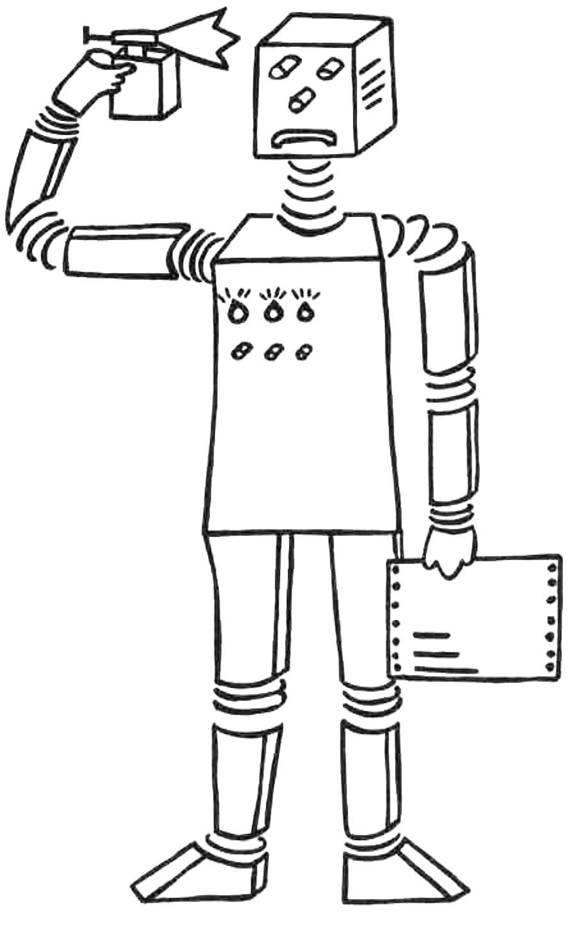
SEKI-REPORT

Artificial
Intelligence
Laboratories

Fachbereich Informatik
Universität Kaiserslautern
Postfach 3049
D-6750 Kaiserslautern 1, W. Germany



NARROWING TECHNIQUES APPLIED TO IDEMPOTENT UNIFICATION

Alexander Herold

August 1986 SEKI-REPORT SR-86-16

Narrowing Techniques
Applied to Idempotent Unification

Alexander Herold

Fachbereich Informatik
Universität Kaiserslautern
Postfach 3049
6750 Kaiserslautern

net-address: UUCP: ...!mcvax!unido!uklirb!herold

Abstract:

A complete unification algorithm for idempotent functions is presented.This algorithm is derivated from the universal unification algorithm, which is based on the narrowing relation. First an improvement for the universal algorithm is shown. Then these results are applied to the special case of idempotence resulting in an idempotent unification algorithm. Finally several refinements for this algorithm are proposed.

Contents

1. Introduction

Unification theory is concerned with problems of the following kind: given two terms built from function symbols, constants and variables, do there exist terms that can be substituted for the variables such that the two terms thus obtained become equal? Robinson [Ro 65] was the first to give an algorithm to find such a substitution with the additional property that the returned 'unifier' is most general (or is an mgu for short), i.e. all other substitutions 'unifying' the two terms can be computed from that substitution. From an algebraic point of view unification is solving equations and an mgu is a 'basis' of the whole set of unifiers.

Equational unification extends the classical unification problem to solving equations in equationally defined theories. But then there may not exist one single mgu. Depending on the equational theory there are finite or infinite sets of mgu's and in some cases even a set of mgu's does not even exist. Equational theories can therefore be classified into unitary (a single mgu exists), finitary (there is a finite set of mgu's) and infinitary (the set of mgus is infinite) theories and the class of nullary theories (i.e. a set of most general unifiers does not exist). For a detailed bibliography we refer to the state-of-the-art survey of J. Siekmann [Si 86].

Since every equational theory T requires a special purpose T-unification algorithm, there are recent attempts to combine these special unification algorithms ([He 85], [Ki 85], [Ti 86] and [Ye 85]). But all these combination algorithms do not work for collapse theories. Collapse theories are those theories which contain an axiom t = x, e.g. idempotence f(x x) = x. On the other hand there is a very powerful tool for canonical theories: the universal unification algorithm [Fy 79], [Hl 80], [SS 81], [Sz 82]. Given a canonical term rewriting system for an equational theory we at once have a unification algorithm for that theory. This algorithm is based on narrowing using rewrite rules, which is essentially oriented paramodulation [RW 69] and it can be extended to equational rewrite systems ([He 82], [Hl 80] and [JK 83]). This extension is one possibility to avoid the restriction in the combination algorithms for collapse free theories. Theories which are only defined by one collapse axiom mostly have a canonical rewrite system and by a result of Hullot the universal unification algorithm terminates for those theories [Hl 80]. Moreover the universal unification algorithm can trivially be combined with uninterpreted function symbols since these symbols do not change the term rewriting system.

As a prototype we studied the theory of idempotence. Since one problem with the universal unification algorithm is its inefficiency we tried to improve the obvious solution given by the universal unification algorithm by pruning the search space. After a introduction into the notions and notations of unification theory we present the universal unification algorithm. We then give a method to improve this universal algorithm. These results are applied to idempotent functions resulting in a first version of an idempotent unification algorithm. Exploiting the fact that we only have idempotent and uninterpreted function symbols we present special results improving this first version for an idempotent unification algorithm.

2. Definitions and Notations

2.1 Terms and Substitutions

Unification theory rests upon the usual algebraic notions (see e.g. [Gr 79], [BS 81]) with the familiar concept of an algebra $\mathcal{A} = (\mathbb{A}, \mathbb{F})$ where $\mathbb{A}$ is the ***carrier*** and $\mathbb{F}$ is a family of ***operators*** given with their arities.

Assuming that there is at least one constant (operator of arity 0) in $\mathbb{F}$ and a denumerable set of variables $\mathbb{V}$, we define $\mathbb{T}$, the set of ***first order terms***, over $\mathbb{F}$ and $\mathbb{V}$, as the least set with (i) $\mathbb{V} \subseteq \mathbb{T}$, and if arity(f) = 0 for $f \in \mathbb{F}$ then $f \in \mathbb{T}$ and (ii) if $t_1, \dots, t_n \in \mathbb{T}$ and arity(f) = n then the string $f(t_1 \dots t_n) \in \mathbb{T}$. Let $\mathbb{V}(s)$ be the set of variables occurring in a term s; a term s is ***ground*** if $\mathbb{V}(s) = \emptyset$.

As usual the algebra with carrier $\mathbb{T}$ and with operators, namely the term constructors corresponding to each operator of $\mathbb{F}$, is the absolutely free (term) algebra, i.e. it just gives an algebraic structure to $\mathbb{T}$. If the carrier is the set of ground terms it is the initial algebra [GT 78], also known as Herbrand Universe [Lo 78].

Terms can be also viewed as labelled trees [JK 83]: A term is a partial application of $\mathbb{N}^*$ into $\mathbb{F} \cup \mathbb{V}$ such that its domain D(t) satisfies the following condition:

the empty word ε is in D(t) and $\pi.i$ is in D(t) iff π is in D(t) and $i \in [1, \text{arity}(t(\pi))]$,

where . denotes the concatenation of strings. D(t) is the set of ***occurrences*** and O(t) denotes the set of non-variable occurrences, t/π the subterm of t at the occurrence π of t and $t[\pi \leftarrow t']$ the term obtained by replacing t/π by t' in t. We say that two occurrences π_1 and π_2 are independent iff one selected term is not a subterm of the other selected term (i.e. there exists an occurrence π such that $\pi_1 = \pi.i.\pi'_1$, $\pi_2 = \pi.j.\pi'_2$ and $i \neq j$).

A ***substitution*** $\sigma: \mathbb{T} \to \mathbb{T}$ is an endomorphism on the term algebra which is identical almost everywhere on $\mathbb{V}$ and can be represented as a finite set of pairs: $\sigma = \{x_1 \leftarrow t_1 \dots x_n \leftarrow t_n\}$. The restriction $\sigma|_V$ of a substitution σ to a set of variables V is defined as $\sigma|_V x = \sigma x$ if $x \in V$ and $\sigma|_V x = x$ else. Σ is the set of substitutions and ε the identity. The application of a substitution σ to a term $t \in \mathbb{T}$ is written as σt. The composition of substitutions is defined as the usual composition of mappings : $(\sigma \circ \tau)t = \sigma(\tau t)$ for $t \in \mathbb{T}$.

Let $\quad DOM\sigma = \{x \in \mathbb{V} \mid \sigma x \neq x\}$ (domain of σ)

$COD\sigma = \{\sigma x \mid x \in DOM\sigma\}$ (codomain of σ)

$VCOD\sigma = \mathbb{V}(COD\sigma)$ (variables in codomain of σ)

If $VCOD\sigma = \emptyset$ then σ is a ***ground substitution.***

A set of substitutions $\Sigma \subseteq \Sigma$ is said to be ***based*** on a set of variables W away from $Z \supseteq W$ iff the following two conditions are satisfied

(i) $DOM\sigma = W$ for all $\sigma \in \Sigma$

(ii) $VCOD\sigma \cap Z = \emptyset$ for all $\sigma \in \Sigma$

In particular for substitutions based on some W we have $DOM\sigma \cap VCOD\sigma = \emptyset$ which is equivalent to the idempotence of σ, i.e. $\sigma \circ \sigma = \sigma$. We shall use this property in the proofs later on.

2.2 Equational Logic and Unification

An ***equation*** $s = t$ is a pair of terms. A set of equations T is called an ***equational theory*** iff an equation e is in T whenever e is true in every model of T i.e. e is a consequence of T (or for short: $e \in T$ whenever $T \models e$). A set of axioms P(T) of an equational theory T is a set of equations such that T is the least equational theory containing this set P(T). We sometimes say that the equational theory T is presented by P(T). For simplicity we do not distinguish between the equational theory and its presentation.

The equality $=_T$ generated by a set of equations T is the finest congruence over $\mathbb{T}$ containing all pairs $\sigma s = \sigma t$ for $s = t \in T$ and $\sigma \in \Sigma$. (i.e. the Σ-invariant congruence relation generated by T). The following is Birkhoffs well-known completeness theorem of equational logic [Bi 35]

Theorem 2.1: $\quad T \models s = t \quad$ iff $\quad s =_T t.$

We extend T-equality in $\mathbb{T}$ to the set of substitutions Σ by:

$$\sigma =_T \tau \quad \text{iff} \quad \forall x \in \mathbb{V} \quad \sigma x =_T \tau x .$$

If T-equality of substitutions is restricted to a set of variables W we write

$$\sigma =_T \tau\ [W] \quad \text{iff} \quad \forall x \in W \quad \sigma x =_T \tau x$$

and say σ and τ are ***T-equal on W***

A substitution τ is ***more general*** than σ on W (or σ is a ***T-instance*** of τ on W):

$$\sigma \le_T \tau\ [W] \quad \text{iff} \quad \exists \lambda \in \Sigma\ \sigma =_T \lambda\tau\ [W].$$

Two substitutions σ,τ are called ***T-equivalent*** on W

$$\sigma \equiv_T \tau\ [W] \quad \text{iff} \quad \sigma \le_T \tau\ [W] \text{ and } \tau \le_T \sigma\ [W]$$

Given two terms s, t and an equational theory T, a unification problem for T is denoted as

$$< s =_T t >$$

We say $\sigma \in \Sigma$ is a solution of $< s =_T t >$ (or σ is a T-unifier of s and t) iff $\sigma s =_T \sigma t$. For the set of all T-unifiers of s and t we write $U\Sigma_T(s, t)$. Without loss of generality we can assume that the unifiers of s and t are idempotent (if not, one can find an equivalent set of unifiers that is idempotent). For a given unification problem $< s =_T t >$, it is not necessary to compute the whole set of unifiers $U\Sigma_T(s, t)$, which is always recursively enumerable for an equational theory T with decidable word problem, but instead a smaller set useful in representing $U\Sigma_T$. Therefore we define $cU\Sigma_T(s, t)$, a ***complete set of unifiers*** of s and t on $W = V(s, t)$ as:

(i) $cU\Sigma_T(s, t) \subseteq U\Sigma_T(s, t)$ (correctness)

(ii) $\forall \delta \in U\Sigma_T(s, t)\ \exists \sigma \in cU\Sigma_T(s, t)\text{: } \delta \le_T \sigma\ [W]$ (completeness)

A ***set of most general unifiers*** $\mu U\Sigma_T(s, t)$ is a complete set with

(iii) $\forall \sigma, \tau \in \mu U\Sigma_T(s, t)\text{: } \sigma \le_T \tau\ [W]$ implies $\sigma = \tau$ (minimality).

For technical reasons it turned out to be useful to have the following requirement: For a set of variables Z with $W \subseteq Z$

(iv) $\mu US_T(s, t)$ (resp. $cU\Sigma_T(s, t)$) is based on W away from Z (protection of Z)

If conditions (i) - (iv) are rulfilled we say $\mu U\Sigma_T$ is a ***set of most general unifiers away from*** Z (resp. $cU\Sigma_T(s, t)$ is a ***complete set of unifiers away from*** Z) [PL 72].

The set $\mu U\Sigma_T$ does not always exist [FH 83][Ba 86][Sc 86]; if it does then it is unique up to the equivalence $\equiv_T$ [W] (see [Hu 76][FH 83]). For that reason it is sufficient to generate just one $\mu U\Sigma_T$ as some representative of the equivalence class $[\mu U\Sigma_T]_{\equiv T}$.

Depending on the cardinality of the set of most general unifiers we classify the equational theories into the following subclasses:

- a theory is ***unitary unifying*** iff $\mu U\Sigma_T$ exists and $|\mu U\Sigma_T(s, t)| = 1$ for all s and t
- a theory is ***finitary unifying*** iff $\mu U\Sigma_T$ exists and $|\mu U\Sigma_T(s, t)| < \infty$ for all s and t
- a theory is ***infinitary unifying*** iff $\mu U\Sigma_T$ exists and $|\mu U\Sigma_T(s, t)| = \infty$ for some s and t
- a theory is ***nullary unifying*** iff $\mu U\Sigma_T$ does not exist for some s and t.

A unification algorithm is called ***complete*** (and ***minimal***) if it returns a correct and complete (and minimal) set of unifiers for every pair of terms.

2.3 Reduction, Narrowing and Unification

A **term rewriting system** $R = \{l_1 \Rightarrow r_1, \ldots, l_n \Rightarrow r_n\}$ is a set of pairs of terms $l_i, r_i \in \mathbb{T}$ with $\mathbb{V}(r_i) \subseteq \mathbb{V}(l_i)$ for $1 \leq i \leq n$. We say that a term s $\rightarrow_R$-reduces to a term t at occurrence π with $l_i \Rightarrow r_i$ and we write $s \rightarrow_R t$ or $s \rightarrow_{[\pi,i]} t$ iff:

$$\exists\, l_i \Rightarrow r_i \in R,\ \sigma \in \Sigma,\ \pi \in O(s) \quad \text{such that} \quad s/\pi = \sigma l_i \ \text{ and } \ t = s[\pi \leftarrow \sigma r_i].$$

The indices of $\rightarrow_R$ are omitted if they are understood from the context. A term t is said to be ***reducible*** if $t \rightarrow_R t'$ for some t', else t is said to be ***irreducible*** or in $\rightarrow_R$***–normal form***. The reflexive and transitive closure of $\rightarrow_R$ is denoted by $\xrightarrow{*}_R$. A term rewriting system is said to be a ***complete*** set of reductions or a ***canonical*** term rewriting system iff:

$\rightarrow_R$ is ***noetherian***, i.e. there does not exist an infinite derivation $t_1 \rightarrow_R t_2 \rightarrow_R t_3 \ldots$

$\rightarrow_R$ is ***confluent***, i.e. for s, s_1, s_2 with $s \xrightarrow{*}_R s_1$ and $s \xrightarrow{*}_R s_2$
there exists t such that $s_1 \xrightarrow{*}_R t$ and $s_2 \xrightarrow{*}_R t$.

For an equational theory T there are techniques to obtain a term rewriting system R_T such that this system has the Church-Rosser property, i.e. $s =_T t$ iff there exists $r \in \mathbb{T}$ with $s \xrightarrow{*} r$ and $t \xrightarrow{*} r$; moreover it is sometimes possible to obtain a canonical term rewriting system [KB 70], [HO 84], [Hl 80a], [Bu 85]. Canonical systems are an important basis for computations in equational logics since they yield a decision procedure for T-equality: $s =_T t$ iff $\|s\| = \|t\|$ where $\|s\|$ denotes the unique normal form of s. A substitution σ is called in normal form or normalized iff all terms in the codomain are in normal form.

The following relation is the basis for a universal unification algorithm. We say s is ***narrowable*** to t at occurrence π with the substitution σ and the rule $l_i \Rightarrow r_i$ and write $s \rightarrowtail_{[\pi,i,\sigma]} t$ or shortly $s \rightarrowtail t$ iff

$$\exists\, l_i \Rightarrow r_i \in R,\ \sigma \in \Sigma,\ \pi \in O(s) \text{ such that } \sigma \text{ is mgu of } s/\pi \text{ and } l_i \text{ and } t = \sigma(s[\pi \leftarrow r_i]).$$

The substitution σ is called a narrower or narrowing substitution. Narrowing is the same as oriented paramodulation [RW 69]. In [Hl 80], [Hl 80b] the relationship between narrowing and reduction is established:

Theorem 2.2: Let s be a term and η be a normalized substitution with $DOM\eta \subseteq V(s)$. For every $\longrightarrow$ - derivation issuing from ηs

$$(1)\quad \eta s = t_0 \longrightarrow_{[\pi 1,k1]} t_1 \longrightarrow_{[\pi 2,k2]} t_2 \longrightarrow \cdots \longrightarrow_{[\pi n,kn]} t_n$$

there exists a $\rightarrowtail$ – derivation issuing from

$$(2)\quad s = s_0 \rightarrowtail_{[\pi 1,k1,\sigma 1]} s_1 \rightarrowtail_{[\pi 2,k2,\sigma 2]} s_2 \cdots \rightarrowtail_{[\pi n,kn,\sigma n]} s_n$$

for each i, $1 \leq i \leq n$, a normalized substitution η_i such that $\eta_i(s_i) = t_i$ and $\eta = \eta_i\theta_i\ [V(s)]$ where $\theta_i = \sigma_i \ldots \sigma_1$.
Conversely, to each $\rightarrowtail$ – derivation (2) and every η such that $\eta \leq \theta_n\ [V(s)]$ we can associate a $\longrightarrow$ - derivation (1).

This result can be depicted in the following diagram:

$$
\begin{array}{ccccccccc}
\eta s = t_0 & \rightarrow_{[\pi 1,k1]} & t_1 & \rightarrow_{[\pi 2,k2]} & t_2 & \rightarrow & \cdots & \rightarrow_{[\pi n,kn]} & t_n \\
\uparrow \eta_0 & & \uparrow \eta_1 & & \uparrow \eta_2 & & & & \uparrow \eta_n \\
s = s_0 & \rightarrowtail_{[\pi 1,k1,\sigma 1]} & s_1 & \rightarrowtail_{[\pi 2,k2,\sigma 2]} & s_2 & \rightarrowtail & \cdots & \rightarrowtail_{[\pi n,kn,\sigma n]} & s_n
\end{array}
$$

This relationship leads to the following universal unification algorithm which is essentially an enumerating process of the narrowing tree: Let t be a term then $\mathbb{N}_T(t)$ is the narrowing tree with

- t is the root of $\mathbb{N}_T(t)$
- if t' is a node in $\mathbb{N}_T(t)$ and $t' \rightarrowtail_{[\pi,k,\sigma]} t''$ then t'' is a successor node of t' in $\mathbb{N}_T(t)$.

Theorem 2.3: Let T be an equational theory that admits a canonical term rewriting system. Let s, t be two terms and h a new function symbol not occurring in $\mathbb{F}$. Let $U\Sigma_0(s, t)$ be the set of all substitutions τ such that there exists a $\rightarrowtail$ – derivation:

$$h(s\ t) \rightarrowtail_{[\sigma 1]} h(s_1\ t_1) \rightarrowtail_{[\sigma 2]} \cdots \rightarrowtail_{[\sigma n]} h(s_n\ t_n)$$

where $\theta_n = \sigma_n \ldots \sigma_1$ is normalized, s_n and t_n are unifiable with the most general unifier β and $\tau = \beta\theta_n$. Then $U\Sigma_0(s, t)$ is a complete set of unifiers of s and t.

The details of basing the unifiers on $\mathbb{V}(s, t)$ are omitted for clarity. Since for every substitution produced by a narrowing sequence there is a corresponding reduction sequence from which we can show that the substitution is indeed a unifier for the given two terms, correctness is obvious. On the other end to every unifier η' there exists a normalized unifier η such that $\eta' =_T \eta$. Hence there exists a derivation $\eta h(s\ t) \xrightarrow{*} h(r\ r)$ and by Theorem 2.2 a corresponding $\rightarrowtail$ – derivation $h(s\ t) \rightarrowtail^{*} h(s_n\ t_n)$ with $\eta \leq \sigma\theta_n\ [\mathbb{V}(s, t)]$ where σ is a most general Robinson unifier of s_n and t_n. This establishes the completeness of the narrowing technique. So by enumeration of the narrowing tree $\mathbb{N}_T(h(s\ t))$ and ordinary unification at each node we can construct a complete unification algorithm. Since this algorithm is very inefficient it is important to find criteria for pruning subtrees out of the narrowing tree.

3. Pruning the Narrowing Tree

A first improvement of the above universal unification algorithm was given by Hullot [Hl 80]. He proposed to use only innermost-outermost $\rightarrowtail$ - derivations as a derivation strategy which he showed to be complete (since completeness is based on the computation of R-normal forms). More exactly he defined a derivation ***based on*** a set of occurrences: For a term t, a prefix-closed set of occurrences $O_0 \subseteq O(t)$ (i.e. if $\pi.i \in O_0$ then $\pi \in O_0$) and a normalized substitution η a $\longrightarrow$ - derivation issuing from ηt

$$\eta t = s_0 \longrightarrow_{[\pi 1,k1]} s_1 \longrightarrow_{[\pi 2,k2]} s_2 \longrightarrow \cdots \longrightarrow_{[\pi n,kn]} s_n$$

or a $\rightarrowtail$ – derivation issuing from t.

$$t = t_0 \rightarrowtail_{[\pi 1,k1,\sigma 1]} t_1 \rightarrowtail_{[\pi 2,k1,\sigma 2]} t_2 \rightarrowtail \cdots \rightarrowtail_{[\pi n,kn,\sigma n]} t_n$$

is based on O_0 iff $\pi_i \in O_i$ with

$$O_i = O_{i-1} \setminus \{\pi_i.\pi' \mid \pi_i.\pi' \in O_{i-1}\} \cup \{\pi_i.\pi'' \mid \pi'' \in O(r_{ki})\}$$

for $1 \le i \le n$.

Roughly spoken it is not allowed to narrow at the occurrences introduced by the substitutions. A derivation of t is said to be ***basic*** iff it is based on $O_0 = O(t)$. Moreover the method of basic derivations gives us a sufficient condition for the termination of the narrowing process [Hl 80]. We shall illustrate the definitions by an example. Let $\{f(x\ x) \Rightarrow x\}$ be the rewriting system then $f(f(x\ y)\ f(a\ z)) \rightarrowtail_{[\varepsilon]} f(a\ y) \rightarrowtail_{[\varepsilon]} a$ is not a basic derivation, but $f(f(x\ y)\ f(a\ z)) \rightarrowtail_{[2]} f(f(x\ y)\ a) \rightarrowtail_{[1]} f(x\ a) \rightarrowtail_{[\varepsilon]} a$ is basic.

Another well-known derivation strategy is to consider only leftmost derivations. For a term t, a prefix-closed set of occurrences $O_0 \subseteq O(t)$ and a normalized substitution η a $\longrightarrow$ - derivation issuing from ηt

$$\eta t = s_0 \longrightarrow_{[\pi 1,k1]} s_1 \longrightarrow_{[\pi 2,k2]} s_2 \longrightarrow \cdots \longrightarrow_{[\pi n,kn]} s_n$$

or a $\rightarrowtail$ – derivation issuing from t

$$t = t_0 \rightarrowtail_{[\pi 1,k1,\sigma 1]} t_1 \rightarrowtail_{[\pi 2,k1,\sigma 2]} t_2 \rightarrowtail \cdots \rightarrowtail_{[\pi n,kn,\sigma n]} t_n$$

is from ***left-to-right*** in O_0 iff $\pi_i \in O_i$ with

$$O_i = O_{i-1} \setminus \{\pi \in O_{i-1} \mid \pi = \pi'.i'.\pi'', \pi_i = \pi'.i.\pi''' \text{ and } i' \leq i\} \\ \cup \{\pi_i.\pi'' \mid \pi'' \in O(r_{ki})\}$$

for $1 \leq i \leq n$. If we consider the above example then $f(f(x\ y)\ f(a, z)) \rightarrowtail_{[1]} f(x\ f(a\ z)) \rightarrowtail_{[2]} f(x\ a)$ is a left-to-right derivation whereas $f(f(x\ y)\ f(a\ z)) \rightarrowtail_{[2]} f(f(x\ y)\ a) \rightarrowtail_{[1]} f(x\ a)$ is not. In other words if we have performed a narrowing step at an occurrence then we forbid to narrow on those occurrences that are left and independent of the current occurrence.

A derivation of t is said to be from left-to-right iff it is from left-to-right in $O_0 = O(t)$. We will now show that we do not loose completeness when we only use left-to-right derivations. Moveover we show that we keep completeness if left-to-right and innermost-outermost derivations are combined.

3.1 Lemma: Let $s = \eta t$ with η normalized, then every leftmost $\longrightarrow$ - derivation issuing from s is a left-to-right derivation and every leftmost innermost-outermost $\longrightarrow$ - derivation issuing from s is a left-to-right basic derivation.

Proofs: obvious.

3.2. Lemma: Let $s = \eta t$ with η normalized, then there exists a leftmost $\longrightarrow$ - derivation issuing from s to its normal form. Moreover there exists a leftmost innermost-outermost $\longrightarrow$ - derivation issuing from s to its normal form.

Proofs: obvious.

Consider now a $\rightarrowtail$ – derivation

$$t = t_0 \rightarrowtail_{[\pi_1,k_1,\sigma_1]} t_1 \rightarrowtail_{[\pi_2,k_1,\sigma_2]} t_2 \rightarrowtail \ldots \rightarrowtail_{[\pi_n,k_n,\sigma_n]} t_n$$

which is linked to a leftmost innermost-outermost $\longrightarrow$ – derivation to the normal form of ηt

$$\eta t = s_0 \longrightarrow_{[\pi_1,k_1]} s_1 \longrightarrow_{[\pi_2,k_2]} s_2 \longrightarrow \ldots \longrightarrow_{[\pi_n,k_n]} s_n$$

with normalized η. This derivation is a left-to-right basic derivation and since the rules are applied at the same occurrences the $\rightarrowtail$ – derivation is also a left-to-right basic derivation. Moreover we can restrict the enumeration process of the narrowing tree to regard only those nodes such that narrowing unifier that belongs to a node is normalized. We summmarize these results in the following proposition:

3.3 Proposition: Given two terms s and t and a normalized T-unifier h then there exists a left-to-right basic $\rightarrowtail$ – derivation of h(s, t)

$$h(s\ t) = h(s_0\ t_0) \rightarrowtail h(s_1\ t_1) \rightarrowtail \ldots \rightarrowtail h(s_n\ t_n)$$

such that s_n and t_n are unifiable with the most general Robinson unifier β and it is

$$\eta \leq \beta\theta_n\ [V(s, t)].$$

The narrower θ_n can always be assumed to be normalized.

We finally want to give a direct proof for the completeness of the left-to-right strategy which does not use the correspondence between $\rightarrowtail$ – derivations and $\longrightarrow$ – derivations.

Before stating the main technical lemma we need some definitions: we say two substitutions σ and τ are unifiable iff there exists a substitution λ such that $\lambda\sigma = \lambda\tau$. In the same way as for terms we define the most general unifier η of σ and τ as the substitution with $\eta\sigma = \eta\tau$ and $\lambda \leq \eta\ [V(\sigma, \tau)]$ for all unifiers λ of σ and τ. The most general instance $\sigma * \tau = \eta\sigma = \eta\tau$ of σ and τ is sometimes called the merge of σ and τ. Note that the merge is commutative $\sigma * \tau = \tau * \sigma$.

3.4 Lemma: Let s and t be terms with the most general Robinson unifier σ and τ be an arbitrary substitution. Then

(i) σ and τ are unifiable iff τs and τt are unifiable

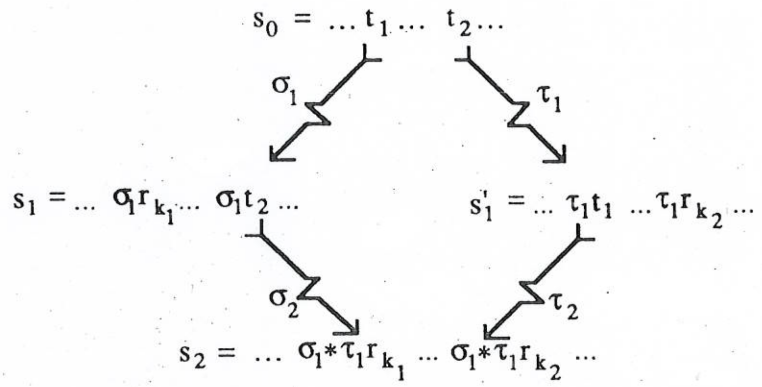
(ii) if σ and τ are unifiable (or τs and τt are unifiable) then $\sigma * \tau = \eta\tau$, where η is a most general unifier of τs and τt.

Proof: [He 83]

3.5 Lemma: Given a $\rightarrowtail$ – derivation $s_0 \rightarrowtail_{[\pi 1, k1, \sigma 1]} s_1 \rightarrowtail_{[\pi 2, k2, \sigma 2]} s_2$ with $\pi_1 = \pi.i.\pi' \in O(s_0)$, $\pi_2 = \pi.i.'\pi'' \in O(s_0)$ and $i < i'$ then there exists a $\rightarrowtail$ – derivation $s_0 \rightarrowtail_{[\pi 2, k2, \tau 2]} s_1{}' \rightarrowtail_{[\pi 1, k1, \sigma 1]} s_2$ with $\tau_2\tau_1 = \sigma_2\sigma_1 = \sigma_1 * \tau_1$.

Proof: We define $t_1 = s_0/\pi_1$ and $t_2 = s_0/\pi_2$ then σ_1 is an mgu of $\tilde{t}_1$ and l_{k1}. Since $\sigma_1 t_2$ and l_{k2} are unifiable with mgu σ_2 and $\sigma_1 l_{k2} = l_{k2}$ we can apply the last lemma and conclude that t_2 and l_{k2} are unifiable with mgu τ_1 and that $\sigma_2\sigma_1 = \sigma_1 * \tau_1$. Moreover of t_2 and l_{k2}. Hence we can perform the first narrowing step $s_0 \rightarrowtail_{[\pi 1, k1, \tau 1]} s_1{}'$. Since $\tau_1 l_{k1} = l_{k1}$ and $\sigma_1 * \tau_1 = \lambda\sigma_1 = \lambda\tau_1$ $\tau_1 t_1$ and l_{k1} are unifiable with mgu τ_2 and by the last lemma we get $\tau_2\tau_1 = \sigma_1 * \tau_1$. Summarizing we have shown the existence of the reversed $\rightarrowtail$ – derivation and $\tau_2\tau_1 = \sigma_2\sigma_1 = \sigma_1 * \tau_1$. ■

We can summarize the situation of this lemma in the following diagram which commutes.



The last lemma can also be used in a negative direction. Given a $\rightarrowtail$ – derivation $s_0 \rightarrowtail_{[\pi 1, k1, \sigma 1]} s_1$ and an occurrence π_2 independent from π_1. If s_0/π_2 is not unifiable with the left side l of a rule then s_1/π_2 is not unifiable with the left side l of that rule. That means that the non-unifiability can be inherited. This fact can be used in an implementation of the universal unification algorithm.

In the next chapter we will apply these results to the equational theory of idempotence.

4. The Theory of Idempotence

The first paper on unification under idempotence was published by P. Raulefs and J. Siekmann [RS 78]. Before giving their own algorithm they proposed to study the relation between unification and rewriting referring to [La 77]. Since the idempotence law I = {f(x x) = x} can be directed into a canonical term rewriting system R_I = {f(x x) ⇒ x}, we can apply the results on the universal unification procedure. Moreover using the termination criterion of Hullot [Hl 80] we have found a complete unification algorithm for idempotence. Since this algorithm is not minimal we tried to find a minimal algorithm for idempotence. Of course we could use the results of [SS 81] [Sz 82], but testing the minimality at each node of the narrowing tree would be more expensive than minimizing the returned set of redundant unifiers. Before we will improve the universal algorithm for the special theory of idempotence we will discuss the algorithm presented by Raulefs and Siekmann since their work was a starting point for our algorithm.

The algorithm they suggested was designed for only one idempotent function symbol f, constants and variables. It is split up into two interlocking parts: the collapsing phase and the R-unification phase. The collapsing phase is the same as narrowing with the idempotence rule on the original terms. The R-unification process unifying the collapsed terms differs from standard Robinson unification in unifying compound terms with constants and variables. In case of unifying terms with constants all variables in the term are replaced by that constant if the term contains at most that constant, if not both terms are not R-unifiable. In case of unifying a term t with a variable x they distinguish two cases: if the variable x does not occur in the term t then the usual unifier is returned, in the other case a subtree q is searched where the variable x does not occur but the brother of this subtree is an occurrence of the variable x. If {x ← q}t $=_I$ q the returned unifier is {x ← q} else both terms are not R-unifiable. As the universal unification algorithm shows it is superfluous to consider these additional subcases and the returned unifiers are redundant. Siekmann and Raulefs then observed that their algorithm is not minimal. To come closer to minimality they proposed to collapse only on hot nodes, i.e. on those nodes in the tree representation of the original terms such that the node of the opposite term is also not a leaf node. But then the modified algorithm is no longer complete: Consider s = x and t = f(f(f(a b) x) f(f(a y) x)) then there are no hot nodes and s and t are not R-unifiable since neither {x ← f(a b)} nor {x ← f(a y)} unify s and t; but s and t are infact I-unifiable with {x ← f(a b), y ← b}.

5. An algorithm for Idempotent Unification

We now want to refine the universal unification algorithm for idempotent functions. We consider a family of function symbols consisting of denumerable many constants, of a finite set of binary idempotent function symbols $\mathbb{F}_I$ and a finite set of arbitrary free function symbols $\mathbb{F}_\emptyset$, i.e. for those function symbols no equational theory is defined. Then the equational theory I is defined as

$$I = \{ f(x\ x) = x \mid f \in \mathbb{F}_I \}$$

and the corresponding canonical term rewriting system is

$$R_I = \{ f(x\ x) \Rightarrow x \mid f \in \mathbb{F}_I \}.$$

We assume that there is a binary function symbol h different from the symbols in $\mathbb{F}_I \cup \mathbb{F}_\emptyset$. If in the examples only one idempotent function symbol occurs we sometimes omit that function symbol and write (s t) for f(s t). Applying Theorem 2.3 and the results of chapter 3 we get a first complete unification algorithm for idempotence.

<u>FUNCTION</u> I-UNIFY

<u>INPUT:</u> Two arbitrary terms s and t in normal form

Enumerate the narrowing tree $\mathbb{N}_I(h(s\ t))$ with basic left-to-right strategy.

<u>OUTPUT:</u> The set $\Pi\Sigma_I(s, t)$ of unifiers of s and t away from $Z \supseteq \mathbb{V}(s, t)$

<u>ENDOF</u> I-UNIFY

We will not show the details of basing the unifiers on the set of variables of the original terms. Moreover we always restrict the domain of the narrowers to the variables in the term to which the rule is applied to, i.e. the new variable x of f(x x) is always omitted. Hence the narrower is the most general unifier of the right and left subtree of the corresponding subterm.

We are now looking for special criteria to confine the narrowing tree in the case of idempotence. A sufficient condition for stopping the enumeration process at a node h(s' t') is the fact $U\Sigma_I(s', t') = \emptyset$. A quick test for non-unifiablitiy under idempotence is to check whether the two terms start or end with different constants, i.e. in the tree representation of s' the first or last leaf is a constant and different from that of t'. We will not show this condition but a more general one.

Towards this end we need some definitions and notation. For a term s we define the corresponding argument list as the list whose toplevel elements are the subterms starting with non-idempotent function symbols, i.e. we neglect the term structure and the idempotent function symbols. The subterms starting with a non-idempotent function symbol remain unchanged. For example let s = f(a f(a f(f'(a g(b x)) f(f(z a) c))))) where f and f' are idempotent function symbols and g is an uninterpreted function symbol then the corresponding argument list is (a, a, a, g(b x), z, a, c). Given two normalized terms s and t and the corresponding argument lists $(s_1, s_2, ..., s_n)$ and $(t_1, t_2, ..., t_m)$ we define the normalized disagreement pair of s and t $d_N(s, t) = (s', t')$ to be the first disagreement pair of s_i and t_i [Ro 65] such that $s_j = t_j$, and $s_j, t_j \notin V$ for all $1 \leq j < i$. If such a pair does not exist, i.e. both argument lists are equal or one argument list is a sublist of the other, we define $d_N(s, t) = (s_n, t_n)$ if n = m and $d_N(s, t) = (c, t_{n+1})$ (w.l.o.g. we assume $n \leq m$) where c is a constant that does not occur in s and t. If one of the terms s and t is not normalized we have to add an additional condition to the above definition. We define the not normalized disagreement pair of s and t $d_{NN}(s, t) = (s', t')$ to be the first disagreement pair of s_i and t_i such that $s_j = t_j$, $s_j \notin \{s_1, ..., s_{j-1}\}$, $t_j \notin \{t_1, ..., t_{j-1}\}$ and $s_j, t_j \notin V$ for all $1 \leq j < i$. For the exception cases we take the above definition. The idea behind this last definition is that the idempotence rule is not applicable in the constant part of s and t. We want to illustrate the above definitions by some examples: let as above s = f(a f(a f(f'(a g(b x)) f(f(z a) c))))) and t_1 = f(a f(a f'(f(a g(a x)) f(f(z a) c))))) then $d_N(s, t_1) = (b, a)$ and $d_{NN}(s, t_1) = (a, a)$; for s and t_2 = f(a f(a f(x y))) we have $d_N(s, t_2) = (a, x)$ and $d_{NN}(s, t_2) = (a, a)$.

5.1 Lemma: (i) If s and t are normalized and $d_N(s, t) = (s', t')$ such that both s' and t' are non-variable terms that start with different uninterpreted constants or function symbols then $U\Sigma_I(s, t) = \emptyset$.

(ii) If s and t are not normalized and $d_{NN}(s, t) = (s', t')$ such that both s' and t' are different non-variable terms that start with different uninterpreted constants or function symbols then $U\Sigma_I(s, t) = \emptyset$.

Proof: We only show part (i), the proof of (ii) is analogous to the first part by the above remark. Suppose s and t are unifiable with unifier σ. Since s' and t' are different, both argument lists can not be equal. Let (s', t') be the disagreement pair of s_i and t_i then σs and its normal form ||σs|| start with the same argument list since there are no variables in $(s_1, ..., s_{i-1})$ and the idempotence rule is not applicable in this part of s. But since ||σs|| = ||σt|| we have $\sigma s_i = \sigma t_i$ which is a contradiction to the fact that the disagreement pair of s and t is a pair of different non-variable terms. ■

5.2 Corollary: Lemma 5.1 remains true if we replace the argument lists of s and t by the reversed argument lists of s and t.

This lemma and its corollary can be regarded as an extended clash criterion for idempotence. As point of reference we define a first stopping criterion:

(S_1) h(s' t') $\in N_I$(h(s t)) and Lemma 5.1 or the Corollary 5.2 are applicable.

As we have seen in Proposition 3.3 we can always restrict ourselves to normalized narrowers and hence as soon as the collected narrower is no longer normalized we can stop the enumeration process at that node:

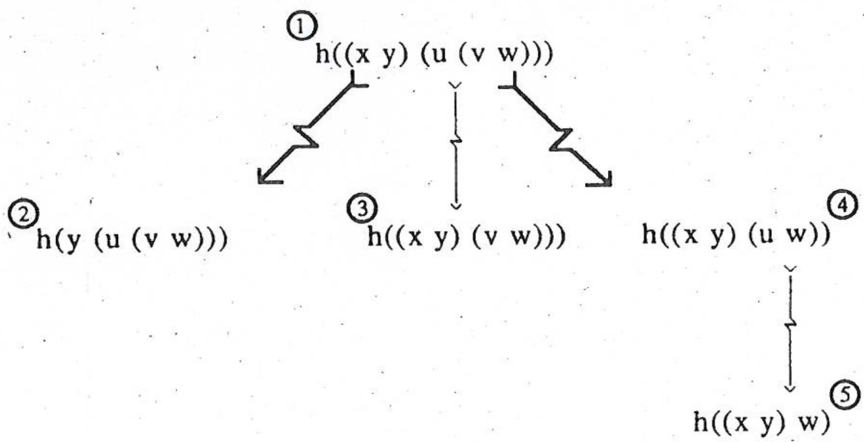
(S_2) h(s' t') $\in N_I$(h(s t)) and h(s t) $\rightarrowtail^{*}_{\theta}$ h(s' t') and θ not normalized.

If a node h(s' t') $\in N_I$(h(s t)) is reached where s' is a variable and s' $\notin$ V(t') then the only most general unifier is {s' $\leftarrow$ t'}.Hence we can formulate a third stopping criterion

(S_3) h(x t') or h(t' x) $\in N_I$(h(s t)) and x $\notin$ V(t')

Another method to diminish the costs of enumerating the tree is to find nodes h(s' t') $\in N_I$(h(s t)) where we must not perform Robinson unification for s' and t', i.e. we are sure that the unifier generated at that node is redundant. But we are not allowed to stop the enumeration process at the node h(s' t') since non-redundant unifiers may be generated in the subtree starting at h(s' t'). Consider the following example <(x y) $=_I$ (u (v w))>.

We show a part of the narrowing tree which generates a complete set of unifiers.



Node	narrower	Robinson-unifier	I-unifier
1: h((x y) (u (v w)))	ε	$\{x \leftarrow u, y \leftarrow (v\ w)\}$	$\sigma_1 = \{x \leftarrow u, y \leftarrow (v\ w)\}$
2: h(y (u (v w)))	$\{x \leftarrow y\}$	$\{y \leftarrow (u\ (v\ w))\}$	$\sigma_2 = \{x \leftarrow (u\ (v\ w)), y \leftarrow (u\ (v\ w))\}$
3: h((x y) (v w))	$\{u \leftarrow (v\ w)\}$	$\{x \leftarrow v, y \leftarrow w\}$	$\sigma_3 = \{u \leftarrow (v\ w), x \leftarrow v, y \leftarrow w\}$
4: h((x y) (u w))	$\{v \leftarrow w\}$	$\{x \leftarrow u, y \leftarrow w\}$	$\sigma_4 = \{v \leftarrow w, x \leftarrow u, y \leftarrow w\}$
5: h((x y) w)	$\{v \leftarrow w, u \leftarrow w\}$	$\{w \leftarrow (x\ y)\}$	$\sigma_5 = \{u \leftarrow (x\ y), v \leftarrow (x\ y), w \leftarrow (x\ y)\}$

We applied the second stop criterion (S_3) to node 2. Remark that $\sigma_4 = \{v \leftarrow w\}\sigma_1$. Hence we have $\mu U\Sigma_I(s, t) = \{\sigma_1, \sigma_2, \sigma_3, \sigma_5\}$. We will now generalize the situation at node 4. If we look at the subterm in (u (v w)), which is opposite to the subterm we have narrowed at, we see that this subterm is a variable occurring only once in both terms. Exactly if $h(s'\ t') \in N_I(h(s\ t))$ and there exist occurrences $1.\pi, 2.\pi \in D(h(s'\ t'))$ with $s'/\pi = z$ (resp. $t'/\pi = z$), and z occurs only once in h(s' t'), and $h(s'\ t') \rightarrowtail_{[2.\pi']} h(s''\ t'')$ (resp. $h(s'\ t') \rightarrowtail_{[1.\pi']} h(s''\ t'')$), and $\pi' = \pi.\pi''$, i.e. we narrow in a subtree, which is opposite to a variable occurring only once in both terms, then the unifier generated at the node h(s'' t'') is an instance of the unifier generated at the node h(s' t') provided that s'' and t'' are Robinson unifiable. Hence we need not perform the Robinson unification of s'' and t''. We call such a criterion a non-evaluation condition.

(NE_1) $h(s\ t) \rightarrowtail_{[2.\pi']} h(s'\ t')$ (resp. $h(s\ t) \rightarrowtail_{[1.\pi']} h(s'\ t')$) and $\pi' = \pi.\pi''$ with $1.\pi, 2.\pi \in D(h(s'\ t'))$ and $s/\pi = z$ (resp. $t/\pi = z$), z occurs only once in h(s t)

This condition is justified by the following lemma which holds also for arbitrary term rewriting systems:

5.3 Lemma: Let $h(s\ t) \rightarrowtail_{[2.\pi',\sigma]} h(s'\ t')$ (resp. $h(s\ t) \rightarrowtail_{[1.\pi',\sigma]} h(s'\ t')$) and $\pi' = \pi.\pi''$ with $1.\pi, 2.\pi \in D(h(s\ t))$ and $s/\pi = z$ (resp. $t/\pi = z$) and z occurs only once in h(s t). If β is a most general Robinson unifier of s and t and β' a most general Robinson unifier of s' and t' then $\beta'\sigma \leq_I \beta\ [V(s, t)]$. If s and t are not Robinson unifiable then s' and t' are not Robinson-unifiable.

Proof: Let $t/\pi = r$, $t'/\pi = r'$ and $h(s''\ t'') = \sigma h(s\ t)$ where σ is a most general Robinson unifier of t/π' and f(u u) where u is a new variable not occurring in s and t. Then $h(s''\ t'') \longrightarrow_{[2.\pi']} h(s'\ t')$. We now define β'' by $\beta''x = \beta'x$ for $x \neq z$ and $\beta''z = \beta'\sigma r =_I \beta'r' = \beta'z$, hence $\beta'' =_I \beta'$. Now β'' is a unifier of s'' and t'' since the rewrite rule is applied to a subterm which is opposite to the variable z, which occurs only once in s and t, and hence we have $\beta''\sigma \leq \beta\ [V(s, t)]$. So finally $\beta'\sigma \leq_I \beta\ [V(s, t)]$. Suppose s' and t' are Robinson unifiable and s and t are not. Then we construct the Robinson unifier β'' of s'' and t'' as above. Hence $\beta''\sigma$ is a unifier of s and t which is a contradiction. ■

Given the narrowing sequence $h(s\ t) \rightarrowtail_{[1.\pi]} h(s_1\ t_1) \rightarrowtail_{[2.\pi]} h(s_2\ t_2)$ we will show that the I-unifier generated at $h(s_2\ t_2)$ is an I-instance of the Robinson unifier of s and t.

5.4 Lemma: Let $h(s\ t) \rightarrowtail_{[1.\pi,\sigma 1]} h(s_1\ t_1) \rightarrowtail_{[2.\pi,\sigma 2]} h(s_2\ t_2)$, β the most general Robinson unifier of s and t and β_2 the most general Robinson unifier of s_2 and t_2 then $\beta_2\sigma_2\sigma_1 \leq_I \beta\ [\mathbb{V}(s, t)]$. If s/π and t/π start with the same function symbol and if s and t are not Robinson unifiable then s_2 and t_2 are not Robinson unifiable.

Proof: Let $s/\pi = f(p_1\ p_2)$ and $t/\pi = f(q_1\ q_2)$ then σ_1 is a Robinson unifier of p_1 and p_2 and σ_2 is a Robinson unifier of $\sigma_1 q_1$ and $\sigma_1 q_2$. Then $\sigma_2\sigma_1$ is a unifier of p_1 and p_2 and of q_1 and q_2. Now β_2 unifies s_2 and t_2 and since $s_2/\pi = \sigma_2\sigma_1 p_1 = \sigma_2\sigma_1 p_2$ and $t_2/\pi = \sigma_2\sigma_1 q_1 = \sigma_2\sigma_1 q_2$ we obtain that $\beta_2\sigma_2\sigma_1$ is a unifier of p_1, p_2, q_1 and q_2 and hence of $s/\pi = f(p_1\ p_2)$ and $t/\pi = f(q_1\ q_2)$. Since $\sigma_2\sigma_1 s$ resp. $\sigma_2\sigma_1 t$ only differs from s_2 resp. t_2 at the occurrence π $\beta_2\sigma_2\sigma_1$ is a unifier of s and t and therefore $\beta_2\sigma_2\sigma_1 \leq_I \beta\ [\mathbb{V}(s, t)]$.
Suppose s_2 and t_2 are unifiable with unifer β_2 then by the above $\beta_2\sigma_2\sigma_1$ is a unifier of s and t which is a contradiction. ■

We can generalize this lemma to a non-evaluation criterion which we call the parallel path condition

(NE_2) Given a left-to-right derivation

$$h(s\ t) \rightarrowtail_{[\pi 1]} h(s_1\ t_1) \rightarrowtail_{[\pi 2]} \dots \rightarrowtail_{[\pi n]} h(s_n\ t_n)$$

where $n = 2m$ and $\pi_i = 1.\pi'_i$ and $\pi_{m+i} = 2.\pi'_i$ for $1 \leq i \leq m$

The parallel path condition is shown by an induction argument. By Lemma 3.5 we can reorder the above derivation to

$$h(s\ t) \rightarrowtail_{[\pi 1]} h(s_1\ t_1) \rightarrowtail_{[\pi m+1]} h(s_2'\ t_2') \rightarrowtail_{[\pi 2]} \dots \rightarrowtail_{[\pi 2m]} h(s_n\ t_n).$$

Now Lemma 2.5 implies that the unifier generated at the node $h(s_n\ t_n)$ is an instance of the most general Robinson unifier of s and t. The idea behind the proofs is to use the fact that a most general Robinson unifier of a left side of a rule f(x x) and a given subterm t is a most general Robinson unifier of the left and right subtree of t if it is restricted to the variables of t.

To use both non-evalution criteria during the enumeration process we need the following lemma:

5.5. Lemma: Let $h(s,t) >\!\!\rightsquigarrow^{*}_{\theta_1} h(s_1, t_1) >\!\!\rightsquigarrow^{*}_{\theta_2} h(s_2, t_2)$ with the Robinson unifiers β, β_1 and β_2. If $\beta_2\theta_2 \leq_I \beta_1 \; [V(s_1, t_1)]$ then $\beta_2\theta_2\theta_1 \leq_I \beta_1\theta_1 \; [V(s, t)]$.

Proof: If the unifiers are properly renamed the proof is straight forward. ∎

A combination of both criteria is possible. Consider the problem

$$< ((x\ y)\ (((a\ b)\ c)\ z)) =_I ((u\ (v\ w))\ (r\ (a\ b))) >$$

and the following basic left-to-right derivation:

$$H_1 >\!\!\rightsquigarrow_{[1.1]} H_2 >\!\!\rightsquigarrow_{[1]} H_3 >\!\!\rightsquigarrow_{[2.1.2]} H_4 >\!\!\rightsquigarrow_{[2.1]} H_5 >\!\!\rightsquigarrow_{[2]} H_6$$

with

Node	I-unifier
H_1	h(((x y) (((a b) c) z)) ((u (v w)) (r (a b)))
	{x ← x', y ← (v' w'), z ← (a b), u ← x', v ← v', w ← w', r ← ((a b) c)}
H_2	h((y (((a b) c) z)) ((u (v w)) (r (a b)))
	{x ← (u' (v' w')), y ← (u' (v' w')), z ← (a b), u ← u', v ← v', w ← w', r ← ((a b) c)}
H_3	h((((a b) c) z) ((u (v w)) (r (a b)))
H_6	h((((a b) c) z) (r (a b)))
	{x ← (((a b) c) (a b)), y ← (((a b) c) (a b)), z ← (a b), u ← (((a b) c) (a b)), v ← (((a b) c) (a b)), w ← (((a b) c) (a b)), r ← ((a b) c)}

The unifiers generated at H_1, H_2, H_4, H_5 are most general and the terms s_3 and t_3 at $H = h(s_3\ t_3)$ are not Robinson unifiable. The unifier $\beta_6\theta_6$ generated at node H_6 is an instance of $\beta_2\theta_2$, generated at H_2, and of β_1, generated at H_1:

Regard the following derivation, which is a reorder of the first:

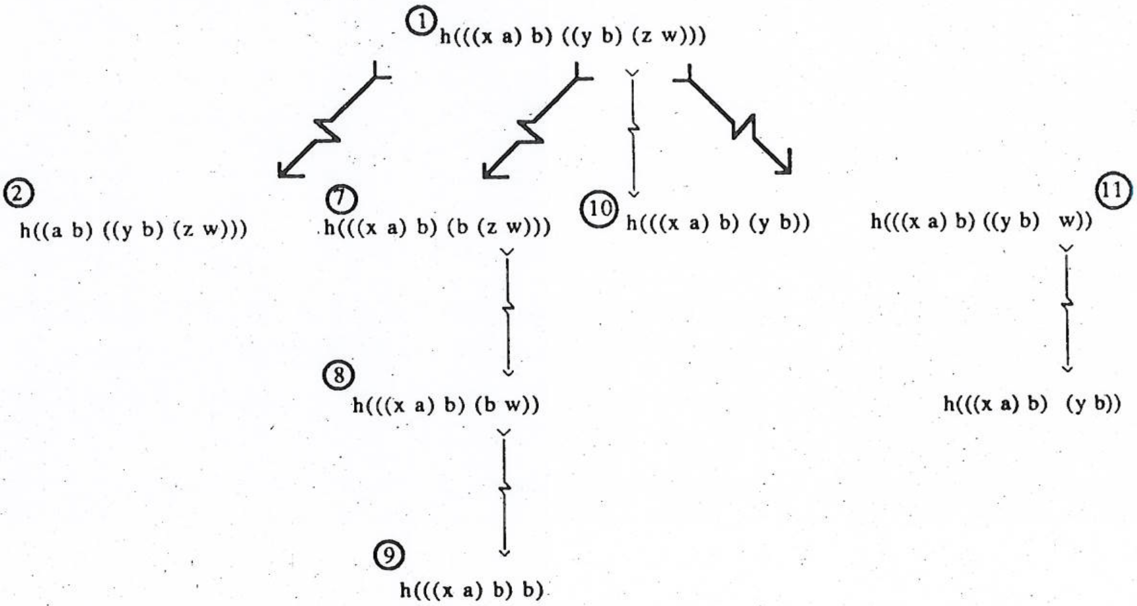
$$H_1 >\!\!\rightsquigarrow_{[2.1.2]} H_2' >\!\!\rightsquigarrow_{[1.1]} H_3' >\!\!\rightsquigarrow_{[1]} H_4 >\!\!\rightsquigarrow_{[2.1]} H_5 >\!\!\rightsquigarrow_{[2]} H_6$$

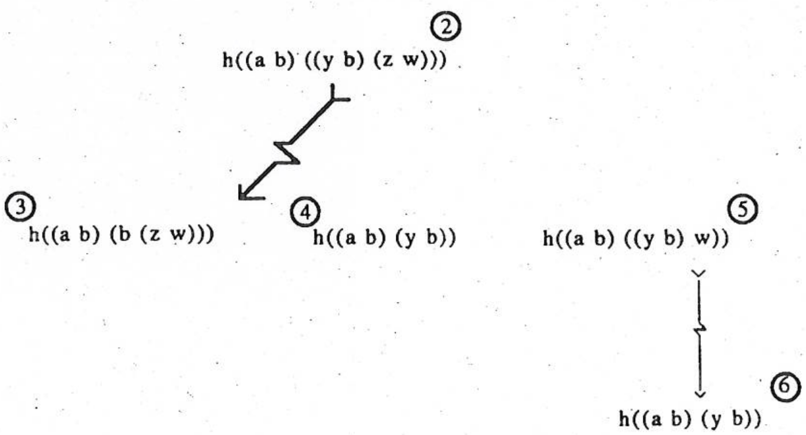
then $\beta_6\theta_6$ is an instance of $\beta_2'\theta_2'$, generated at node H_2', by (NE_2) and $\beta_2'\theta_2'$ is in turn an instance of β_1 by (NE_1) since $H_1/1.1.2 = y$ occurs only once in H_1. If we consider the derivation

$$H_1 >\!\!\rightsquigarrow_{[1.1]} H_2 >\!\!\rightsquigarrow_{[2.1.2]} H_3'' >\!\!\rightsquigarrow_{[2.1]} H_4'' >\!\!\rightsquigarrow_{[1]} H_5 >\!\!\rightsquigarrow_{[2]} H_6$$

then $\beta_6\theta_6$ is an instance of $\beta_4''\theta_4''$, generated at node H_4'', by (NE_2). By (NE_1) $\beta_4''\theta_4''$ is an instance of $\beta_3''\theta_3''$, which in turn by (NE_1) is an instance of $\beta_2\theta_2$ since $H_2/1.1 = H_3''/1.1 = y$ occurs only once in H_2 resp. H_3.

Before we give a final version of the idempotent unification algorithm we will discuss the strategy running through the derivation tree. Of course there are the two possiblities for a complete enumeration: depth–first and breadth–first. We first consider another example: < ((x a) b) $=_I$ ((y b) (z w)) >.





Node	narrower	Robinson unifier	I-unifier
4 h((a b) (y b))	{x ← a, z ← y, w ← b}	{y ← a}	{x ← a, z ← a, w ← b, y ← a}
6 h((a b) (y b))	{x ← a, z ← (y b), w ← (y b)}	{y ← a}	{x ← a, z ← (a b), w ← (a b), y ← a}
10 h((x a) b) (y b))	{z ← y, w ← b}	{y ← (x a)}	{z ← (x a), w ← b, y ← (x a)}
12 h((x a) b) (y b))	{z ← (y b), w ← (y b)}	{y ← (x a)}	{z ← ((x a) b), w ← ((x a) b), y ← (x a)}

To ease the notation we write $h(s_k\ t_k)$ for the term at node k, for the narrower collected so far θ_k, for the Robinson-unifier β_k and σ_k for the I-unifier generated at node k. At the nodes 1, 2, 3, 5, 7, 8, 9 and 11 we do not obtain unifiers caused by non-unifiability of s_k and t_k. At node 3 we can apply the stop criterion (S_1), by which two unnecessary narrowing steps can be avoided.

If we choose a depth-first strategy we first generate the unifiers σ_4 and σ_6. But to get a minimal set of unifiers we have to reject them after generating σ_{10} and σ_{12} since σ_4 is an instance of σ_{10} and σ_6 is an instance of σ_{12}. Using a right-to-left strategy we would find the instances without an instance test since for the derivations $h(s_{10}\ t_{10}) \succ\!\!\!-\!\!\!\not\!\!-\!\!\!\rightarrow_{[1.1]} h(s_4\ t_4)$ and $h(s_{12}\ t_{12}) \succ\!\!\!-\!\!\!\not\!\!-\!\!\!\rightarrow_{[1.1]} h(s_6\ t_6)$ the non evaluation criterion (NE_1) is applicable. But of course there are examples for which a right-to-left strategy will not work (take the above example reversed $<((y\ b)\ (z\ w)) =_I ((x\ a)\ b)>$). We did not find a criterion to decide which strategy works best for which terms.

But if we look at the derivation length of the unifiers in the above example we see that the length of instances is larger than the length of more general unifiers. We did not find either a proof or a counterexample for this phenomenon, so we state it as an open conjecture:

If $\sigma_i \leq_I \sigma_j$ then the derivation length of σ_j is smaller than that of σ_i.

But this is the key idea of the minimal universal unification algorithm of Siekmann and Szabó [SS 81], [Sz 82] who give a decision algorithm to test at any point in the narrowing tree whether a more general unifier than the just generated will be found later in the unification process. Hence a proof of the above conjecture would make this decision procedure absolete in the case of idempotence and a minimal unification algorithm were found. This observation and the possibility to find a reordering of the narrowing steps such that we can apply the conditions (NE_1) and (NE_2) leads us to use a breadth-first search in the narrowing tree.

We will state an algorithm which is a not minimal but incorporates the results collected so far:

<u>FUNCTION</u> I-UNIFY

<u>INPUT:</u> Two arbitrary terms s and t in normal form

Enumerate the narrowing tree $N_I(h(s\ t))$ with basic, left-to-right and breadth-first strategy.
At every node try

(i) to apply the stop criteria (S_1) to (S_3)

(ii) to find a reorder of the narrowing steps such that the non evaluation criteria (NE_1) and (NE_2) are applicable

<u>OUTPUT:</u> The set $\Pi\Sigma_I(s, t)$ of unifiers of s and t away from $Z \supseteq V(s, t)$

<u>ENDOF</u> I-UNIFY.

6. Conclusion

We have presented a complete unification algorithm for idempotent functions. The algorithm is based on the universal unification algorithm as described by Hullot [Hl 80] and Kirchner [Ki 85]. This algorithm uses narrowing with rewrite rules as its central computation rule. We have improved on ordinary narrowing by using special derivation strategies and by giving some conditions where the enumeration of the narrowing tree can be stopped.

We did not succeed in finding a minimal unification algorithm for idempotent function symbols. Using the results of Siekmann and Szabo [SS 81], [Sz 82] the universal unification algorithm can be extended to be minimal. This extension involves at each node of the narrowing tree computing a complete set of solutions of a matching problem which is a very expensive operation. But since the narrowing tree is always finite the theory of idempotence is finitary unifying and hence the redundant unifiers can be eliminated by minimizing the returned set of unifiers. This minimizing step only involves a decision test for a matching problem which is less expensive.

We have shown conditions which prevent the generation of redundant unifiers. We called them non-evaluation conditions since we were able to show that on certain nodes of the narrowing tree only redundant unifiers were generated. These conditions only depend on information that is known. Another possibility to reduce the minimizing costs after the enumeration of the narrowing tree is to prevent the generation of T-equal unifiers, i.e. not normalized unifiers. Since the narrowers are always normalized (confer stop condition (S_2)) there remain two possibilities. First the Robinson unifier b may not be in normal form and hence the I-unifier $\beta\theta$ is not. But excluding such unifiers results in an uncompatibility with the non-evaluation condition (NE_1). Consider the following example < ((x a) b) $=_I$ (((y b) a) y) > then the Robinson unifier $\beta = \{x \leftarrow (b\ b), y \leftarrow b\}$ is not normalized. The only succesful narrowing derivation h(((x a) b) (((y b) a) y)) $\succ\!\!-\!\!\!\rightarrow_{[2.1.1,\ \sigma = \{y \leftarrow b\}]}$ h(((x a) b) ((b a) y))
yields the I-unifier $\beta'\sigma = \{x \leftarrow b, y \leftarrow b\}$. But the non-evaluation condition can be applied to this derivation eliminating the generation of b's. Hence combining both ideas results in incompleteness since both terms from the example are I-unifiable but no unifier is returned. With the elimination of not normalized I-unifiers bh where both b and h are normalized we get analoguous difficulties.

The original version of the universal unification algorithm was not based on ordinary narrowing but on superreduction, i.e. after each narrowing step the newly generated terms are normalized as described in [La 75], [Fy 79], [RK 85] and [Ki 85]. But there are several compatibility problems with superreduction. First superreduction does not fit into the concept of non-evaluation conditions: Consider the example < ((a x) (x a)) $=_I$ (u a) > then the superreduction derivation h(((a x) (x a) (u a)) $\succ\!\!-\!\!\!\rightarrow_{[1]}$ h(a, (u a)) $\succ\!\!-\!\!\!\rightarrow_{[2]}$ h(a a) is possible, where ((a x) (x a)) and (u a) are not Robinson unifiable and a and a are. This contradicts the parallel path condition. The same problem arises with the given strategies: basic and left-to-right derivations are not compatible with superreduction because the reduction steps are performed at occurrences that are not in the subsets U_i of occurrences defined by the strategies. So we have to perform superfluous reductions using the concept of superreduction. Beside that if the terms at the nodes of the narrowing tree are always normalized we loose the pruning effect of the given strategies.

7. References:

[Ba 86] Baader,F., The Theory Of Idempotent Semigroups Is Of Unification Type Zero', Journal of Automated Reasoning, Vol. 2, No. 3, 283-286, (1986)

[Bi 35] Birkhoff, G., 'On The Structure Of Abstract Algebra', Proc. Cambrigde Phil. Soc., Vol. 31, 433-454, (1935)

[BS 81] Burris, S. and Sankappanavar, H.P., 'A Course In Universal Algebra', Springer-Verlag, (1981)

[Bu 85] Buchberger, B., 'Basic Features And Developments Of The Critical-Pair/Completion Procedure', Proc. of 'Rewriting Techiques and Applications' (ed J.-P. Jouannaud), Springer-Verlag, LNCS 202, 1-45, (1985)

[FH 83] Fages, F. and Huet, G., 'Unification And Matching In Equational Theories', Proc. of CAAP'83 (ed. G. Ausiello and M. Protasi), Springer-Verlag, LNCS 159, 205-220, (1983)

[Fy 79] Fay,M., 'First Order Unification In An Equational Theory', Proc.of 4th Workshop on Automated Deduction', 161-167, Texas, (1979)

[GT 78] Goguen, J. A., Thatcher, J.W. and Wagner, E. G., 'An Initial Algebra Approach To The Specification, Correctness And Implementation Of Abstract Data Types', in 'Current Trends in Programming Methodology, Vol.4, Data Structuring' (ed. R. T. Yeh), Prentice Hall, (1978)

[Gr 79] Grätzer, G., 'Universal Algebra', Springer-Verlag, (1979)

[He 82] Herold, A., 'Universal Unification And A Class Of Equational Theories', Proc. of GWAI-82 (ed. W.Wahlster),Springer-Verlag, IFB-82, 177-190, (1982)

[He 83] Herold, A., 'Some Basic Notions Of First-Order Unifcation Theory'. MEMO SEKI-VIII-KL, Universität Karlsruhe, (1983)

[He 85] Herold, A., 'A Combination Of Unification Algorithms', MEMO SEKI-85-VIII-KL, Universität Kaiserslautern, (1985)

[Hl 80] Hullot,J.M.,'Canonical Forms And Unification', Proc. of 5th CADE (eds. W.Bibel and R.Kowalski), Springer-Verlag, LNCS 87, 318-334, (1985)

[Hl 80a] Hullot,J.M.,'A Catalogue Of Canonical Term Rewriting Systems', Technical Report CSL-113, SRI International, (1980)

[Hl 80b] Hullot,J.M.,'Compilation De Formes Canoniques Dans Des Théories Equationalles Thèse du 3ème Cycle, Université de Paris-Sud, (1980)

[HO 80] Huet, G. and Oppen, D. C., 'Equations And Rewrite Rules: A Survey', in 'Formal Languages: Perspectives and Open Problems (ed R. Book), Academic Press, (1980)

[Hu 76] Huet, G., 'Résolution d'équations dans des langages d'ordre 1, 2 ... ω', Thèse de doctorat d'état, Université Paris VII, (1976) [JK 83] Jouannaud,J.-P., Kirchner, C. and Kirchner, H., 'Incremental Construction Of Unifcation Algorithms In Equational Theories', Proc. of 10^{th} ICALP (ed J.Diaz),LNCS 154,361-373 (1983)

[KB 70] Knuth, D.E. and Bendix, P.B., 'Simple Word Problems In Universal Algebras', in 'Computational Problems In Abstract Algebras' (ed. J. Leech), Pergamon Press, 263-297, (1970)

[Ki 85] Kirchner, C., 'Methodes Et Outils De Conception Systematique D'Algorithmes D'Unification Dans Les Théories Equationelles', Thèse de doctorat d'état, (in French) Université de Nancy 1, (1985)

[La 75] Lankford, D.S., 'Canonical Inference', Report ATP-32, University of Texas, Austin, (1975)

[La 77] Lankford, D.S., 'Complete Sets Of Reductions', Report ATP-35, ATP-37, ATP-39, University of Texas, Austin, (1977)

[Lo 78] Loveland, D., 'Automated Theorem Proving', North-Holland, (1978)

[Pl 72] Plotkin, G., 'Building-In Equational Theories', Machine Intelligence 7, 73-90, (1972)

[RK 85] Réty,P., Kirchner, C., Kirchner, H. and Lescanne, P. 'Narrower: A New Algorithm For Unifcation And Its Application To Logic Programming', Proc. of 'Rewriting Techiques and Applications' (ed J.-P. Jouannaud), Springer-Verlag, LNCS 202, 141-157, (1985)

[Ro 65] Robinson, J. A., 'A Machine-Oriented Logic Based On The Resolution Principle', JACM 12, Nº. 1, 23-41, (1965)

[RS 78] Raulefs,P. and Siekmann, J., 'Unifcation Of Idempotent Functions', MEMO SEKI-78-II-KL, Universität Karlsruhe, (1978)

[Sc 86] Schmidt-Schauß, M., 'Unification Under Associativity And Idempotence Is Of Type Nullary', Journal of Automated Reasoning, Vol. 2, No. 3, 277-282, (1986)

[Si 86] Siekmann, J., 'Universal Unification', in Proc. of 8^{th} ECAI'86, Brighton (1986)

[SS 81] Siekmann, J,. and Szabó, P.,'Universal Unification And Regular ACFM Theories', Proc.of IJCAI-81,Vancouver, (1981)

[Sz 82] Szabó, P., Unifikationstheorie Erster Ordnung, Dissertation (in German), Universität Karlsruhe, (1982)

[Ta 79] Taylor, W., 'Equational Logic', Houston Journal of Mathematics 5, (1979)

[Ti 86] Tidén, E., 'Unification In Combinations Of Equational Theories', Thesis, Stockholm, Sweden

[RW 69] Robinson, G. and Wos, L.,'Paramodulation And Theorem-Proving In First-Order Theories With Equality', Machine Intelligence 4 (eds. B.Meltzer and D.Michie),135-151, (1969)

[Ye 85] Yelick, K., 'Combining Unification Algorithms for Confined Regular Equational Theories', Proc. of 'Rewriting Techiques and Applications' (ed J.-P. Jouannaud), Springer-Verlag, LNCS 202, 365-380, (1985)