal. In fact these are all the normal terms.

Example 2.4 For TRS R_2: The numerals $0, \mathsf{s}(0), \mathsf{s}(\mathsf{s}(0)), \ldots$ are normal. $x_1 + x_2$ and $x_1 \cdot x_2$ are normal. (Exercise: determine all normal terms)

A TRS R is said to be *confluent* if for any terms r, s_1, s_2 with $r \to^* s_1$ and $r \to^* s_2$ there is a term t such that $s_1 \to^* t$ and $s_2 \to^* t$.

We note that if a term s is normal and $s \to^* t$, then $s = t$. Using this observation have the following simple but important result.

Lemma 2.5 *If R is a confluent TRS, then normal forms are unique if they exist, i.e. for any term r and normal terms s_1, s_2 with $r \to^* s_1$ and $r \to^* s_2$, it holds that $s_1 = s_2$.*

A TRS R is *weakly normalising* if any term reduces to a normal term. It is *strongly normalising* there are is no infinite sequence of terms such that

$$t_1 \to t_2 \to \cdots \to t_n \to \cdots$$

For a strongly normalising TRS any sequence of choices of subterms and applicable rules will thus eventually lead to a normal term. E.g. any "mindless" application of the directed equation will give the result.

Moreover, this means that any strongly normalising TRS is weakly normalising.

Example 2.6 Let $\Sigma_3 = \{0,1,2,3\}$. Consider TRSs given by the following rules over Σ_3.

$R_{3,1} = \{1 \to 0, 1 \to 2\}$ is strongly normalising but not confluent.

$R_{3,2} = \{1 \to 0, 1 \to 2, 2 \to 1, 2 \to 3\}$ is weakly normalising but not strongly normalising.

$R_{3,3} = \{1 \to 0, 1 \to 2, 2 \to 1\}$ is weakly normalising and confluent.

$R_{3,4} = \{1 \to 0, 1 \to 2, 0 \to 0, 2 \to 2\}$ is neither weakly normalising nor confluent.

For a TRS R let $=_R$ be the reflexive, symmetric and transitive closure of the one-step rewrite relation $\to_R$. Thus $s =_R t$ if and only if s can be gotten from t by a series of one-step reductions possibly applying certain of them backwards.

A TRS is *complete* if it is confluent and strongly normalising. The importance of complete TRSs is given by the following result.

Theorem 2.7 *Let R be a complete TRS. Then the relation $s =_R t$ is decidable.*

Proof. We note that by the confluence property $s =_R t$ is equivalent to the existence of some r with $s \to_R^* r$ and $t \to_R^* r$. To decide the equality $s =_R t$ we compute normal forms s', t' with $s \to_R^* s'$ and $t \to_R^* t'$ using the strong normalisation property and that R is finite. If $s' = t'$ then equality holds. Suppose that $s' \neq t'$ but that $s =_R t$ still holds. Then for some r with $s \to_R^* r$ and $t \to_R^* r$. Using confluence we find s'' and t'' with

$$\begin{array}{ccc} s' & \to_R^* & s'' \\ r & \to_R^* & s'' \\ r & \to_R^* & t'' \\ t' & \to_R^* & t'' \end{array}$$

Now since s' and t' are normal we have $s' = s''$ and $t' = t''$ by confluence. But this means that r reduces to two different normal forms, which is impossible. Hence actually, $s =_R t$ is false. □

Let E be an equational theory over Σ. Let R be a TRS over Σ. We say that R is a *TRS for E* if for all s,t

$$E \vdash_{\mathrm{eq}} s = t \Longleftrightarrow s =_R t.$$

A main result is now:

Corollary 2.8 *Suppose E be an equational theory over Σ. Let R be a complete* TRS for E. *Then the provability relation*

$$E \vdash_{\mathrm{eq}} s = t$$

is decidable.

Proof. This follows since $E \vdash_{\mathrm{eq}} s = t$ is equivalent to $s =_R t$, which is decidable. □

A complete TRS does not always exists for a given equational theory E. However one can sometimes use the *Knuth-Bendix completion* procedure to obtain a complete TRS.

We say that a wellfounded partial order $<$ on $\mathrm{Ter}(\Sigma)$ is a *reduction order* if

(a) $s^\sigma < t^\sigma$ whenever $s < t$ and σ is a substitution,

(b) $C^{\{z:=s\}} < C^{\{z:=t\}}$ whenever $s < t$ and C is a term with exactly one occurence of z.

The Knuth-Bendix completion procedure (see Klop 1992) takes E an equational theory over Σ and a reduction order $<$ on $\mathrm{Ter}(\Sigma)$. It may then produce a complete TRS for E, or report that that it is not possible to orient the equations of E, or it may go on forever searching for a TRS.

A main obstacle for the existence of complete TRS of a given equational theory is the presence of commutativity axioms like

$$x \cdot y = y \cdot x.$$

They tend to be impossible to orient.

References

F. Baader and T. Nipkow. *Term Rewriting and All That*. Cambridge University Press 1999.

N. Dershowitz and J.P. Jouannaud. Rewrite Systems. In: J. van Leeuwen (ed.) *Handbook of Theoretical Computer Science.* North-Holland 1990.

J.A. Goguen and G. Malcolm. *Algebraic Semantics of Imperative Programming Languages.* MIT Press, 1996.

W. Klop: Term Rewriting Systems. In: S. Abramsky *et al.* (eds.) *Handbook of Logic in Computer Science,* Vol 2. Oxford University Press 1992.

K. Meinke and J.V. Tucker. Universal Algebra. In: S. Abramsky *et al.* (eds.): *Handbook of Logic in Computer Science, Vol. 1.* Oxford University Press 1992.

W. Wechler. *Universal Algebra for Computer Scientists.* Springer 1992.