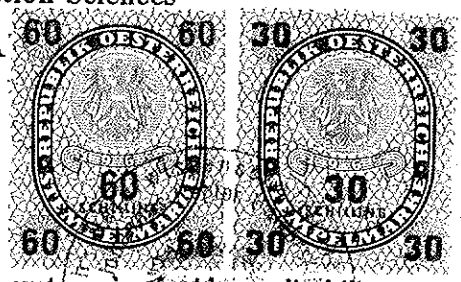
(1)

Reducing the Complexity of the Knuth-Bendix Completion Algorithm: A "Unification" of Different Approaches

Franz Winkler[1]

Department of Computer and Information Sciences
University of Delaware
Newark, DE 19716, USA



Abstract

The Knuth-Bendix completion procedure for rewrite rule systems is of wide applicability in symbolic and algebraic computation. Attempts to reduce the complexity of this completion algorithm are reported in the literature. Already in their seminal 1967 paper, D.E. Knuth and P.B. Bendix have suggested to keep all the rules interreduced during the execution of the algorithm. G. Huet has presented a version of the completion algorithm in which every rewrite rule is kept in reduced form with respect to all the other rules in the system. Borrowing an idea of Buchberger's for the completion of bases of polynomial ideals the author has proposed in 1983 a criterion for detecting "unnecessary" critical pairs. If a critical pair is recognized as unnecessary then one need not apply the costly process of computing normal forms to it. It has been unclear whether these approaches can be combined. We demonstrate that it is possible to keep all the rewrite rules interreduced and still use a criterion for eliminating unnecessary critical pairs.

1. Introduction

We assume familiarity with the basic notions of the theory of term rewriting systems and in particular with the Knuth-Bendix completion algorithm (see, for instance, [KB67], [Hu80]). The Knuth-Bendix completion algorithm (if it terminates successfully) solves the following problem:

given: a (finite) set of equations E over some term algebra T

find: a set of rewrite rules (directed equations) R over T such that

(C)

$=_E = \leftarrow\rightarrow_R^*$,

$\rightarrow_R$ is noetherian, and

$\rightarrow_R$ has the Church-Rosser property.

$=_E$ is the equational congruence over T generated by the equations in E. $\rightarrow_R$ denotes the reducibility in one step with respect to the rewrite rule system R , i.e. $s \rightarrow_R t$ iff there is a rule $u \rightarrow u'$ in R, such that for some occurrence p in s and for some substitution σ we have $s/p = \sigma(u)$, $t = s[p := \sigma(u')]$. We write $\leftarrow_R$, $\rightarrow_R^+$, $\rightarrow_R^*$, $\leftarrow\rightarrow_R$, $\leftarrow\rightarrow_R^*$ for the inverse relation, the

[1]The author is currently on leave from Johannes Kepler University in Linz, Austria.

transitive closure, the transitive-reflexive closure, the symmetric closure, and the symmetric-transitive-reflexive closure of $\rightarrow_R$, respectively. By $\trianglelefteq$ we denote the subsumption preorder (see [Hu80]).

$\rightarrow_R$ has the Church-Rosser property iff for all terms s and t: if $s \overset{*}{\longleftrightarrow}_R t$ then there is a term u such that $s \overset{*}{\rightarrow}_R u \overset{*}{\leftarrow}_R t$. The Church-Rosser property is equivalent to confluence and if $\rightarrow_R$ is noetherian then confluence is equivalent to local confluence. If $\rightarrow_R$ has the Church-Rosser property then we also say that R is a complete rewrite rule system.

Once we have a complete noetherian rewrite rule system R that solves the problem (C) for the given system of equations E, then we can decide $=_E$: $s =_E t$ if and only if s and t have the same normal form with respect to $\rightarrow_R$. s´ is a normal form of s with respect to $\rightarrow_R$ iff $s \overset{*}{\rightarrow}_R s'$ and s´ is irreducible with respect to $\rightarrow_R$.

The Knuth-Bendix algorithm proceeds by computing critical pairs of rewrite rules, reducing them to normal forms and checking these normal forms for syntactical equality. If the reduction results of a critical pair turn out to be different then they are combined to form a new rule, which is added to the rewrite system. This process might fail if the addition of such a newly derived rule destroys the noetherianity of the rewrite rule system.

Attempts to reduce the complexity of the Knuth-Bendix completion algorithm are reported in the literature. Already in their seminal paper [KB67] D.E. Knuth and P.B. Bendix have suggested to keep all the rules interreduced during the execution of the algorithm. G. Huet has presented a version of the algorithm in which every rewrite rule is kept in reduced form with respect to all the other rules in the system. A detailed correctness proof of this variant of the algorithm is given in [Hu81]. Borrowing an idea of Buchberger's [Bu79] for the completion of bases of polynomial ideals (Gröbner bases algorithm) the author has proposed in [WB83] a criterion for detecting "unnecessary" critical pairs. If a critical pair is recognized as unnecessary then we need not apply the costly process of computing normal forms to it. It has been unclear whether these approaches can be combined. In the subsequent chapters we show that this is possible.

For the completion algorithm in the case of polynomial ideals (Gröbner bases algorithm) the combination of mutual reduction and criteria for eliminating unnecessary critical pairs has been proposed by B. Buchberger in [Bu65], [Bu70], and [Bu85].

Besides being of importance in deciding equational theories, the Knuth-Bendix algorithm has also been proposed as a substitute for inductive proofs in the initial model of an equational theory by D.R. Musser [Mu80] and G. Huet and J.M. Hullot [HH80]. Hsiang [Hs82] uses the completion algorithm for refutation theorem proving. For further applications we refer to [De83].

2. The concept of connectedness and a generalized Newman lemma

In this section we describe the concept of connectedness (introduced in [WB83], see also [Wi84]) which leads to a generalized version of Newman's lemma [Ne42].

Def. 2.1: Let M be a set, $>$ a partial ordering on M, $\rightarrow$ a binary relation on M. Then for every x,y, and z in M we define: $x \overset{*}{\longleftrightarrow} (<z)\ y$ iff there is a finite sequence $u_1,\ldots,u_n$ in M such that $x=u_1 \longleftrightarrow u_2 \longleftrightarrow \cdots \longleftrightarrow u_n=y$ and $u_i < z$ for all $1 \leq i \leq n$. (Read: x and y are connected below z with respect to $>$ and $\rightarrow$.)

Lemma 2.1 (generalized Newman Lemma, [WB83]): Let $>$ be a noetherian partial ordering on the set M and $\rightarrow$ a binary relation on M such that $\rightarrow$ is contained in $>$. Then $\rightarrow$ is confluent if and only if for all x,y,z in M: if $x \leftarrow z \rightarrow y$ then $x \overset{*}{\longleftrightarrow} (<z)\ y$.

The Newman Lemma reduces the question of confluence to the question of local confluence for noetherian reduction relations (see, for instance, the chapter on algebraic simplification in [BC83]). In the case of a reduction relation which is generated by a system of rewrite rules this means that one has to search for common successors for all the critical pairs. Quite frequently it is known from computations in previous steps of the completion algorithm that the two sides of a critical pair are connected below the term from which they are derived. In such situations the generalized Newman Lemma allows to omit the search for a common successor for this critical pair. Algorithms which are based on the generalized rather than on the conventional Newman Lemma have to use mechanisms for detecting that the two sides of a critical pair are connected. The subalgorithm NCP in Section 3 serves this purpose in our version of the completion algorithm.

3. The completion algorithm

Problem:
given: E , a finite set of equations (over some term algebra T)
$>$, a reduction ordering (in the sense of [Wi84], Def. 3.19)

find: R, a (finite) set of rewrite rules such that
$=_E = \overset{*}{\longleftrightarrow}_R$,
(C′) $\rightarrow_R$ is contained in $>$, and
$\rightarrow_R$ has the Church-Rosser property.

We will consider a refined version of Huet's [Hu81] algorithm for completing a given system of rewrite rules. Instead of considering all critical pairs arising from a rule in an intermediate version R_i of the rewrite rule system, we add only "necessary" critical pairs to the set of equations E_i. The test, whether a critical pair is necessary or not, is incorporated in the subalgorithm NCP. The stated completion algorithm, contrary to the one in [Hu81], does not abort immediately if an equation s=t is encountered in E_i such that the normal forms of s and t w.r.t R_i are incomparable. We still have the option of choosing an other equation in E_i. Only when all the equations in E_i are reduced w.r.t. R_i and their left hand sides and right hand sides are incomparable the algorithm has to stop with failure.

COMPL (E, $>$)
[Completion algorithm. If the algorithm stops successfully, then its output is a rewrite rule system R which solves (C′).]

E_0 := E; R_0 := {}; i := 0; p := 0;
LOOP
  WHILE $E_i \neq$ {} DO
    IF all equations in E_i are marked THEN FAILURE
    ELSE s=t := an unmarked equation in E_i ;
      s′ := a normal form of s w.r.t. $\rightarrow_{R_i}$;
      t′ := a normal form of t w.r.t. $\rightarrow_{R_i}$;
      IF s′=t′
      THEN E_{i+1} := E_i − {s=t};
        R_{i+1} := R_i;
        the equations in E_{i+1} and the rules in R_{i+1}
        are marked as they are in E_i, R_i;
        i := i+1;

```
      ELSE IF s'>t' or t'>s'
         THEN u := max_>(s',t');
            v := min_>(s',t');
            K := set of labels k of rules in R_i whose left
                    hand side s_k is reducible by u→v, say to s_k';
            E_{i+1} := E_i - {s=t} ∪ {s_k'=t_k | k:s_k→t_k in R_i, k in K};
            all the equations in E_{i+1} are unmarked;
            p := p+1;
            R_{i+1} := {j:s_j→t_j' | j:s_j→t_j in R_i, j not in K,
               t_j' is a normal form of t_j w.r.t R_i ∪ {u→v}}
               ∪ {p:u→v};
            the rules coming from R_i are marked or unmarked
            as they were in R_i, the new rule p:u→v is unmarked;
            i := i+1;
         ELSE E_{i+1} := E_i - {s=t} ∪ {s'=t'};
            the equation s'=t' is marked;
            R_{i+1} := R_i;
            the rules in R_{i+1} are marked or unmarked as they were in R_i;
            i := i+1;
   ENDWHILE;
   IF all rules in R_i are marked THEN RETURN R_i
   ELSE k:s_k→t_k := an unmarked rule in R_i;
      E_{i+1} := NCP (R_i,k), i.e., the set of necessary critical pairs computed between rule k
         and any rule of R_i of label not greater than k;
      all equations in E_{i+1} are unmarked;
      R_{i+1} := R_i ;
      the rules in R_{i+1} are marked or unmarked as they were in R_i,
      except that rule k is marked in R_{i+1};
      i := i+1;
ENDLOOP □
```

For computing the necessary critical pairs we use the following algorithm:

```
CP ← NCP(R_i,k)
[compute set of necessary critical pairs in R_i between rule k and any
 rule of label not greater than k]
CP := {};
S := set of rules in R_i with label not greater than k;
S' := S;
j := 0;
WHILE S' ≠ {} DO
   j := j+1;
   select l:s_l→t_l in S';
   FOR p an occurrence in s_k, s_k/p not a variable, s_k/p and s_l unifiable DO
      σ := most general unifier of s_k/p and s_l;
      IF there are no m:s_m→t_m in S−S', q prefix of p, such that
         s_m ◁ σ(s_k)/q and
         ((m is marked and m≥l) or (l is marked and l≥m))
      THEN CP := CP ∪ {σ(t_k)=σ(s_k[p:=t_l])};
```

FOR p an occurrence in s_l, s_l/p not a variable, s_l/p and s_k unifiable DO
  σ := most general unifier of s_l/p and s_k;
  IF there are no $m{:}s_m \to t_m$ in S−S′, q prefix of p, such that
    $s_m \neq \sigma(s_l)/q$ and
    ((m is marked and $m \geq l$) or (l is marked and $l \geq m$))
  THEN CP := CP ∪ $\{\sigma(t_l) = \sigma(s_k[p:=t_k])\}$;
S′ := S′ − $\{l{:}s_l \to t_l\}$;
ENDWHILE;
RETURN CP □

As in [Hu81] we adopt the fairness of selection hypothesis:

> for every rule label k there is an iteration i such that either the rule of label k is deleted from R_i (i.e. k in K at iteration i), or the rule of label k is selected at "compute necessary critical pairs" (i.e. as argument for NCP).

As Huet points out, this hypothesis will usually be met in practice. Essentially it requires that the selection process takes into account the "age" of the rules.

Before we can go on and prove the correctness of this version of the completion alogrithm we have to introduce some notation (suggested in [Hu81]).

$R := \bigcup_{i \geq 0} R_i$. $R_\infty := \{k{:}s \to t \mid \exists\, i\ \forall\, j \geq i\ k{:}s \to t \text{ in } R_j\}$.

R_∞ is called the limit rewriting system.

R_∞ may be infinite. $R_\infty \subseteq R$. If the completion algorithm stops with success at iteration i, then $R_\infty = R_i$.

4. Correctness of the completion algorithm

If we suppose that the completion algorithm does not stop with failure, then we can show as in [Hu81] (corollary to Lemma 1, Lemma 3, corollary to Lemma 3, corollary to Lemma 4):

Lemma 4.1:

(a) Every term s which is reducible by R_i is also reducible by every R_j, with $j \geq i$.

(b) $\forall\, i \geq 0\ \forall\, s{=}t \text{ in } E_i\ \exists\, u$ $(s \to_R^* u \text{ and } t \to_R^* u)$.

(c) $\leftarrow\!\!\to_R^* = =_E$.

(d) R_∞ is in reduced form, i.e. no rule $s \to t$ in R_∞ is reducible by the other rules in R_∞.

Lemma 4.2: For all critical pairs $\sigma(t_k)$, $\sigma(s_k[p:=t_l])$ derived from rules in R we have

$\sigma(t_k) \leftarrow\!\!\to_R^* (<\sigma(s_k))\ \sigma(s_k[p:=t_l])$.

Proof:
We prove the lemma by noetherian induction on the ordering $>>^2$ of the set of all pairs of terms.

$(s,t) >>^2 (s',t')$ iff $s >> s'$ and $(t >> t' \text{ or } t = t')$

or

$(s >> s' \text{ or } s = s')$ and $t >> t'$.

$>>$ is the containment ordering defined in [Hu81]: $s >> t$ iff s contains t (i.e. some subterm of s is a substitution instance of t) and t does not contain s.
$>>$ is a well-founded ordering, and so is $>>^2$.

Induction hypothesis 1:
for all $s \to t$, $s' \to t'$ in R such that $(\overline{s,s'}) >>^2 (s,s')$:
if $\sigma(t)$, $\sigma(s[p:=t'])$ is a critical pair derived from unifying s/p and s' by the most general unifier σ then
$\sigma(t) \leftarrow\!\!\rightarrow_R^* (<\sigma(s))\ \sigma(s[p:=t'])$.

For every pair of rewrite rules $k:s_k \to t_k, l:s_l \to t_l$ in R (w.l.o.g. $k \geq l$) there exists, because of the fairness of selection hypothesis, an iteration i in the completion algorithm such that either

(a) the rule k is selected for computing critical pairs and s_k and s_l have not been reduced in previous iterations

or

(b) the left hand side of one of the rules k,l is reduced.

Denote this iteration i by $it(s_k \to t_k, s_l \to t_l)$. We induct on i.

Induction hypothesis 2:
for all $s \to t, s' \to t'$ in R such that $(s,s') = (\overline{s,s'})$ and $it(s \to t, s' \to t') < i$:
if $\sigma(t)$, $\sigma(s[p:=t'])$ is a critical pair derived from unifying s/p and s' by the most general unifier σ then
$\sigma(t) \leftarrow\!\!\rightarrow_R^* (<\sigma(s))\ \sigma(s[p:=t'])$.

Now consider two rules in R
$k:s_k \to t_k$, $l:s_l \to t_l$
such that $(s_k,s_l) = (\overline{s,s'})$ and $it(s_k \to t_k, s_l \to t_l) = i$.

Case (a): assume k greater or equal to l (k less than l is handled analogously).
So at the i-th iteration the rule k is selected for computing critical pairs and s_k, s_l have not been reduced in previous iterations. Thus, in R_i we have rules $k:s_k \to t_k'$, $l:s_l \to t_l'$, where $t_k \to_R^* t_k'$ and $t_l \to_R^* t_l'$. The algorithm for computing necessary critical pairs is called with the input R_i and k. During the execution of this algorithm the rule l will be selected from S' at some iteration j.
We induct on j.

j=l: $S = S'$, so $\sigma(t_k') = \sigma(s_k[p:=t_l'])$ is added to CP and hence to E_{i+1}.
By Lemma 4.1 (b) there exists a term u such that
$\sigma(t_k') \to_R^* u \leftarrow_R^* \sigma(s_k[p:=t_l'])$.
So $\sigma(t_k) \to_R^* u \leftarrow_R^* \sigma(s_k[p:=t_l])$.
See Figure 1.

Induction hypothesis 3:
for all j' less than j:
if $m:s_m \to t_m$ is selected from S' at step j' and u,u' is a critical pair between the rules k and m derived from v, then $u \leftarrow\!\!\rightarrow_R^* (<v)\ u'$.

j>1: if the considered critical pair turns out to be necessary, then we can show $\sigma(t_k) \leftarrow\!\!\rightarrow_R^* (<\sigma(s_k))\ \sigma(s_k[p:=t_l])$ as in the induction basis. Otherwise there exists a rule $m:s_m \to t_m$ in S−S', and a prefix q of p (p=q.q') such that $s_m \lessdot \sigma(s_k)/q$ and rule m is marked and m is greater or equal to l or rule l is marked and l is greater or equal to m. Let ψ be such that $\psi(s_m) = \sigma(s_k)/q$. Then $\sigma(t_k')$ and $\sigma(s_k)[q:=\psi(t_m)]$ are the substitution instance of a critical pair between the rules k and m (see Proposition 3.7 in [Hu80]) and by the induction hypothesis 3 we have

$$\sigma(t_k') \leftarrow\!\!\rightarrow_R^* (<\sigma(s_k))\ \sigma(s_k)[q:=\psi(t_m)].$$

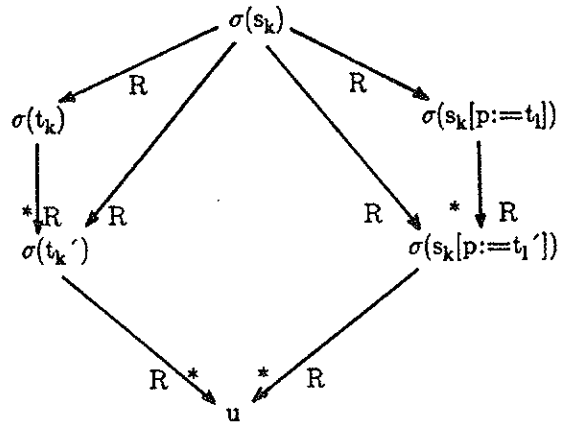


Figure 1

If q' is an occurrence of s_m and s_m/q' is not a variable then $\psi(t_m)$ and $\psi(s_m)[q':=\sigma(t_l')] = \sigma(s_k/q\ [q':=t_l'])$ are substitution instances of a critical pair between the rules m and l (see Proposition 3.7 in [Hu80]) and by the induction hypothesis 2 we have

$$\psi(t_m) \leftarrow\!\!\rightarrow_R^* (<\sigma(s_k)/q)\ \sigma(s_k/q\ [q':=t_l']) \qquad (*).$$

Otherwise, by an argument already used in [KB67], one can show that the two sides of (*) have a common successor. So,

$$\sigma(s_k)[q:=\psi(t_m)] \leftarrow\!\!\rightarrow_R^* (<\sigma(s_k))\ \sigma(s_k[p:=t_l']),$$

and therefore

$$\sigma(t_k) \leftarrow\!\!\rightarrow_R^* (<\sigma(s_k))\ \sigma(s_k[p:=t_l]).$$

See Figure 2.

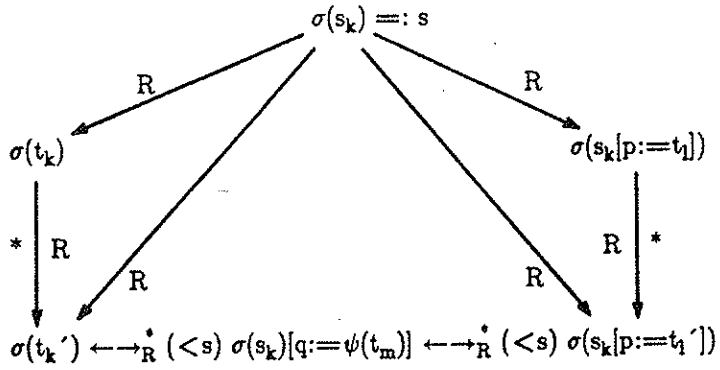


Figure 2

Case (b): The left hand side of at least one of the rules k and l is reduced in the iteration i of

the completion algorithm.
So there are rules $m:s_m \to t_m$, $n:s_n \to t_n$ in R such that $(s_k,s_l) >>^2 (s_m,s_n)$. Let (s_m,s_n) be maximal under this condition. Let $s_k/q = \psi(s_m)$, $s_l/q' = \phi(s_n)$. If $s_k = s_m$ then $q = \Lambda$, $\psi = \{\}$ and therefore trivially

$$\sigma(t_k') \longleftrightarrow_R^* (<\sigma(s_k))\ \sigma(s_k[q:=\psi(t_m)]),$$

where $t_k \to_R^* t_k'$.
Otherwise $s_k[q:=\psi(t_m)] = t_k'$ is added to E_{i+1} at the iteration i and therefore, by Lemma 4.1 (b)

$$\sigma(t_k') \longleftrightarrow_R^* (<\sigma(s_k))\ \sigma(s_k[q:=\psi(t_m)]).$$

Similarly we see that

$$\sigma(s_k[p.q':=\phi(t_n)]) \longleftrightarrow_R^* (<\sigma(s_k))\ \sigma(s_k[p:=t_l']).$$

Furthermore,

$$\sigma(s_k[q:=\psi(t_m)]) \longleftrightarrow_R^* (<\sigma(s_k))\ \sigma(s_k[p.q':=\phi(t_n)]),$$

since both sides result from a critical pair between rules m and n (so the induction hypothesis 1 applies) or it can be shown by an argument used in [KB67] that they have a common successor.
See Figure 3.

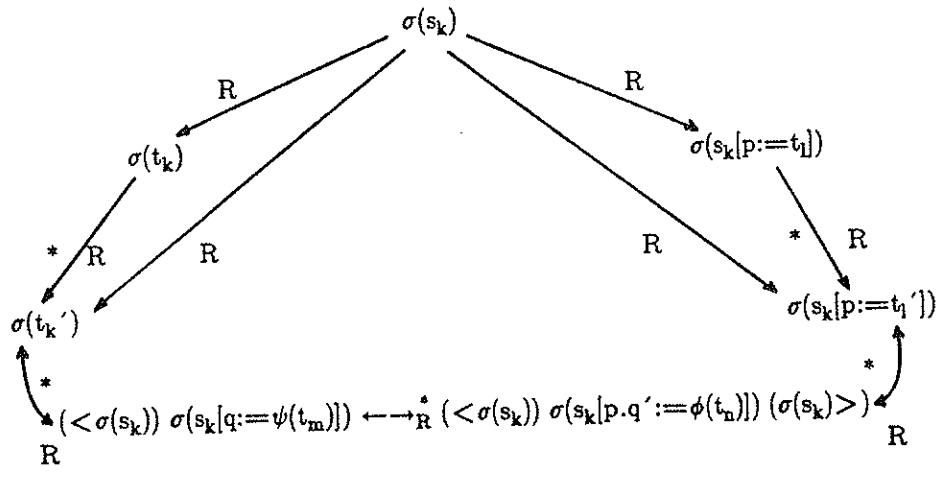


Figure 3

□

Using Lemma 4.2 and Huet's Lemma 3.1 [Hu80] we can show (see [Wi85])

Lemma 4.3: For all terms s,u_1,u_2 such that $u_1 \leftarrow_{R_\infty} s \rightarrow_{R_\infty} u_2$ we have $u_1 \longleftrightarrow_R^* (<s)\ u_2$.

Now we can prove a lemma similar to Lemma 6 of [Hu81]. This Lemma provides the basis for showing the confluence of R_∞ and R.

Lemma 4.4: For every term s:

(a) for all t such that $s \to_R^+ t$ there exists a term u such that $s \to_{R_\infty}^+ u \overset{*}{\leftarrow\!\!\rightarrow}_{R_\infty}^* (<s)\ t$

(b) for all t_1, t_2 such that $t_1 \leftarrow_{R_\infty}^+ s \to_{R_\infty}^+ t_2$: $t_1 \leftarrow\!\!\rightarrow_{R_\infty}^* (<s)\ t_2$

(c) for all t_1, t_2 such that $t_1 \leftarrow_R^+ s \to_R^+ t_2$: $t_1 \leftarrow\!\!\rightarrow_{R_\infty}^* (<s)\ t_2$.

Proof:
We show simultaneously (a), (b), and (c) by noetherian induction on the reduction ordering $>$.

Induction hypothesis:
for all terms s′ such that $s > s'$ (a),(b), and (c) hold.

ad (a): Let s_1 be such that $s \to_R s_1 \to_R^* t$, with $k: r_1 \to r_2$ the rule of R used to reduce s to s_1. We use induction on r_1, w.r.t. the well-founded order $>>$. There are two cases.

Case 1: There exists a rule with label k in R_∞, say $k: r_1 \to r_2'$, with $r_2 \to_R^* r_2'$.

This implies that for some u we have $s \to_{R_\infty} u$ and $s_1 \to_R^* u$. By induction hypothesis (c) applied to s_1 we get $u \leftarrow\!\!\rightarrow_{R_\infty}^* (\leq s_1)\ t$, and therefore $u \leftarrow\!\!\rightarrow_{R_\infty}^* (<s)\ t$. (Here "$\leq$" denotes the partial order $s \leq t$ iff $s < t$ or $s = t$.)
See Figure 4.

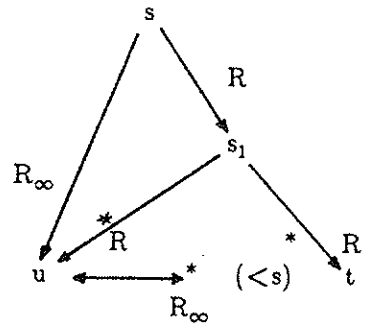


Figure 4

Case 2: The rule with label k gets reduced on its left hand side at some iteration i.
That is, there is in R_i some $k: r_1 \to r_2'$, with $r_2 \to_R^* r_2''$, such that r_1 is reducible, say to r_1', by the newly introduced rule $r_1'' \to r_2''$. By the compatibility and stability of the reduction, s is reducible by $r_1'' \to r_2''$ to say s_1', and the reduction $r_2 \to_R^* r_2'$ corresponds to a reduction $s_1 \to_R^* s_2$. Now we have $r_1' = r_2'$ in E_{i+1}, and by Lemma 4.1 (b) we get $s_1' \to_R^* s_3 \leftarrow_R^* s_2$ for some s_3. Using the induction hypothesis (c) at s_1 we get $s_3 \leftarrow\!\!\rightarrow_{R_\infty}^* (\leq s_1)\ t$, so $s_3 \leftarrow\!\!\rightarrow_{R_\infty}^* (<s)\ t$. Since $r_1 >> r_1''$ we may apply the induction hypothesis (a) to the reduction $s \to_R s_1' \to_R^* s_3$, which gives us a term u such that $s \to_{R_\infty}^+ u \leftarrow\!\!\rightarrow_{R_\infty}^* (<s)\ s_3$, and therefore $s \to_{R_\infty}^+ u \leftarrow\!\!\rightarrow_{R_\infty}^* (<s)\ t$. See Figure 5.

ad (b): Let s_1, and s_2 be such that

$$t_1 \leftarrow\!\!\rightarrow_{R_\infty}^* s_1 \leftarrow_{R_\infty} s \to_{R_\infty} s_2 \to\!\!\rightarrow_{R_\infty}^* t_2.$$

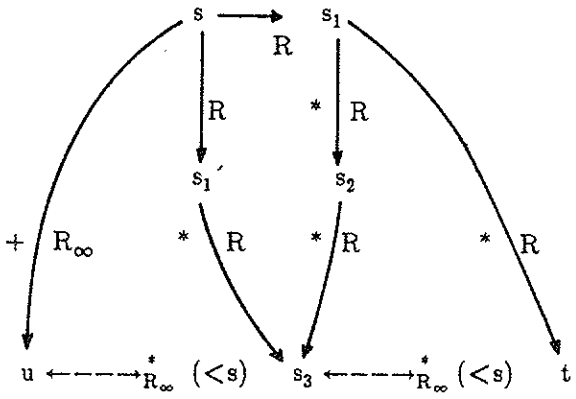


Figure 5

By Lemma 4.3 we have $s_1 \longleftrightarrow^*_R (<s)\ s_2$.
I.e. there are $u_1, \ldots, u_n$ such that

$$s_1 = u_1 \longleftrightarrow_R u_2 \longleftrightarrow_R \ldots \longleftrightarrow_R u_n = s_2 \text{ and } u_i < s \text{ for } 1 \leq i \leq n.$$

We show (*) $u_1 \longleftrightarrow^*_{R_\infty} (<s)\ u_n$ by induction on n.
n=1: clear.

Induction hypothesis 2: (*) holds for $\bar{n}$.
Now let $u_1 \longleftrightarrow_R u_2 \longleftrightarrow_R \cdots \longleftrightarrow_R u_{\bar{n}} \longleftrightarrow_R u_{\bar{n}+1}$ and $u_i < s$ for $1 \leq i \leq \bar{n}+1$. By induction hypothesis 2 we have $u_1 \longleftrightarrow^*_{R_\infty} (<s)\ u_{\bar{n}}$. If $u_{\bar{n}} \rightarrow_R u_{\bar{n}+1}$ then by induction hypothesis (a) there exists a term u such that $u_{\bar{n}} \rightarrow^+_{R_\infty} u \longleftrightarrow^*_{R_\infty} (<s)\ u_{\bar{n}+1}$. If $u_{\bar{n}+1} \rightarrow_R u_{\bar{n}}$ then by induction hypothesis (a) there exists a term u such that $u_{\bar{n}+1} \rightarrow^+_{R_\infty} u \longleftrightarrow^*_{R_\infty} (<s)\ u_{\bar{n}}$. In either case we have $u_1 \longleftrightarrow^*_{R_\infty} (<s)\ u_{\bar{n}+1}$.

So, $s_1 \longleftrightarrow^*_{R_\infty} (<s)\ s_2$, and therefore $t_1 \longleftrightarrow^*_{R_\infty} (<s)\ t_2$. See Figure 6.

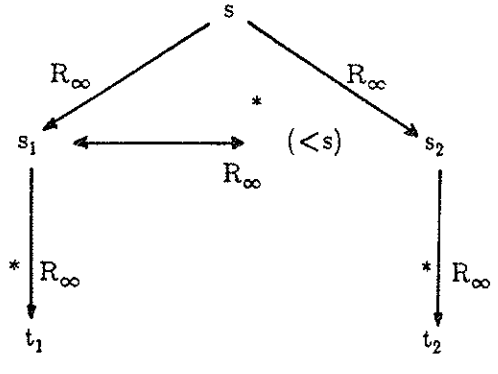


Figure 6

ad (c): Straightforward application of (a) and (b). See Figure 7.

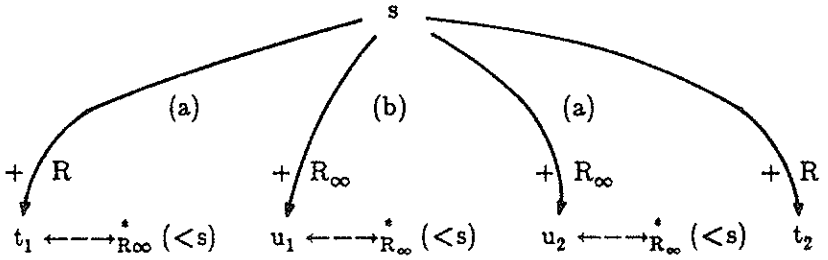


Figure 7

□

The generalized Newman Lemma (Lemma 2.1) together with Lemma 4.1 and Lemma 4.4 yield the desired result (see [Wi85]).

Theorem 4.5:
Both $\rightarrow_R$ and $\rightarrow_{R_\infty}$ have the Church-Rosser property and $\overset{*}{\longleftrightarrow}_{R_\infty} = =_E$.

Conclusion

As we have pointed out in the introduction, it is possible to keep the system of rewrite rules interreduced during the execution of the Knuth-Bendix completion algorithm and simultaneously apply a criterion for eliminating unnecessary critical pairs. Such a version of the completion algorithm is given in Section 3. The criterion is incorporated in the subalgorithm for computing necessary critical pairs. Theorem 4.5 and 4.6 show the correctness of this version of the completion algorithm.

References

[Bu65] B. Buchberger: Ein Algorithmus zum Auffinden der Basiselemente des Restklassenringes nach einem nulldimensionalen Polynomideal, doctoral dissertation, Univ. Innsbruck, 1965.

[Bu70] B. Buchberger: Ein algorithmisches Kriterium für die Lösbarkeit eines algebraischen Gleichungssystems, Aequationes Math. 4/3, 374-383, 1970.

[Bu79] B. Buchberger: A Criterion for Detecting Unnecessary Reductions in the Construction of Gröbner Bases, Proc. EUROSAM 79, Marseille, 1979, Lecture Notes in Computer Science 72, 3-21, Berlin-Heidelberg-New York, Springer-Verlag, 1979.

[Bu85] B. Buchberger: Gröbner Bases: An Algorithmic Method in Polynomial Ideal Theory, in: Recent Trends in Multidimensional Systems Theory, N.K. Bose, ed., D. Reidel Publishing Comp., to appear in 1985. Available also as CAMP-Report Nr. 83-29.0, Inst. of Math., Univ. Linz, 1983.

[BC83] B. Buchberger, G.E. Collins, R. Loos: Computer Algebra - Symbolic and Algebraic Computation, 2nd ed., Springer-Verlag, 1983.

[De83] N. Dershowitz: Applications of the Knuth-Bendix Completion Procedure, Laboratory Operations, The Aerospace Corp., El Segundo, Calif. 90245, Aerospace Rep. ATR-83(8478)-2, 1983.

[Hs82] J. Hsiang: Topics in Automated Theorem Proving and Program Generation, Ph.D. thesis, Dept. of Comp. Sci., Univ. of Illinois, Urbana, Ill., 1982.

[Hu80] G. Huet: Confluent Reductions: Abstract Properties and Applications to Term Rewriting Systems, JACM 27/4, 797-821, 1980.

[Hu81] G. Huet: A Complete Proof of Correctness of the Knuth-Bendix Completion Algorithm, J. Computer and System Sciences 23, 11-21, 1981.

[HH80] G. Huet, J.M. Hullot: Proofs by Induction in Equational Theories with Constructors, 21st IEEE Symp. on Foundations of Comp. Sci., 96-107, 1980.

[KB67] D.E. Knuth, P.B. Bendix: Simple Word Problems in Universal Algebra, Proc. of the Conf. on Computational Problems in Abstract Algebra, Oxford, 1967, J. Leech (ed.), Pergamon Press, 1970.

[Mu80] D.R. Musser: On Proving Inductive Properties of Abstract Data Types, 7th ACM Symp. on Principles of Progr. Languages, 154-162, 1980.

[Ne42] M.H.A. Newman: On Theories with a Combinatorial Definition of "Equivalence", Ann. Math. 43/2, 223-243, 1942.

[Wi83] F. Winkler: A Criterion for Eliminating Unnecessary Reductions in the Knuth-Bendix Algorithm, Tech. Rep. CAMP-83-14.1, Inst. f. Math., J. Kepler Univ., Linz, 1983.

[Wi84] F. Winkler: The Church-Rosser Property in Computer Algebra and Special Theorem Proving: An Investigation of Critical-Pair/Completion Algorithms, doctoral dissertation, Inst. f. Math., J. Kepler Univ., Linz, 1984.

[Wi85] F. Winkler: A Note on Improving the Complexity of the Knuth-Bendix Algorithm, Tech. Rep. No. 85-04, Dept. of Comp. and Inf. Sci., Univ. of Delaware, 1985.

[WB83] F. Winkler, B. Buchberger: A Criterion for Eliminating Unnecessary Reductions in the Knuth-Bendix Algorithm, Proc. Colloquium on Algebra, Combinatorics and Logic in Computer Science, Györ, Hungary, 1983, Colloquia Mathematica Societatis J. Bolyai, J. Bolyai Math. Soc. and North-Holland, 1985.