ICOT Technical Report: TR-0752

TR-0752

Complete Equational Unification Based on an Extension of the Knuth-Bendix Completion Procedure

by
A. Ohsuga (Toshiba) & K. Sakai

March, 1992

ICOT
Mita Kokusai Bldg. 21F
4-28 Mita 1-Chome
(03)3456-3191～5
Telex ICOT J32964
Minato-ku Tokyo 108 Japan

Institute for New Generation Computer Technology

Complete Equational Unification Based on an Extension of the Knuth-Bendix Completion Procedure

Akihiko Ohsuga
Toshiba Corporation, 70 Yanagicho, Saiwai-ku, Kawasaki-shi 210, Japan

and

Kô Sakai
ICOT Research Center, 1-4-28 Mita, Minato-ku, Tokyo 108, Japan

ABSTRACT

A unifier is a substitution that makes two terms syntactically equal. In this paper, we discuss a more semantical unifier: an equational unifier, which is a substitution that makes two terms equal modulo a congruence relation. As a result we will give a general procedure that enumerates a complete set of equational unifiers for a given pair of terms under a given congruence.

1. Introduction

We assume the reader has elementary knowledge on universal algebra, in particular, on term rewriting systems (see [Huet 80], for example).

Let $\simeq$ be a congruence relation on terms and s, t be terms. A substitution θ is called a **$\simeq$-unifier** of s and t if $s\theta \simeq t\theta$. The set of all $\simeq$-unifiers of s and t are denoted $U(s,t)$. Let $V(s,t)$ be the set of all the variables occurring in s or t. Since we are only interested in substitutions for their effect on s and t in this paper, we regard two unifiers as identical if they differ only on the variables not occurring in s or t, so that we can avoid the subtle treatment of the domains of substitutions. Accordingly, we relativize all notions on unifiers to $V(s,t)$. For example, a unifier θ is said to be **more general** than another unifier ϕ under $\simeq$ (denoted $\theta \leq \phi$) if there is a substitution ψ such that $(v\theta)\psi \simeq v\phi$ for any variable $v \in V(s,t)$. A subset C of $U(s,t)$ is said to be **complete** if, for any $\theta \in U(s,t)$, there exists a unifier $\theta' \in C$ such that $\theta' \leq \theta$. Moreover, a complete subset C is called the **minimum** if $\theta = \theta'$ for any $\theta, \theta' \in C$ such that $\theta \leq \theta'$. We write $\theta \approx \theta'$ if $\theta \leq \theta'$ and $\theta' \leq \theta$. Then, relation $\approx$ is an equivalence on unifiers and, if the minimum complete set exists, it is unique up to $\approx$ [Fages 86].

In the case of ordinary unification (or, more precisely, in the case that $\simeq$ is the identity relation), unifiability is decidable and the most general unifier always exists for any unifiable pair of terms [Robinson 65]. For a general congruence $\simeq$, however, the

existence of the most general unifier is not guaranteed. In this situation, a complete set of $\simeq$-unifiers plays the role that the most general unifier plays in ordinary unification: a representative of all unifiers.

We call a pair of terms an **equation**. As well-known, a set of equations presents a congruence relation. Let $\Re$ be a set of equations. First of all, we give a definition of the congruence presented by $\Re$ in terms of reduction. A term u is **reduced** to another term u' by $\Re$ (denoted by $u \Rightarrow u'$) if there is an equation $\langle l, r \rangle \in \Re$, a context $c[]$, and a substitution θ such that $c[l\theta] = u$ and $c[r\theta] = u'$. In other words, if a term has a subterm matched with the left hand side of an equation, then it is reduced to the term obtained by replacing the subterm with the right hand side. We write $u \Leftrightarrow u'$ if $u \Rightarrow u'$ or $u' \Rightarrow u$. Then, the **congruence** presented by $\Re$ is defined as the reflexive transitive closure of relation $\Leftrightarrow$.

In what follows, we assume that a finite presentation of congruence $\simeq$ is given. We call problems concerning $\simeq$-unifiers for such an congruence $\simeq$ equational unification problems. The main equational unification problems are the following.

(1) Is $\simeq$-unifiability decidable?

(2) Does the minimum complete set of $\simeq$-unifiers exist? Can it be enumerated?

(3) Is there a finite complete set of $\simeq$-unifiers?

(4) Is there an efficient procedure to enumerate a complete set of $\simeq$-unifiers?

It is undecidable in general whether two given terms have a $\simeq$-unifier. As for the answers to these problems on specific sets of equations, there is a wide-ranged survey by Siekmann [Siekmann 89]. The result on AC-unification, in which the equation set consists only of the associative and the commutative laws, seems to be the most important from a practical point of view. That is, the minimum complete set of AC-unifiers always exists and it is finite and computable [Stickel 81][Fages 84][Huet 78].

In this paper, we address problem (4) for a general set of equations. A procedure is said to be a **complete equational unification procedure** if it enumerates a complete set of $\simeq$-unifiers of given terms s and t. It is clear from the definition that, for any terms s and t, the set $U(s,t)$ is recursively enumerable and complete. Therefore, enumeration of $U(s,t)$ is a complete (but not interesting) equational unification procedure. What is interesting is a more efficient procedure than simple enumeration of all unifiers.

From a theoretical point of view, the minimum complete set may be the most interesting since it is unique and not redundant. However, there is no reality of computation of the minimum complete set for the following reasons. First of all, the minimum set may not exist. That is, there may be a complete set C of $\simeq$-unifiers with the following property: for any $\theta \in C$, there is a unifier $\theta' \in C$ such that $\theta' < \theta$ (that is, $\theta' \leq \theta$, but $\theta \not\leq \theta'$) [Fages 86]. Even if the minimum set exists, there may be no procedures to enumerate its elements. Even if it is enumerable, it may need more cost to compute than other (redundant) complete sets.

Several researchers proposed equational unification procedures based on narrowing [Fay 79] and basic narrowing [Hullot 80][Bosco 87], under the assumption that the given set of equations (viewed as left-to-right rewrite rules) is confluent and terminating. These are efficient, but the assumption is seldom satisfied in actual cases. Gallier and Snyder proposed a universal equational unification procedure [Gallier 87], but it does not seem efficient enough for actual applications.

We propose another general procedure and prove its completeness. We confirmed that it has little redundancy (and, therefore, hopefully efficient) in many cases by actual implementation and experiment using simple but real mathematical problems. We do not discuss the implementation details in this paper, but Examples 4.1 and 4.2 show some material of experiment.

The procedure is based on a combination of the Knuth-Bendix completion [Knuth 70][Huet 81] (or, more precisely, completion without failure [Bachmair 87]) and narrowing. The procedure applies narrowing to s and t, while constructing a (possibly infinite) confluent and terminating set of equations (viewed as rewrite rules). Since, as shown in [Huet 81] and [Bachmair 87], a confluent and terminating set can be obtained virtually even if the completion process does not terminate, the narrowing process eventually enumerates a complete set of unifiers. Moreover, since the procedure is an extension of the Knuth-Bendix completion, it may obtain a finite confluent and terminating set on the way of equational unification. Once such a set is obtained, the subsequent process becomes ordinary narrowing. Therefore, Fay's result [Fay 79] is viewed as a special case.

The essential idea is common with the refutational theorem proving in first-order logic with equality proposed by Hsiang and Rusinowitch [Hsiang 87][Rusinowitch 88]. The purpose of this paper is not to claim originality of the idea but to claim its naturality and effectiveness and to give a proof of its completeness from the viewpoint of equational unification.

2. Inference rules for equational unification

In the following discussion, let $\preceq$ be a fixed strong simplification order on terms, namely, a simplification order [Dershowitz 82] which is total on ground terms. We use the lexicographic subterm ordering [Sakai 84] in the examples as such an order. In the following definitions, we assume that given terms and equations do not have common variables for simplicity of discussion.

First we change the concept of reduction by set of equations. Usually, as defined in the previous section, when an equation is viewed as a rewrite rule, it is assumed to be used from left to right only. However, we do not assume this any longer, that is, an equation is used as a rewrite rule in both directions. Instead, we control the direction of rewriting by order $\preceq$. For this definition of reduction, it is simpler to consider an equation as an unordered pairs of terms. Therefore, from now on, we regard equations $\langle l, r\rangle$ and $\langle r, l\rangle$ as the same.

To be precise, the definition of reduction is the following: a term u is **reduced** to another term u' (denoted $u \Rightarrow u'$) if $u' \prec u$ and there is an equation $\langle l, r\rangle$ (or its

equivalent $\langle r, l\rangle$), a context $c[]$, and a substitution σ such that $c[l\sigma] = u$ and $c[r\sigma] = u'$. Let us denote the reflexive transitive symmetric closure of $\Rightarrow$ by $\simeq'$. It is a routine to verify that $\simeq'$ is a congruence relation.

This above reduction has somewhat different properties from the ordinary left-to-right reduction. First, since $\preceq$ is well-founded [Dershowitz 82][Sakai 84], it is always terminating. Second, congruence $\simeq'$ may be weaker than the congruence relation $\simeq$ presented by the given set of equation. For example, let us consider equation set $\{\langle x+y, y+x\rangle\}$. Then, reduction $\Rightarrow$ is not terminating in the ordinary sense (defined in the previous section) since $u+u' \Rightarrow u'+u \Rightarrow u+u' \Rightarrow u'+u \Rightarrow \cdots$, and $v+w \not\simeq' w+v$ for any variables v and w since neither $v+w \prec w+v$ nor $w+v \prec v+w$.

The word problem involves the decision of not $\simeq'$ but $\simeq$. However, in many cases, we can assume that terms are ground without loss of generality by substituting fresh constants to variables in the terms. For ground terms, the symmetric closures $\Leftrightarrow$ coincide in both definitions of reduction since $\preceq$ is total and, therefore, so do congruences $\simeq$ and $\simeq'$.

Next, we define narrowing, which is somewhat modified of that by Fay [Fay 79] as well. A term u is said to be **narrowed** to another term u' (denoted $u \hookrightarrow u'$) if there are a non-variable subterm u_0 of u, an equation $\langle l, r\rangle$ such that $u_0\theta = l\theta$ and $u\theta \not\preceq u' = c[r]\theta$, where $u = c[u_0]$ and θ is the most general unifier of u_0 and l. If necessary, we suffix the most general unifier, for example, as $u \hookrightarrow_\theta u'$. In what follows, we discuss narrowing of a pair of terms. A pair is narrowed to another pair if one of the terms is narrowed. To be precise, notation $\langle u_1, u_2\rangle \hookrightarrow_\theta \langle u_1', u_2'\rangle$ means that either $u_1 \hookrightarrow_\theta u_1'$ and $u_2\theta = u_2'$, or $u_1\theta = u_1'$ and $u_2 \hookrightarrow_\theta u_2'$.

Let us extend the definition of critical pairs [Knuth 70] as well. Let $\langle l_1, r_1\rangle$ and $\langle l_2, r_2\rangle$ be equations and u be a non-variable subterm of l_2 unifiable with l_1. If $l_1\theta \not\preceq r_1\theta$ and $l_2\theta \not\preceq r_2\theta$, then pair $\langle c[r_1]\theta, r_2\theta\rangle$ is called a **critical pair**, where $l_2 = c[u]$ and θ is the most general unifier of l_1 and u.

Now, we give a universal equational unification procedure, which inputs a set $\Re$ of equations and terms s, t and outputs $\simeq$-unifier of s and t for the congruence $\simeq$ presented by $\Re$. It is given below in the form of inference rules.

E-generation: $\dfrac{(E, R, G, U)}{(E \cup \{\langle u_1, u_2\rangle\}, R, G, U)}$ $\langle u_1, u_2\rangle$ is a critical pair between equations in R

E-reduction: $\dfrac{(E \cup \{\langle u_1, u_2\rangle\}, R, G, U)}{(E \cup \{\langle u_1, u_2'\rangle\}, R, G, U)}$ $u_2 \Rightarrow u_2'$ by an equation in R

E-deletion: $\dfrac{(E \cup \{\langle u, u\rangle\}, R, G, U)}{(E, R, G, U)}$

R-generation: $\dfrac{(E \cup \{\langle u_1, u_2\rangle\}, R, G, U)}{(E, R \cup \{\langle u_1, u_2\rangle\}, G, U)}$

G-generation: $\dfrac{(E, R, G \cup \{\langle u_1, u_2, \theta\rangle\}, U)}{(E, R, G \cup \{\langle u_1, u_2, \theta\rangle, \langle u_1', u_2', \theta \circ \theta'\rangle\}, U)}$

U-generation:
$$\frac{(E, R, G \cup \{\langle u_1, u_2, \theta\rangle\}, U)}{(E, R, G \cup \{\langle u_1, u_2, \theta\rangle\}, U \cup \{\theta \circ \theta'\})}$$
$\langle u_1, u_2\rangle \hookrightarrow_{\theta'} \langle u_1', u_2'\rangle$ by an equation in R

θ' is the most general unifier of u_1 and u_2

Each inference rule expresses operation to transform the quadraple above the horizental line to the quadraple below. Both E and R are sets of equations. G is a set of triples $\langle u_1, u_2, \theta\rangle$ (called **goals**) where u_1 and u_2 are terms and θ is a substitution. We regare triples $\langle u_1, u_2, \theta\rangle$ and $\langle u_2, u_1, \theta\rangle$ as the same similarly to equations. U is a set of substitutions. At the beginning of the procedure, these are set as follows:

$$E = \mathcal{R}, \quad R = \emptyset, \quad G = \{\langle s, t, \epsilon\rangle\}, \quad U = \emptyset,$$

where ϵ denotes the identity substitution. The procedure enumerates $\simeq$-unifiers of s and t as elements of U.

When one of the inference rules is applied, a quadruple (E, R, G, U) is transformed to another quadruple (E', R', G', U'), denoted by $(E, R, G, U) \vdash (E', R', G', U')$. If necessary, the name of the applied inference rule are suffixed to symbol $\vdash$. Let

$$(E_0, R_0, G_0, U_0) \vdash (E_1, R_1, G_1, U_1) \vdash (E_2, R_2, G_2, U_2) \vdash \cdots$$

be a sequence of applications of the inference rules. We denote $\bigcup_{i=0}^{\infty} E_i$ by E_∞, $\bigcup_{i=0}^{\infty} R_i$ by R_∞, $\bigcup_{i=0}^{\infty} G_i$ by G_∞, and $\bigcup_{i=0}^{\infty} U_i$ by U_∞. An inference sequence is called **fair**, if it satisfies the following conditions.

(1) Any critical pair between equations in R_∞ is contained in E_∞.

(2) $\bigcup_{i=0}^{\infty} \bigcap_{j=i}^{\infty} E_j = \emptyset$. In other words, any equation in E_∞ is removed from E eventually by E-reduction, E-deletion, or R-generation.

(3) Any goal obtained from a goal in G_∞ and an equation in R_∞ by G-generation is contained in G_∞.

(4) Any substitution obtained from a goal in G_∞ by U generation is contained in U_∞.

We claim that any fair inference sequence can enumerate a complete set of $\simeq$-unifiers as U_∞.

Example 2.1

Consider equation set $\mathcal{R} = \{\langle x \times x, x^2\rangle, \langle 1 \times y, y\rangle\}$ of equations and terms $s = z^2$ and $t = 1$, where $\preceq$ is the lexicographic subterm ordering based on total order $1 <^2 <^2 \times$

on the function symbols. Then, the following is a fair inference sequence.

$$
\begin{aligned}
&(E_0 = \{\langle x \times x, x^2\rangle, \langle 1 \times y, y\rangle\}, R_0 = \emptyset, G_0 = \{\langle z^2, 1, \epsilon\rangle\}, U_0 = \emptyset)\\
&\quad \vdash_{R\text{ generation}} (E_1 = \{\langle 1 \times y, y\rangle\}, R_1 = \{\langle x \times x, x^2\rangle\}, G_1 = G_0, U_1 = \emptyset)\\
&\quad \vdash_{R\text{-generation}} (E_2 = \emptyset, R_2 = R_1 \cup \{\langle 1 \times y, y\rangle\}, G_2 = G_1, U_2 = \emptyset)\\
&\quad \vdash_{E\text{-generation}} (E_3 = \{\langle 1^2, 1\rangle\}, R_3 = R_2, G_3 = G_2, U_3 = \emptyset)\\
&\quad \vdash_{R\text{-generation}} (E_4 = \emptyset, R_4 = R_3 \cup \{\langle 1^2, 1\rangle\}, G_4 = G_3, U_4 = \emptyset)\\
&\quad \vdash_{G\text{-generation}} (E_5 = \emptyset, R_5 = R_4, G_5 = G_4 \cup \{\langle 1, 1, [1/z]\rangle\}, U_5 = \emptyset)\\
&\quad \vdash_{U\text{-generation}} (E_6 = \emptyset, R_6 = R_5, G_6 = G_5, U_6 = \{[1/z]\})
\end{aligned}
$$

Thus, $\simeq$-unifier $[1/z]$ of $s = z^2$ and $t = 1$ is obtained as an element of U_6 in the above sequence, where notation $[1/z]$ expresses the substitution θ such that $z\theta = 1$ and $v\theta = v$ for any variables v other than z.

3. Completeness of the unification procedure

First of all, we prove the soundness of the procedure given in the previous section.

Theorem 3.1

Let

$$(\Re, \emptyset, \{\langle s, t, \epsilon\rangle\}, \emptyset) = (E_0, R_0, G_0, U_0) \vdash (E_1, R_1, G_1, U_1) \vdash (E_2, R_2, G_2, U_2) \vdash \cdots$$

be an inference sequence. Then, any element of U_∞ is an $\simeq$-unifier of s and t.

Proof: Let $\simeq_i$ be the congruence relation presented by $E_i \cup R_i$. Then it is easy to prove the following by induction on i.

(1) $\simeq_i = \simeq$.

(2) For any $\langle u_1, u_2, \theta\rangle \in G_i$, $s\theta \simeq u_1$ and $t\theta \simeq u_2$.

(3) For any $\theta \in U_i$, $s\theta \simeq t\theta$. ∎

The proof of completeness of the procedure consists of two parts. First, R_∞ is proved to be confluent by evidence transformation method [Bachmair 86]. Second, narrowing is proved to be able to trace any rewriting by R_∞.

Hereafter, we use symbols $\Rightarrow$ to denote reduction in the new sense defined in Section 2. On the other hand, we use symbols $\Leftrightarrow$ to denote the symmetric closure of reduction in the old sense defined in Section 1.

Let $\Re$ be a set of equations, and g and g' ground terms such that $g \simeq g'$. Then, from the definition, there is a finite sequence of terms

$$g = g_0 \Leftrightarrow g_1 \Leftrightarrow \cdots \Leftrightarrow g_m = g'.$$

Let us define sequences of this form in a more general framework. A sequence $g = g_0 \Xi_1 g_1 \Xi_2 \cdots \Xi_m g_m = g'$ is called an **evidence** of $g \simeq g'$ by E and R, if each g_i is a ground term and Ξ_i is one of the following symbols:

(1) $\Leftrightarrow$, which indicates that $g_{i-1} \Leftrightarrow g_i$ by E.

(2) $\Leftarrow$, which indicates that $g_i \Rightarrow g_{i-1}$ by R.

(3) $\Rightarrow$, which indicates that $g_{i-1} \Rightarrow g_i$ by R.

An evidence is said to be **normal** if it has the following form

$$g = g_0 \Rightarrow g_1 \Rightarrow \cdots \Rightarrow g_{m-1} \Rightarrow h \Leftarrow g'_{n-1} \Leftarrow \cdots \Leftarrow g'_1 \Leftarrow g'_0 = g' \quad (m \geq 0, n \geq 0)$$

Now, we will define the weight of an evidence. First, the weight $w(g \Xi g')$ of each step $g \Xi g'$ of an evidence is defined as follows:

$$w(g \Leftrightarrow g') = \{g, g'\}, \quad w(g \Leftarrow g') = \{g'\}, \quad w(g \Rightarrow g') = \{g\}$$

where $\{g, g'\}$, $\{g'\}$, and $\{g\}$ are not sets but multi-sets, and are compared by the multi-set ordering [Dershowitz 79]. The weight of an evidence is defined as the multi-set consisting of the weights of all the steps of the evidence. Note that, since the weight of a step is a multi-set, the weight of an evidence is a doubly-multi-set (a multi-set of multi-sets of terms). The set of the weights of evidences is well-founded since the base order is well-founded. Let us denote the order also by $\preceq$.

Theorem 3.2

Let

$$(E_0 = \Re, R_0 = \emptyset, G_0, U_0) \vdash (E_1, R_1, G_1, U_1) \vdash (E_2, R_2, G_2, U_2) \vdash \cdots$$

be a fair inference sequence. Then, R_∞ is a confluent set of equations for $\simeq$ w.r.t. ground terms.

Proof: It is sufficient to prove that, for any ground terms g and g' such that $g \simeq g'$, there exist a normal evidence of $g \simeq g'$ by E_∞ and R_∞. (Since the evidence is normal, it expresses reduction of g and g' to the same term by R_∞.) Let g and g' be arbitrary ground terms such that $g \simeq g'$. Then, there is an evidence of $g \simeq g'$ by E_0 and R_0, which is also an evidence by E_∞ and R_∞ of course. Let ¶ be an evidence by E_∞ and R_∞ with minimal weight. We prove that ¶ is normal. First we prove that ¶ contains no steps of the form

$$c[u_1\theta] \Leftrightarrow c[u_2\theta] \qquad (A)$$

where $\langle u_1, u_2 \rangle \in E_i$ for some i. Suppose that such a step exists. From fairness condition (2), for some j such that $i < j$, equation $\langle u_1, u_2 \rangle$ must be deleted from E_j; that is, inference rule E-reduction, E-deletion, or R-generation must be applied to $\langle u_1, u_2 \rangle$. If it is E-reduction, $\langle u_1, u \rangle \in E_j$ (or $\langle u, u_2 \rangle \in E_j$) for some u such that $u_2 \Rightarrow u$ (or $u_1 \Rightarrow u$ by R_j). Therefore, by replacing the step of form (A) with two steps

$$c[u_1\theta] \Leftrightarrow c[u\theta] \Leftarrow c[u_2\theta] \quad (\text{or} \quad c[u_1\theta] \Rightarrow c[u\theta] \Leftrightarrow c[u_2\theta]),$$

we can obtain a new evidence ¶′. Comparing the weight of the steps, that is, $\{c[u_1\theta], c[u_2\theta]\}$ in ¶ and $\{c[u_1\theta], c[u\theta]\}, \{c[u_2\theta]\}$ (or $\{c[u_1\theta]\}, \{c[u\theta], c[u_2\theta]\}$) in ¶′, we can easily see that $w(¶') \preceq w(¶)$, which contradicts that ¶ has minimal weight. If the inference step is E-deletion, u_1 must be equal to u_2. Therefore, by simply removing the step of

form (A), we can obtain a new evidence, which again contradicts that ¶ has minimal weight. If the inference step is R-generation, R_j contains equation $\langle u_1, u_2 \rangle$. In this case, the step of form (A) can be replaced with

$$c[u_1\theta] \Rightarrow c[u_2\theta] \quad \text{or} \quad c[u_1\theta] \Leftarrow c[u_2\theta]$$

since $\preceq$ is total for ground terms, and a contradiction follows. Next, we prove that ¶ contains no steps of the form

$$h_1 \Leftarrow h \Rightarrow h_2 \tag{B}$$

Suppose that there are steps of form (B), in which term h is reduced in two ways, say, to h_1 by equation $\langle l_1, r_1 \rangle \in R_i$ and to h_2 by equation $\langle l_2, r_2 \rangle \in R_j$. There are several cases. First assume that the reduced parts do not overlap, that is h, h_1, and h_2 have forms $c[l_1\theta_1, l_2\theta_2]$, $c[r_1\theta_1, l_2\theta_2]$, and $c[l_1\theta_1, r_2\theta_2]$. In this case, by replacing the steps of form (B) with

$$h_1 \Rightarrow c[r_1\theta_1, r_2\theta_2] \Leftarrow h_2$$

we can obtain a new evidence, which contradicts that ¶ has minimal weight. Next assume that the reduced parts overlap. Since the discussion is symmetrical, we can assume that $h = d[c[l_1\theta_1]] = d[l_2\theta_2]$, $h_1 = d[c[r_1\theta_1]]$, and $h_2 = d[r_2\theta_2]$ without loss of generality. If $l_1\theta_1$ occurs at a variable position in l_2, we can easily arrive at a contradiction similarly to the non-overlapping case. Otherwise, $\langle c[r_1\theta_1], r_2\theta_2 \rangle$ is an instance of a critical pair of equations $\langle l_1, r_1 \rangle$ and $\langle l_2, r_2 \rangle$. From fairness condition (1), the critical pair must be in some E_k. Then, by replacing the steps of form (B) with

$$h_1 = d[c[r_1\theta_1]] \Leftrightarrow d[r_2\theta_2] = h_2,$$

we arrive at a contradiction again. Thus, we have proved that ¶ contains no steps of form (A) or (B). Such an evidence is clearly normal. ∎

If there is a normal evidence

$$g_0 \Rightarrow g_1 \Rightarrow \cdots \Rightarrow g_{m-1} \Rightarrow h \Leftarrow g'_{n-1} \Leftarrow \cdots \Leftarrow g'_1 \Leftarrow g'_0 \tag{C}$$

we can always convert it to a one-way reduction sequence of pairs of terms of the following form:

$$\langle g_0, g'_0 \rangle = p_0 \Rightarrow p_1 \Rightarrow \cdots \Rightarrow p_{m+n} = \langle h, h \rangle. \tag{C'}$$

In each step, either the left or the right element of pairs is reduced. In what follows, sequences of form (C') are called normal evidences instead of those of form (C) for simplicity of discussion.

A substitution σ is said to be **irreducible** if $v\sigma$ is irreducible for any variable v.

Theorem 3.3 [Hullot 80]

Let u be a term (or pair of terms) and θ be an irreducible substitution. Then, for any sequence of reduction

$$u\theta = g_0 \Rightarrow g_1 \Rightarrow \cdots \Rightarrow g_n,$$

there is a sequence of narrowing

$$u = u_0 \hookrightarrow_{\theta_0} u_1 \hookrightarrow_{\theta_1} \cdots \hookrightarrow_{\theta_{n-1}} u_n$$

and a sequence of irreducible substitutions $\psi_0, \psi_1, \ldots, \psi_n$ *such that*

$$g_i = u_i \psi_i \ (i = 0, 1, \ldots, n)$$

and

$$\theta = \psi_0 = \theta_0 \circ \psi_1 = \cdots = \theta_0 \circ \cdots \circ \theta_{n-1} \circ \psi_n.$$

In the original form of the above theorem, the concepts of reduction and narrowing are the conventional left-to-right-only ones, the set of equations is assumed to be confluent and terminating, and substitution θ is assumed to be normal. However, the above form of the theorem can also be proved in the same way as the original.

Now, we are ready to prove the completeness of the $\simeq$-unification procedure.

Theorem 3.4

Let $\mathcal{R}$ *be a set of equations,* s *and* t *be terms, and*

$$(\mathcal{R}, \emptyset, \langle s, t, \epsilon \rangle, \emptyset) = (E_0, R_0, G_0, U_0) \vdash (E_1, R_1, G_1, U_1) \vdash (E_2, R_2, G_2, U_2) \vdash \cdots$$

be a fair inference sequence. Then, U_∞ *is a complete set of* $\simeq$*-unifiers of* s *and* t*. That is, for any* $\simeq$*-unifier* θ *of* s *and* t*, there is a substitution* $\theta' \in U_\infty$ *more general than* θ*.*

Proof: By replacing variables in $s\theta$ and $t\theta$ with fresh constants, we can assume that $s\theta$ and $t\theta$ are ground terms without loss of generality. Moreover, by replacing the value of θ at each variable with its normal form w.r.t. R_∞, we can assume that θ is irreducible. Since $s\theta \sim t\theta$, there is a term h and a normal evidence

$$\langle s, t \rangle \theta = p_0 \Rightarrow p_1 \Rightarrow \cdots \Rightarrow p_n = \langle h, h \rangle$$

by R_∞. Then, from Theorem 3.3, there is a sequence of narrowing by R_∞

$$\langle s, t \rangle = \langle s_0, t_0 \rangle \hookrightarrow_{\theta_0} \langle s_1, t_1 \rangle \hookrightarrow_{\theta_1} \cdots \hookrightarrow_{\theta_{n-1}} \langle s_n, t_n \rangle$$

and a sequence of irreducible substitutions $\psi_0, \psi_1, \ldots, \psi_n$ such that $p_i = \langle s_i, t_i \rangle \psi_i \ (i = 0, 1, \ldots, n)$ and

$$\theta = \psi_0 = \theta_0 \circ \psi_1 = \cdots = \theta_0 \circ \cdots \circ \theta_{n-1} \circ \psi_n.$$

From fairness condition (3), we can easily prove by induction that, for each i, $\langle s_i, t_i, \theta_0 \circ \cdots \circ \theta_{i-1} \rangle \in G_\infty$, in particular, $\langle s_n, t_n, \theta_0 \circ \cdots \circ \theta_{n-1} \rangle \in G_\infty$. Since $s_n \psi_n = h = t_n \psi_n$, s_n and t_n are unifiable. Let ψ be the most general unifier of s_n and t_n. Then, from fairness condition (4), $\theta' = \theta_0 \circ \cdots \circ \theta_{n-1} \circ \psi \in U_\infty$, which is more general than $\theta = \theta_0 \circ \cdots \circ \theta_{n-1} \circ \psi_n$. ∎

As shown in Theorem 3.2, the $\simeq$-unification is an extension of the Knuth-Bendix completion procedure. In particular, if $R_i = R_\infty$ for some i, a finite confluent and terminating set of equations is obtained after a finite number of steps of inference. Then, the subsequent process can be assumed to consist only of G-generations and U-generations since the other rules cause no essential change in R_i, G_i, and U_i. Therefore, the procedure can be viewed as an extension of Fay's procedure. Moreover, if $G_j = G_\infty$ for some j (in fact, Example 2.1 is this case), we can obtain a finite complete set U_∞ of $\simeq$-unifiers of s and t. Note that, even in this case, U_∞ is not necessarily the minimum complete set.

4. Implementation issues and examples

There are a lot of things to be considered for efficiency in actual implementation of the procedure discussed in the previous section.

If the proof of Theorem 3.2 is examined, it can be easily seen that the inference rules E-reduction and E-deletion do not contribute to the completeness of the procedure. In fact, these rules are introduced for efficiency. To improve efficiency further, the following inference rules should be taken into consideration. If these rules are given priority over the generation rules, they will save a lot of time by not applying useless inferences.

R-reduction: $\dfrac{(E, R \cup \{\langle u_1, u_2\rangle\}, G, U)}{(E \cup \{\langle u_1, u_2'\rangle\}, R, G, U)}$ $u_2 \Rightarrow u_2'$ by an equation in R

G-reduction: $\dfrac{(E, R, G \cup \{\langle u_1, u_2, \theta\rangle\}, U)}{(E, R, G \cup \{\langle u_1, u_2', \theta\rangle\}, U)}$ $u_2 \Rightarrow u_2'$ by an equation in R

G-deletion: $\dfrac{(E, R, G \cup \{\langle u_1, u_2, \theta\rangle\}, U)}{(E, R, G, U)}$

θ is reducible by R or an element of U is more general than θ

U-deletion: $\dfrac{(E, R, G, U \cup \{\theta\})}{(E, R, G, U)}$

θ is reducible by R or an element of U is more general than θ

The reader can clearly see the role of R-reduction and G-reduction. Rules G-deletion and U-deletion play a similar role to that the basic narrowing plays in Hullot's procedure [Hullot 80]

Even if the above inference rules are also employed, the procedure is still complete. To prove its completeness, however, the evidence order and the limits need more subtle treatment, and this would introduce a simple but long discussion, which we have avoided in the proof of Theorem 3.2. For example, if R-reduction is employed, R_∞ must not be defined as $\bigcup_{i=1}^{\infty} R_i$ but as $\bigcup_{i=1}^{\infty} \bigcap_{j=i}^{\infty} R_j$, since R_i is no longer increasing.

We will show several examples of $\simeq$-unifications in combinatory logic. In the examples, we use the strong simplification order $\preceq$ based on lexicographic subterm ordering. Terms of the form $*(\cdots(*(x, y), \cdots), z)$ are abbreviated to the form $xy\cdots z$ in the following inference sequence.

Example 4.1

An identity combinator **i** is defined as a combinator with property $\forall x\ \mathbf{i}x = x$. Here, we show the example of automatic construction of **i** from **s** and **k** by $\simeq$-unification. Let $\Re$ be $\{\langle \mathbf{s}xyz, xz(yz)\rangle, \langle \mathbf{k}xy, x\rangle\}$ (that is, consist of the defining equation for **s** and **k**), and let us try to $\simeq$-unify $s = vc$ and $t = c$. Function symbols are ordered as

$c < \mathbf{k} < \mathbf{s} < *$.

$$
\begin{aligned}
&(E_0 = \{\langle \mathbf{s}xyz, xz(yz)\rangle,\ \langle \mathbf{k}xy, x\rangle\}, R_0 = \emptyset, G_0 = \{\langle vc, c, \epsilon\rangle\}, U_0 = \emptyset)\\
&\quad \vdash_{R\text{-generation}} (E_1 = \{\langle \mathbf{s}xyz, xz(yz)\rangle\}, R_1 = \{\langle \mathbf{k}xy, x\rangle\}, G_1 = G_0, U_1 = \emptyset)\\
&\quad \vdash_{R\text{-generation}} (E_2 = \epsilon, R_2 = R_1 \cup \{\langle \mathbf{s}xyz, xz(yz)\rangle\}, G_2 = G_1, U_2 = \emptyset)\\
&\quad \vdash_{E\text{-generation}} (E_3 = \{\langle \mathbf{sk}xy, y\rangle\}, R_3 = R_2, G_3 = G_2, U_3 = \emptyset)\\
&\quad \vdash_{R\text{-generation}} (E_4 = \epsilon, R_4 = R_3 \cup \{\langle \mathbf{sk}xy, y\rangle\}, G_4 = G_3, U_4 = \emptyset)\\
&\quad \vdash_{G\text{ generation}} (E_5 = \epsilon, R_5 = R_4, G_5 = G_4 \cup \{\langle c, c, [\mathbf{sk}x/v]\rangle\}, U_5 = \emptyset)\\
&\quad \vdash_{U\text{-generation}} (E_6 = \epsilon, R_6 = R_5, G_6 = G_5, U_6 = \{[\mathbf{sk}x/v]\})
\end{aligned}
$$

Thus, identity combinator $\mathbf{sk}x$ is obtained as the term substituted to v.

Remark: Strictly speaking, an $\simeq$-unifier of vc and c is not necessarily an identity combinator, since it may depend on c (that is, the term substituted to v may contain c as its subterm). If we want to restrain such a unifier from being generated, we should $\simeq$-unify $vc(v)$ and $c(v)$. Note that disequation $vc(v) \neq c(v)$ is the Skolem form of the negation of formula $\forall x\ cx = x$.

Example 4.2

Next let us try the mockingbird problem [Smullyan 85]. A mockingbird is a combinator $\mathbf{m}$ with property $\forall x\ \mathbf{m}x = xx$. The problem is to construct a fixed point of a given combinator c from $\mathbf{m}$, $\mathbf{b}$, and c itself, where $\mathbf{b}$ is a composition combinator, which has property $\forall x\ \forall y\ \forall z\ \mathbf{b}xyz = x(yz)$. A fixed point of c is defined as a combinator f with property $cf = f$.

We set E_0 to be $\{\langle \mathbf{m}x, xx\rangle, \langle \mathbf{b}yzw, y(zw)\rangle\}$, R_0 to be $\emptyset$, G_0 to be $\{\langle cv, v, \epsilon\rangle\}$, and U_0 to be $\emptyset$, and execute the $\simeq$-unification procedure. We do not trace the details, but a fixed point of c is obtained through the following process.

(1) New equation $\langle \mathbf{m}(\mathbf{b}yz), y(z(\mathbf{b}yz))\rangle$ is obtained as a critical pair of equations $\langle \mathbf{m}x, xx\rangle$ and $\langle \mathbf{b}yzw, y(zw)\rangle$ by E-generation.

(2) New goal $\langle \mathbf{m}(\mathbf{b}cz), z(\mathbf{b}cz), [z(\mathbf{b}cz)/v]\rangle$ is obtained from goal $\langle cv, v, \epsilon\rangle$ and equation $\langle \mathbf{m}(\mathbf{b}yz), y(z(\mathbf{b}yz))\rangle$ by G-generation.

(3) Finally, we can generate $\simeq$-unifier $[\mathbf{m}(\mathbf{bcm})/v]$ of cv and v from the above goal $\langle \mathbf{m}(\mathbf{b}cz), z(\mathbf{b}cz), [z(\mathbf{b}cz)/v]\rangle$ by U-generation, and $\mathbf{m}(\mathbf{bcm})$ is a fixed point of c in fact.

REFERENCES

[Bachmair 86] Bachmair, L., Dershowitz, N., and Hsiang, J.: *Ordering for equational proof*, Proc. Symp. Logic in Computer Science, Cambridge, Massachusetts (June 1986)

[Bachmair 87] Bachmair, L., Dershowitz, N., and Plaisted, D. A.: *Completion without failure*, Proc. Colloquium on Resolution of Equations in Algebraic Structures (1987)

[Bosco 87] Bosco, P. G., Giovannetti, F., and Moiso, C.: *Refined strategies for equational proofs*. TAPSOFT '87, LNCS 250, pp. 276–290 (1987)

[Dershowitz 79] Dershowitz, N. and Manna, Z.: *Proving termination with multiset orderings*. Comm. ACM 22, pp. 465–467 (1979)

[Dershowitz 82] Dershowitz, N.: *Orderings for term-rewriting systems*. Theoretical Computer Science 17, pp. 279–301 (1982)

[Fages 84] Fages, F.: *Associative-commutative unification*. 7th International Conference on Automated Deduction, LNCS 170, pp. 194–208 (1984)

[Fages 86] Fages, F. and Huet, G.: *Complete sets of unifiers and matchers in equational theories*. Theoretical Computer Science 43, pp. 189–200 (1986)

[Fay 79] Fay, M.: *First order unification in an equational theory*. 4th workshop on Automated Deduction, Austin, Texas, pp. 161–167 (1979)

[Gallier 87] Gallier, J. H. and Snyder, W.: *A general complete E-unification procedure*. Rewriting Techniques and Applications, LNCS 256, pp. 216–227 (1987)

[Hsiang 87] Hsiang, J.: *Rewrite method for theorem proving in first order theory with equality*. J. Symbolic Computation, 3, 133–151 (1987)

[Huet 78] Huet, G.: *An algorithm to generate the basis of solutions to homogeneous linear Diophantine equations*. Inform. Process. Lett. 7, pp. 144–147 (1978)

[Huet 80] Huet, G. and Oppen, D. C.: *Equations and Rewrite Rules: a survey*. Formal Language: Perspectives and Open Problems Academic Press, pp. 349–405 (1980)

[Huet 81] Huet, G.: *A complete proof of correctness of the Knuth-Bendix completion algorithm*. J. Computer and System Science 23, pp. 11–21 (1981)

[Hullot 80] Hullot, J. M.: *Canonical forms and unification*. 5th Workshop on Automated Deduction, LNCS 87, pp. 318–334 (1980)

[Knuth 70] Knuth, D. E. and Bendix, P. B.: *Simple word problems in universal algebras*. Computational problems in abstract algebra, Pergamon Press, Oxford (1970)

[Makanin 77] Makanin, G. S.: *The problem of solvability of equations in a free semigroup*. Soviet Akad. Nauk SSSR, Tom 233, No. 2 (1977)

[Robinson 65] Robinson, J. A.: *A machine-oriented logic based on the resolution principle*. J. ACM 12, pp. 23–41 (1965)

[Rusinowitch 88] Rusinowitch, M.: *Theorem-proving with resolution and superposition: an extension of the Knuth and Bendix procedure to acomplete set of inference*

rules. International Conference on Fifth Generation Computer Systems, 1988, pp. 524–531 (1988)

[Sakai 84] Sakai, K.: *An ordering method for term rewriting systems*. Technical Report 062, ICOT (1984)

[Siekmann 89] Siekmann, J. H.: *Unification theory*. J. Symbolic Computation 7, pp. 207–274 (1989)

[Smullyan 85] Smullyan, R. M.: *To Mock a Mockingbird*. Alfred A. Knopf, Inc. (1985)

[Stickel 81] Stickel, M.E.: *A unification algorithm for associative-commutative functions*. J. ACM, 28, 3, pp. 423–434 (1981)