COLLOQUIA MATHEMATICA SOCIETATIS JÁNOS BOLYAI
42. Algebra, Combinatorics and Logic in Computer Science
Györ (Hungary), 1983.

A CRITERION FOR ELIMINATING UNNECESSARY REDUCTIONS IN THE KNUTH-BENDIX ALGORITHM [1)]

F. WINKLER - B. BUCHBERGER

Abstract.

Given a finite set E of multivariate polynomials, the second author's 1965 algorithm constructs a set E' of polynomials such that E and E' generate the same ideal congruence, but E' has the additional property that the congruence can be decided by "reduction to normal forms w.r.t. E'". The two essential ideas of this algorithm are: the consideration of certain "critical pairs" associated with elements in E and the successive "completion" of E by adjoining the difference of normal forms of critical pairs. In a different context, the analogous two ideas appeared later (1967) in the K n u t h - B e n d i x algorithm: given a finite set E of equations between first-order terms, the K n u t h - B e n d i x algorithm constructs a set E' of equations such that E and E' generate the same first order equational congruence and, again, this congruence can be decided by "reduction to normal forms w.r.t. E'". Exploiting the structural similarity between the two algorithms, in

1) An essential part of the results reported in this paper will be included in the first author's doctoral dissertation at the University of Linz, Austria. The work for this paper was supported by the Austrian Research Fund under grant Nr. 4567.

the present paper it is shown how a recent (1979) improvement of the first algorithm based on a certain "overlap condition" can be carried over to the K n u t h - B e n d i x algorithm. The formulation and correctness proof of this refined version of the K n u t h - B e n d i x algorithm need some new insights and techniques specific for the context of first-order terms. On the other hand, as a general framework for such proofs a generalized N e w m a n lemma is proven that establishes the C h u r c h - R o s s e r property for reduction relations under a very weak hypothesis.

1. INTRODUCTION

In [Bu65],[Bu70] an algorithm has been introduced that solves the following problem: given a finite set E of multivariate polynomials over a field construct a set E' of polynomials such that $\equiv_E = \equiv_{E'}$ and $\rightarrow_{E'}$ has the C h u r c h - R o s s e r property. Here, for sets E of polynomials, $\equiv_E$ is the ideal congruence w.r.t. the ideal generated by E and $\rightarrow_E$ is a certain reduction relation on polynomials induced by E with the property that $\leftrightarrow^*_E$ (the reflexive-symmetric transitive-closure of $\rightarrow_E$) is equal to $\equiv_E$. The C h u r c h - R o s s e r property of $\rightarrow_{E'}$ (see definition in Section 2.1), then, guarantees that, for polynomials α and β, "$\alpha \leftrightarrow^*_E \beta$" ("$\alpha \equiv_{E'} \beta$", "$\alpha \equiv_E \beta$", the *"main problem of polynomial ideal theory"*, see [VW67], p.166) can be decided by simply reducing α and β to normal forms w.r.t. $\rightarrow_{E'}$. As a special case, this method includes an algorithmic solution to the word problem for finitely generated commutative semigroups. Also, a large number of other computability and decidability problems can be solved easily for polynomial ideals generated by sets E' whose $\rightarrow_{E'}$ has the C h u r c h - R o s s e r property [Bu65],[Bu70], [WBLR81], and [WBLR85]. In [Bu76] such sets E' have been called "Gröbnerbases". New applications of the method have been given in [Gu82], [Ba82], and [MM83].

Roughly, the basic structure of the algorithm in [Bu65], [Bu70] is:

CPC algorithm: $E' \leftarrow E$;
while not all "critical pairs" of E' are considered *do*
  $(t_1,t_2) \leftarrow$ a "critical pair" of E' which has not yet been considered;
  (*) $(\beta_1,\beta_2) \leftarrow$ the reduced forms of t_1,t_2 w.r.t. E';
  if $\beta_1 \neq \beta_2$ *then*
    "complete" E' by β_1-β_2.

The two basic strategies of the algorithm, thus, are the formation of *"critical pairs"* and the successive *"completion"* step.

In the context of general first-order terms instead of polynomials, the same two ideas appeared later (1967) in the well known K n u t h - B e n d i x algorithm [KB67] which now is widely used in computer algebra and software technology, in particular in the manipulation of abstract data type specifications. The K n u t h - B e n d i x algorithm solves the following problem: given a finite set E of equations between first-order terms (satisfying certain conditions) construct a set E' of equations such that $\equiv_E = \equiv_{E'}$ and $\rightarrow_{E'}$ has the C h u r c h - R o s s e r property. Here, for sets E of first-order equations, $\equiv_E$ is the equational congruence generated by E and $\rightarrow_E$ is the reduction relation on terms induced by E viewed as a system of rewrite rules. $\rightarrow_E$ has the property $\leftrightarrow^*_E = \equiv_E$. Again, the C h u r c h - R o s s e r property, then, guarantees that $\equiv_E$ can be decided by reduction to normal forms.

In fact, the *CPC* algorithm shown above can now as well be read as the K n u t h - B e n d i x algorithm if the appropriate notion of "critical pair" is used and "complete E' by β_1-β_2" is replaced by "complete E' by the equation β_1=β_2". The common aspects of such "critical-pair/completion" (CPC) algorithms have first been pointed out in [Lo81] and [BL82]. Actually, after some earlier

attempts [Wi82], we have been able to show recently [Wi84] that there exists a common "super-algorithm" to the [Bu65], [Bu70] algorithm and the Knuth-Bendix algorithm, namely a modified version of the Peterson-Stickel algorithm [PS81]. R. Llopis [Ll83] has provided many of the concepts needed for establishing this relationship.

In [Bu79] the basic scheme of the CPC algorithm was improved by introducing the possibility of applying certain "criteria" at point (*) of the algorithm. These criteria allow to detect that for certain critical pairs t_1,t_2 the subsequent instructions, including the reductions of t_1,t_2 to normal forms β_1,β_2 may be skipped without affecting the correctness of the algorithm. This may result in a substantial speed-up of the algorithm because, in general, the reduction process is an expensive operation: without application of the criteria, the number b of critical pairs that have to be reduced is quadratic in the number ℓ of polynomials in E' (ℓ, however, in general grows dynamically in the course of the algorithm!). By application of the criteria, in typical cases b is only linear in ℓ!

Given the formal similarity between the [Bu65], [Bu70] algorithm and the Knuth-Bendix algorithm it seems to be natural to carry over the [Bu79] improvement of the first algorithm to the Knuth-Bendix algorithm. Although the idea is near at hand, the actual adaption of the strategy of "criteria" to the Knuth-Bendix algorithm is by no means straightforward: the kind of "overlappings" of terms that must be considered in the appropriate criteria are much richer than in the case of polynomials. The detailed study and algorithmical exploitation of these phenomena forms the central part of this paper (Section 4) and, finally, leads to an improved version of the Knuth-Bendix algorithm along the lines of [Bu79].

In addition, we are able to single out the part of the correctness proof for the improved algorithm which is common to CPC algorithms that apply "criteria": we formulate and prove a new lemma of

the "Newman-type", which establishes the Church-Rosser property of general reduction relations under a very weak hypothesis, and seems to be the strongest lemma of this type known so far (Section 3).

Section 2 of the paper compiles well known definitions and lemmata, first, about general reduction relations and, second, about terms, equations, and rewrite rules. These definitions and lemmata are a necessary prerequisite for both the original formulation of the Knuth-Bendix algorithm and the improvement given here. The proofs of these lemmata may be found in [KB67], [Hu80], [HO80], and [Wi83]. The proofs on reduction systems are also presented in [BL82]. Section 5 provides an extensive example.

2. PREREQUISITES: DEFINITIONS AND LEMMATA

2.1 General reduction relations

For a binary relation $\to$ on some set M the inverse relation, the transitive closure, the reflexive-transitive closure and the reflexive-symmetric-transitive closure of $\to$ will be denoted by $\leftarrow$, $\to^+$, $\to^*$, $\leftrightarrow^*$. If $\to$ is clear from the context, by $\underset{\sim}{x}$ we denote that $x \in M$ is in *$\to$-normal form* (i.e. there is no $y \in M$ such that $x \to y$). x and y have a *common successor in one step* ($x \downarrow y$) iff for some $z \in M$: $x \to z \leftarrow y$. x and y have a *common successor* ($x \downarrow^* y$) iff for some $z \in M$: $x \to^* z \leftarrow^* y$. x and y have a *common predecessor in one step* ($x \uparrow y$) iff for some $z \in M$: $x \leftarrow z \to y$. x and y have a *common predecessor* ($x \uparrow^* y$) iff for some $z \in M$: $x \leftarrow^* z \to^* y$.

Definition. A function $S: M \to M$ is called a *canonical simplifier* for $\leftrightarrow^*$ iff for all $x,y \in M$: $x \leftrightarrow^* S(x)$ and if $x \leftrightarrow^* y$ then $S(x) = S(y)$.

Let $\to$ be a noetherian reduction relation on M (i.e. there

are no infinitely descending chains) and let $Sel: M \to M$ be such that for all x which are not in $\to$-normal form: $x \to Sel(x)$. Let S be defined as follows:

$S(x) := x$ if x is in normal form w.r.t. $\to$ and $S(Sel(x))$ otherwise.

Then

Lemma 2.1.1. *S is a canonical simplifier for $\leftrightarrow^*$ iff $\to$ has the Church-Rosser property (i.e. for all $x,y \in M$: if $x \leftrightarrow^* y$ then $x \downarrow^* y$).*

Lemma 2.1.2. *$\to$ has the Church-Rosser property iff $\to$ is confluent (i.e. for all $x,y \in M$: if $x \uparrow^* y$ then $x \downarrow^* y$).*

Lemma 2.1.3. (Newman lemma [Ne42]) *$\to$ is confluent iff $\to$ is locally confluent (i.e. for all $x,y \in M$: if $x \uparrow y$ then $x \downarrow^* y$).*

For the proofs of these lemmata we refer to [BL82].

2.2. Reduction of terms w.r.t. rewrite rules

Let V be a denumerable set of variables, Σ a finite or denumerable set of operator symbols disjoint from V, and a a function from Σ to $\mathbf{N}_0$, the arity function for Σ. The set T of *terms* is defined as the free a-graded Σ-algebra generated by V. That is, a term is either a variable or is of the form $f\alpha_1 \ldots \alpha_{a(f)}$ for some $f \in \Sigma$ and $\alpha_1, \ldots, \alpha_{a(f)} \in T$.

Let $N^0 := \{\Lambda\}$, $N^* := \bigcup_{j=0}^{\infty} N^j$. We write an element $p \in N^j$, $j \geq 1$, as $p_1 \cdot \ldots \cdot p_j$.

A function S from the set of terms into the set of terms is called a *substitution* iff $S(v)=v$ for all but a finite number of variables and $S(\alpha)=fS(\alpha_1)\ldots S(\alpha_n)$ for $\alpha=f\alpha_1\ldots\alpha_n \in T-V$.

Let in the sequal v be a metavariable ranging over the set V of variables, f a metavariable ranging over the set Σ of operator symbols, $\alpha,\beta,\gamma,\delta$ be metavariables ranging over the set T of terms, p,q,r be metavariables ranging over the set N^*, and S,T,U be metavariables ranging over the set of substitutions.

p is an *occurrence* of α iff either $p=\Lambda$ or (if $\alpha=f\alpha_1\ldots\alpha_n$) $p=i \cdot q$, $1 \leq i \leq n$, and q is an occurrence of α_i. By $occ(\alpha)$ we denote the *set of occurrences* of α. For $p \in occ(\alpha)$ we denote the *subterm* of α at p by α/p, i.e. $\alpha/p=\alpha$ if $p=\Lambda$ and $\alpha/p=\alpha_i/q$ if $p=i \cdot q$ and $\alpha=f\alpha_1\ldots\alpha_n$. β *occurs* in α iff $\beta=\alpha/p$ for some $p \in occ(\alpha)$. α and β are *unifiable* ($\alpha \nabla \beta$) iff there are S,S' such that $S(\alpha)=S'(\beta)$. By $\preccurlyeq$ we denote the *instantiation* or *subsumption preorder* on the set of terms, i.e. $\alpha \preccurlyeq \beta$ iff $S(\alpha)=\beta$ for some S. For $p \in occ(\alpha)$ the *replacement* of the subterm of α at p by β is defined as $\alpha[p \leftarrow \beta]=\beta$ if $p=\Lambda$ and $\alpha[p \leftarrow \beta]=f\alpha_1\ldots\alpha_{i-1}\,\alpha_i[q \leftarrow \beta]\,\alpha_{i+1}\ldots\alpha_n$ if $p=i \cdot q$ and $\alpha=f\alpha_1\ldots\alpha_n$.

Lemma 2.2.1.

(*i*) *If $p \cdot q \in occ(\alpha)$ then $q \in occ(\alpha/p)$ and $\alpha/p \cdot q = (\alpha/p)/q$.*

(*ii*) *If $p \in occ(\alpha)$ then $p \in occ(S(\alpha))$ and $S(\alpha/p)=S(\alpha)/p$.*

(*iii*) *If $p=q \cdot r \in occ(\alpha)$ then $\alpha[p \leftarrow \beta]=\alpha[q \leftarrow \alpha/q[r \leftarrow \beta]]$.*

(*iv*) *If $p \in occ(\alpha)$, $q \in occ(\beta)$ then $p \cdot q \in occ(\alpha[p \leftarrow \beta])$ and $\alpha[p \leftarrow \beta[q \leftarrow \gamma]] = \alpha[p \leftarrow \beta]\,[p \cdot q \leftarrow \gamma]$.*

(*v*) *If $p \in occ(\alpha)$ then $S(\alpha[p \leftarrow \beta]) = S(\alpha)[p \leftarrow S(\beta)]$.*

Let $>_t$ be a partial ordering on the set of terms such that

(a) $>_t$ is noetherian and

(b) if $\alpha >_t \beta$ then $S(\alpha) >_t S(\beta)$ and $\gamma[p \leftarrow \alpha] >_t \gamma[p \leftarrow \beta]$ for $p \in occ(\gamma)$

("$>_t$ *is stable and compatible*").

If neither $\alpha \geq_t \beta$ nor $\beta \geq_t \alpha$ then we write $\alpha \#_t \beta$

For instance the partial ordering defined in [KB67], p.267, satisfies the conditions (a) and (b).

Lemma 2.2.2. *If $\alpha >_t \beta$ and $S(\alpha)=S'(\alpha)$ then $S(\beta)=S'(\beta)$.*

Definition. A set $R \subseteq T \times T$ is called a *reduction system* iff

$\lambda >_t \rho$ for all $(\lambda,\rho) \in R$.

Definition. Let R be a reduction system. α *reduces to* β *modulo* R *in one step*, $\alpha \to_R \beta$, iff there is an occurrence $p \in occ(\alpha)$, a substitution S and a pair (λ,ρ) in R such that $\alpha/_p = S(\lambda)$ and $\beta = \alpha[p \leftarrow S(\rho)]$.

Lemma 2.2.3. *Let R be a reduction system. If $\alpha \to_R \beta$ then $\alpha >_t \beta$.*

Corollary. to Lemma 2.2.3. *Let R be a reduction system. Then $\to_R$ is a noetherian relation on the set of terms.*

Lemma 2.2.4. *Let R be a reduction system, $p \in occ(\gamma)$.*

(i) If $\alpha \to_R \beta$ then $S(\alpha) \to_R S(\beta)$.

(ii) If $\alpha \to_R \beta$ then $\gamma[p \leftarrow \alpha] \to_R \gamma[p \leftarrow \beta]$.

2.3. Equational congruence and reduction

If E is a set of pairs of terms then by $\equiv_E$ we denote the *smallest congruence relation* on the set of terms which contains E (*equivalence modulo* E).

Lemma 2.3.1. *Let R be a reduction system. Then $\equiv_R = \leftrightarrow^*_R$.*

Now, let Sel_R be any algorithm that satisfies "if α is not in normal form w.r.t. $\to_R$ then $\alpha \to_R Sel_R(\alpha)$" and let S be defined by Sel_R as in Section 2.1. Then we know from the preceding lemma and the lemmata in Section 2.1 that $\equiv_R$ can ben decided by S in the following simple way "$\alpha \equiv_R \beta$ iff $S(\alpha)=S(\beta)$" in case $\to_R$ is locally confluent, i.e. for all β_1,β_2: if $\beta_1 \uparrow_R \beta_2$ then $\beta_1 \downarrow^*_R \beta_2$. The test for local confluence can be made algorithmic by the K n u t h - B e n d i x theorem.

2.4. The Knuth-Bendix theorem and algorithm

Definition. Let $p \in occ(\beta)$. $unif(\alpha,p,\beta) := \{\gamma \mid (\exists S,T)(\gamma = S(\beta) \ \& \ \gamma/_p = T(\alpha))\}$.

Lemma 2.4.1. ([KB67]) *Let $p \in occ(\beta)$. If $\alpha \nabla \beta/_p$ then there is a term γ such that $unif(\alpha,p,\beta) = \{\delta \mid (\exists S)(\delta = S(\gamma))\}$.*

Definition. For $p \in occ(\beta)$ and $\alpha \nabla \beta/_p$ let $\sigma(\alpha,p,\beta)$ be such that $unif(\alpha,p,\beta) = \{\delta \mid (\exists S)(\delta = S(\sigma(\alpha,p,\beta)))\}$. $\sigma(\alpha,p,\beta)$ is called the *superposition* of α on β at p.

Definition. Let R be a reduction system, $(\lambda,\rho),(\lambda',\rho') \in R$, $p \in occ(\lambda')$ such that $\lambda \nabla \lambda'/_p$. Let T,U be such that $\sigma(\lambda,p,\lambda') = U(\lambda') = U(\lambda')[p \leftarrow T(\lambda)]$. Then

$(\tau_1(\lambda,\rho,p,\lambda',\rho'))$, $\tau_2(\lambda,\rho,p,\lambda',\rho')) := (U(\lambda')[p \leftarrow T(\rho)],\ U(\rho'))$

is the *critical pair* associated with $(\lambda,\rho),(\lambda',\rho')$ and p.

Definition. Let R be a reduction system.

$CP(R) := \{(\lambda,\rho,p,\lambda',\rho') \mid (\lambda,\rho),(\lambda',\rho') \in R,\ p \in occ(\lambda'),\ \lambda'/_p \in T\text{-}V \ \lambda \nabla \lambda'/_p\} - \{(\lambda,\rho,\Lambda,\lambda,\rho) \mid (\lambda,\rho) \in R\}$.

Theorem 1. (K n u t h - B e n d i x theorem, [KB67]): *Let R be a reduction system. Then $\to_R$ is locally confluent (and, hence, S is a canonical simplifier for $\equiv_R$) iff $\tau_1(\lambda,\rho,p,\lambda',\rho') \downarrow^*_R \tau_2(\lambda,\rho,p,\lambda',\rho')$ for all $(\lambda,\rho,p,\lambda',\rho') \in CP(R)$.*

The value of the K n u t h - B e n d i x theorem lies in the fact that the test for local confluence in infinitely many situations is reduced to the finitely many "critical situations" $\tau_1(\lambda,\rho,p,\lambda',\rho') \leftarrow_R \sigma(\lambda,p,\lambda') \to_R \tau_2(\lambda,\rho,p,\lambda',\rho')$.

The K n u t h - B e n d i x theorem gives rise to the Knuth-Bendix completion algorithm which for a given finite reduction system R computes (if it stops successfully) a finite reduction system R' such that $\equiv_R = \equiv_{R'}$ and $\to_R$ has the C h u r c h - R o s s e r property.

[K n u t h - B e n d i x algorithm]

$R' \leftarrow R$; $C \leftarrow CP(R')$;
while C not empty *do*
  $(\lambda,\rho,p,\lambda',\rho') \leftarrow$ an element of C;
  $C \leftarrow C - \{(\lambda,\rho,p,\lambda',\rho')\}$;
  $(\beta_1,\beta_2) \leftarrow \rightarrow_R$-normal forms of $(\tau_1(\lambda,\rho,p,\lambda',\rho'),\tau_2(\lambda,\rho,p,\lambda',\rho'))$;
  if $\beta_1 \neq \beta_2$ *then*
    if $\beta_1 \#_t \beta_2$ *then failure*
      else $R'' \leftarrow R' \cup \{(max_{>_t}(\beta_1,\beta_2), min_{>_t}(\beta_1,\beta_2))\}$;
        $C \leftarrow C \cup (CP(R'')-CP(R'))$;
        $R' \leftarrow R''$ •

3. A GENERALIZED N E W M A N LEMMA

From the lemmata quoted in Section 2, the following can be seen: the proof of the fact, that the C h u r c h - R o s s e r property of $\rightarrow_R$ can be tested by checking whether all "critical pairs" w.r.t. R have a common successor, can be decomposed into the proof of Newman's lemma which is true for arbitrary noetherian reduction relations and the proof of the K n u t h - B e n d i x theorem which incorporates the special knowledge available for term rewrite systems.

In this section we formulate and prove a stronger version of Newman's lemma which may serve as the basis for the proof of our improved version of the K n u t h - B e n d i x algorithm, as far as the overall structure of the proof is concerned. This lemma incorporates also the structure of the correctness proof for the algorithm in [Bu79] and may be useful in other contexts, too.

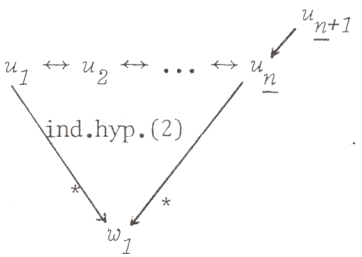
Lemma 3.1. *Let $>$ be noetherian partial ordering on M and $\rightarrow \subseteq >$. Then $\rightarrow$ is confluent iff for all $x,y,z \in M$: if $x \leftarrow y \rightarrow z$ then $x \leftrightarrow^*$ $(<z)$ y.*

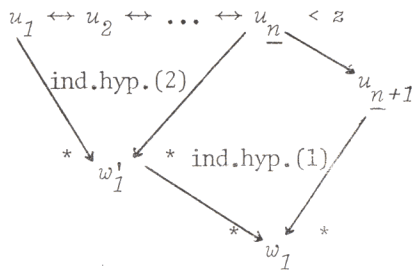
Here, we define $x \leftrightarrow^*$ $(<z)$ y iff there exists a finite sequence of elements $u_1,\ldots,u_n \in M$ such that $x=u_1 \leftrightarrow u_2 \leftrightarrow \ldots \leftrightarrow u_{n-1} \leftrightarrow u_n=y$ and $u_i < z$ for $i=1,\ldots,n$.

Proof. "=>": Clear.

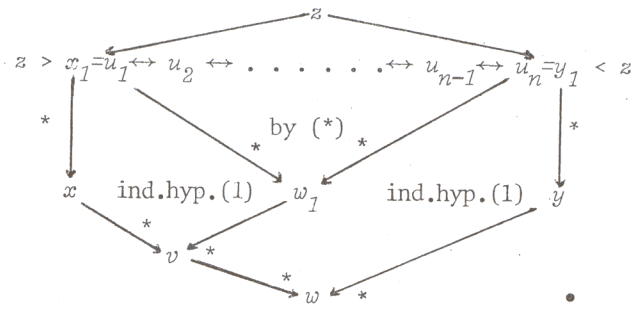
"<=": By noetherian induction on $<$.

Assume z,x,y are arbitrary, but fixed such that $x \leftarrow^* z \rightarrow^* y$.
Induction hypothesis (1): for all $z'<z$, x',y': if $x' \leftarrow^* z' \rightarrow^* y'$ then $x' \downarrow^* y'$.
We have to show $x \downarrow^* y$, i.e. $x \rightarrow^* w \leftarrow^* y$ for some w.
Case $z=x$ and case $z=y$: easy.
Case $z \neq x$ and $z \neq y$: in this case there exist x_1 and y_1 such that $x \leftarrow^* x_1 \leftarrow z \rightarrow y_1 \rightarrow^* y$. By the assumption of the lemma, there exist $u_1,u_2,\ldots,u_n$ such that $x_1=u_1 \leftrightarrow u_2 \leftrightarrow \ldots \leftrightarrow u_{n-1} \leftrightarrow u_n = y_1$ and all $u_i < z$.
We now proceed by induction on n and show:
For all n, for all $u_1,\ldots,u_n$:
(*) if $u_1 \leftrightarrow u_2 \leftrightarrow \ldots \leftrightarrow u_{n-1} \leftrightarrow u_n$ and all $u_i < z$ then $u_1 \downarrow^* u_n$.
(*) is clear for $n=1$.
Induction hypothesis (2): (*) is true for $n=\underline{n}$.
We now have two cases in which the existence of a common successor w_1 to x_1 and y_1 can be shown by the following two arguments:

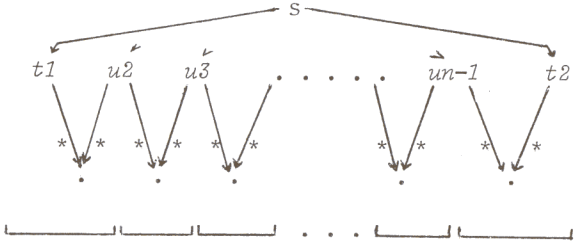




The proof can now be completed by:



Essentially, this lemma, in [Bu79] leads to the idea that, for guaranteeing the existence of a common successor to a "critical pair" t_1,t_2 it suffices to guarantee the existence of a "chain of critical pairs" between t_1 and t_2 that already have common successors and stay "below" s, the "least common multiple" from which t_1 and t_2 are formed:



"chain of critical pairs"

The difficulty in exploiting this idea for a concrete reduction relation lies in the specific way the concept of "staying below s" must be formulated. In [Bu79] it was simple: the heads of the intermediate basis polynomials must be divisors of s. In this paper, for the term reduction relation, the condition is much more complicated and will be developed in the next section.

4. REFINEMENT OF THE K N U T H - B E N D I X THEOREM AND ALGORITHM

Definition. Let R be a reduction system, $(\lambda,\rho,p,\lambda',\rho') \in CP(R)$. *connected*$(\lambda,\rho,p,\lambda',\rho',R)$ iff

$(\exists k \in \mathbb{N}, (\lambda_0,\rho_0),\ldots,(\lambda_k,\rho_k) \in R, q_0,\ldots,q_{k-1} \in N^*)$

$((\lambda_0,\rho_0)=(\lambda,\rho), (\lambda_k,\rho_k)=(\lambda',\rho'), p=q_{k-1}\cdot\ldots\cdot q_0,$

$(\forall 0\leq m\leq k-1)\ (q_m \in occ(\lambda_{m+1}),$

$\tau_1(\lambda_m,\rho_m,q_m,\lambda_{m+1},\rho_{m+1}) \downarrow^*_R \tau_2(\lambda_m,\rho_m,q_m,\lambda_{m+1},\rho_{m+1}),$

$\sigma(\lambda_m,q_m,\lambda_{m+1})$ *exists and*

$\sigma(\lambda_m,q_m,\lambda_{m+1}) \preccurlyeq \sigma(\lambda,p,\lambda')/q_{k-1}\cdot\ldots\cdot q_{m+1}$

where, by definition, $q_{k-1}\cdot\ldots\cdot q_{m+1}=\Lambda$ if $m=k-1$)).

Theorem 2. *Let R be a reduction system. Then $\rightarrow_R$ is locally confluent iff connected $(\lambda,\rho,p,\lambda',\rho',R)$ for all $(\lambda,\rho,p,\lambda',\rho') \in CP(R)$.*

Proof. "==>" : Clear.

"<==": By the Lemma 3.1 it suffices to show that for arbitrary α, α_1,α_2 with the property $\alpha_1 \leftarrow_R \alpha \rightarrow_R \alpha_2$ one can construct $\eta_0,\eta_1,\ldots,\eta_k$ such that $\alpha_1=\eta_0 \downarrow^*_R \eta_1 \downarrow^*_R \ldots \downarrow^*_R \eta_k = \alpha_2$ and $\eta_m <<_t \alpha$ for $0\leq m\leq k$.

In [KB67] it is shown that α_1 and α_2 always have a common successor with the possible exception of the following case:

$$\alpha_{/r} = S(\sigma(\lambda,p,\lambda')),$$

$$\alpha_1=\alpha[r\leftarrow S(\tau_1(\lambda,\rho,p,\lambda',\rho'))],\ \alpha_2=\alpha[r\leftarrow S(\tau_2(\lambda,\rho,p,\lambda',\rho'))],$$

where $(\lambda,\rho,p,\lambda',\rho') \in CP(R)$.

By assumption there are $(\lambda_0,\rho_0),\ldots,(\lambda_k,\rho_k) \in R$, $q_0,\ldots,q_{k-1} \in N^*$ and substitutions $S^0,\ldots,S^{k-1}$ such that

$(\lambda_0,\rho_0)=(\lambda,\rho)$, $(\lambda_k,\rho_k)=(\lambda',\rho')$.

$p=q_{k-1}\cdot\ldots\cdot q_0$,

(1) $(\forall\ 0\leq m\leq k-1)\ (q_m \in occ(\lambda_{m+1}),$

$\tau_1(\lambda_m,\rho_m,q_m,\lambda_{m+1},\rho_{m+1}) \downarrow^*_R \tau_2(\lambda_m,\rho_m,q_m,\lambda_{m+1},\rho_{m+1}),$

$\sigma(\lambda_m, q_m, \lambda_{m+1})$ exists and

$S^m(\sigma(\lambda_m, q_m, \lambda_{m+1})) = \sigma(\lambda, p, \lambda')/q_{k-1}\cdot\ldots\cdot q_{m+1}$.

Let T, U be the substitutions such that

(2) $\sigma(\lambda, p, \lambda') = U(\lambda') = U(\lambda')[p \leftarrow T(\lambda)]$.

Furthermore let T^m, U^m $(0 \leq m \leq k-1)$ be the substitutions such that

(3) $\sigma(\lambda_m, q_m, \lambda_{m+1}) = U^m(\lambda_{m+1}) = U^m(\lambda_{m+1})[q_m \leftarrow T^m(\lambda_m)]$.

For $0 \leq m \leq k-2$ we have

$S^m(U^m(\lambda_{m+1})) =$ (3)

$S^m(\sigma(\lambda_m, q_m, \lambda_{m+1})) =$ (1)

$\sigma(\lambda, p, \lambda')/q_{k-1}\cdot\ldots\cdot q_{m+1} =$ Lemma 2.2.1 (i)

$(\sigma(\lambda, p, \lambda')/q_{k-1}\cdot\ldots\cdot q_{m+2})/q_{m+1} =$ (1)

$(S^{m+1}(\sigma(\lambda_{m+1}, q_{m+1}, \lambda_{m+2})))/q_{m+1} =$ Lemma 2.2.1 (ii)

$S^{m+1}(\sigma(\lambda_{m+1}, q_{m+1}, \lambda_{m+2})/q_{m+1}) =$ (3)

$S^{m+1}(T^{m+1}(\lambda_{m+1}))$.

Therefore, by Lemma 2.2.2,

$\sigma(\lambda, p, \lambda')[q_{k-1}\cdot\ldots\cdot q_{m+1} \leftarrow S^m(U^m(\rho_{m+1}))] =$

$\sigma(\lambda, p, \lambda')[q_{k-1}\cdot\ldots\cdot q_{m+1} \leftarrow S^{m+1}(T^{m+1}(\rho_{m+1}))]$

for $0 \leq m \leq k-2$.

We introduce the notation

$\beta_{m+1} := \sigma(\lambda, p, \lambda')[q_{k-1}\cdot\ldots\cdot q_{m+1} \leftarrow S^m(U^m(\rho_{m+1}))] =$

$\sigma(\lambda, p, \lambda')[q_{k-1}\cdot\ldots\cdot q_{m+1} \leftarrow S^{m+1}(T^{m+1}(\rho_{m+1}))]$

for $0 \leq m \leq k-2$.

$T(\lambda) \underset{(2)}{=} \sigma(\lambda, p, \lambda')/p \underset{(1)}{=} (S^0(\sigma(\lambda_0, q_0, \lambda_1)))/q_0 \underset{(3)}{=}$

$(S^0(U^0(\lambda_1)[q_0 \leftarrow T^0(\lambda_0)]))/q_0 \underset{\text{Lemma 2.2.1 (ii)}}{=} S^0(T^0(\lambda_0)) = S^0(T^0(\lambda))$.

So by Lemma 2.2.2

(4) $T(\rho) = S^0(T^0(\rho)) = S^0(T^0(\rho_0))$.

$\tau_1(\lambda, \rho, p, \lambda', \rho') \underset{(2)}{=}$

$\sigma(\lambda, p, \lambda')[p \leftarrow T(\rho)] \underset{\text{Lemma 2.2.1 (iii)}}{=}$

$\sigma(\lambda, p, \lambda')[q_{k-1}\cdot\ldots\cdot q_1 \leftarrow \sigma(\lambda, q, \lambda')/q_{k-1}\cdot\ldots\cdot q_1\,[q_0 \leftarrow T(\rho)]] \underset{(1)}{=}$

$\sigma(\lambda, p, \lambda')[q_{k-1}\cdot\ldots\cdot q_1 \leftarrow S^0(\sigma(\lambda_0, q_0, \lambda_1))[q_0 \leftarrow T(\rho)]] \underset{(3)}{=}$

$\sigma(\lambda, p, \lambda')[q_{k-1}\cdot\ldots\cdot q_1 \leftarrow S^0(U^0(\lambda_1))[q_0 \leftarrow T(\rho)]] \underset{(4),\ \text{Lemma 2.2.1 (v)}}{=}$

$\sigma(\lambda, p, \lambda')[q_{k-1}\cdot\ldots\cdot q_1 \leftarrow \underbrace{S^0(U^0(\lambda_1)\,[q_0 \leftarrow T^0(\rho_0)])}_{\tau_1(\lambda_0, \rho_0, q_0, \lambda_1, \rho_1)}]$.

On the other hand

$\beta_1 = \sigma(\lambda, p, \lambda')[q_{k-1}\cdot\ldots\cdot q_1 \leftarrow \underbrace{S^0(U^0(\rho_1))}_{\tau_2(\lambda_0, \rho_0, q_0, \lambda_1, \rho_1)}]$.

So by assumption and Lemma 2.2.4 we have $\tau_1(\lambda, \rho, p, \lambda', \rho') \downarrow^*_R \beta_1$.

For $1 \leq m \leq k-2$ we have

$\beta_m = \sigma(\lambda, p, \lambda')[q_{k-1}\cdot\ldots\cdot q_m \leftarrow S^m(T^m(\rho_m))] \underset{\text{Lemma 2.2.1 (iii)}}{=}$

$\sigma(\lambda, p, \lambda')[q_{k-1}\cdot\ldots\cdot q_{m+1} \leftarrow \sigma(\lambda, p, \lambda')/q_{k-1}\cdot\ldots\cdot q_{m+1}\,[q_m \leftarrow S^m(T^m(\rho_m))]] \underset{(1),(3)}{=}$

$\sigma(\lambda, p, \lambda')[q_{k-1}\cdot\ldots\cdot q_{m+1} \leftarrow S^m(U^m(\lambda_{m+1}))[q_m \leftarrow S^m(T^m(\rho_m))]] \underset{\text{Lemma 2.2.1(v)}}{=}$

$\sigma(\lambda, p, \lambda')[q_{k-1}\cdot\ldots\cdot q_{m+1} \leftarrow \underbrace{S^m(U^m(\lambda_{m+1})[q_m \leftarrow T^m(\rho_m)])}_{\tau_1(\lambda_m, \rho_m, q_m, \lambda_{m+1}, \rho_{m+1})}]$,

and on the other hand

$\beta_{m+1} = \sigma(\lambda, p, \lambda')[q_{k-1}\cdot\ldots\cdot q_{m+1} \leftarrow \underbrace{S^m(U^m(\rho_{m+1}))}_{\tau_2(\lambda_m, \rho_m, q_m, \lambda_{m+1}, \rho_{m+1})}]$.

So by assumption and Lemma 2.2.4 we have $\beta_m \downarrow^*_R \beta_{m+1}$ for $1 \leq m \leq k-2$.

$$U(\lambda') \underset{(2)}{=} \sigma(\lambda,p,\lambda') \underset{(1)}{=} S^{k-1}(\sigma(\lambda_{k-1},q_{k-1},\lambda_k)) \underset{(3)}{=} S^{k-1}(U^{k-1}(\lambda_k)) = S^{k-1}(U^{k-1}(\lambda')).$$

So by Lemma 2.2.2

$$(5)\quad U(\rho') = S^{k-1}(U^{k-1}(\rho')) = S^{k-1}(U^{k-1}(\rho_k)).$$

$$\tau_2(\lambda,\rho,p,\lambda',\rho') \underset{(2)}{=} U(\rho') \underset{(5)}{=} S^{k-1}(\underbrace{U^{k-1}(\rho_k)}_{\tau_2(\lambda_{k-1},\rho_{k-1},q_{k-1},\lambda_k,\rho_k)}).$$

On the other hand

$$\beta_{k-1} = \sigma(\lambda,p,\lambda')[q_{k-1} \leftarrow S^{k-1}(T^{k-1}(\rho_{k-1}))] \underset{(1)}{=}$$

$$(S^{k-1}(\sigma(\lambda_{k-1},q_{k-1},\lambda_k)))[q_{k-1} \leftarrow S^{k-1}(T^{k-1}(\rho_{k-1}))] =$$

Lemma 2.2.1(v),(3)

$$S^{k-1}(\underbrace{U^{k-1}(\lambda_k)[q_{k-1} \leftarrow T^{k-1}(\rho_{k-1})]}_{\tau_1(\lambda_{k-1},\rho_{k-1},q_{k-1},\lambda_k,\rho_k)}).$$

So by assumption and Lemma 2.2.4 we have $\beta_{k-1} \downarrow^*_R \tau_2(\lambda,\rho,p,\lambda',\rho')$.

Let $\eta_o := \alpha[r \leftarrow S(\tau_1(\lambda,\rho,p,\lambda',\rho'))]$

$\eta_m := \alpha[r \leftarrow S(\beta_m)]$ *for* $1 \leq m \leq k-1$,

$\eta_k := \alpha[r \leftarrow S(\tau_2(\lambda,\rho,p,\lambda',\rho'))]$.

Then we have (by Lemma 2.2.4) $\eta_i \downarrow^*_R \eta_{i+1}$ *for* $0 \leq i \leq k-1$.

But $\alpha >_t \eta_i$ for all $0 \leq i \leq k$, since

$\alpha = \alpha[r \leftarrow S(\sigma(\lambda,p,\lambda'))]$

and $\sigma(\lambda,p,\lambda') >_t \tau_1(\lambda,\rho,p,\lambda',\rho')$, *so* $\alpha >_t \eta_o$,

$\sigma(\lambda,p,\lambda') >_t \tau_2(\lambda,\rho,p,\lambda',\rho')$, *so* $\alpha >_t \eta_k$,

and for $1 \leq m \leq k-1$

$$\sigma(\lambda_{m-1},q_{m-1},\lambda_m) \underset{(3)}{=} U^{m-1}(\lambda_m) >_t U^{m-1}(\rho_m),$$

so

$$\sigma(\lambda,p,\lambda') \underset{(1)}{=}$$

$$\sigma(\lambda,p,\lambda')[q_{k-1}\cdot\ldots\cdot q_m \leftarrow S^{m-1}(\sigma(\lambda_{m-1},q_{m-1},\lambda_m))] =$$

$$\sigma(\lambda,p,\lambda')[q_{k-1}\cdot\ldots\cdot q_m \leftarrow S^{m-1}(U^{m-1}(\lambda_m))] >_t$$

$$\sigma(\lambda,p,\lambda')[q_{k-1}\cdot\ldots\cdot q_m \leftarrow S^{m-1}(U^{m-1}(\rho_m))] = \beta_m$$

and therefore

$\alpha >_t \alpha[r \leftarrow S(\beta_m)] = \eta_m$.

This completes the proof of Theorem 2. •

Theorem 2 yields a criterion for eliminating unnecessary reductions in the K n u t h - B e n d i x algorithm in the following way: whenever for some element $(\lambda,\rho,p,\lambda',\rho') \in CP(R')$ one can find $(\lambda_o,\rho_o),\ldots,(\lambda_k,\rho_k) \in R'$, $q_o,\ldots,q_{k-1} \in \mathbb{N}^*$ such that

$$(\lambda_o,\rho_o)=(\lambda,\rho),\ (\lambda_k,\rho_k)=(\lambda',\rho'),\ p=q_{k-1}\cdot\ldots\cdot q_o,$$

$$(\forall\ 0 \leq m \leq k-1)\ (q_m \in occ(\lambda_{m+1}),$$

$$(\lambda_m,\rho_m,q_m,\lambda_{m+1},\rho_{m+1}) \in CP(R')-C,$$

$$\sigma(\lambda_m,q_m,\lambda_{m+1}) \preccurlyeq \sigma(\lambda,p,\lambda')/_{q_{k-1}\cdot\ldots\cdot q_{m+1}})$$

then one can just delete $(\lambda,\rho,p,\lambda',\rho')$ from C without reducing $\tau_1(\lambda,\rho,p,\lambda',\rho')$ and $\tau_2(\lambda,\rho,p,\lambda',\rho')$ modulo R'. However, it is difficult to handle this criterion algorithmically.

In [Bu79] it has been shown how a criterion of this kind can be used more easily by replacing the search for an intermediate chain by the test of the existence of only one suitable intermediate point. If combined with the strategy to consider first "least common multiples" of low degree ("superpositions" which are low in the subsumption oredering) this simplified form of the criterion is nearly as powerful as the original form. We apply the same technique here and, finally, arrive at the following criterion:

crit$(\lambda,\rho,p,\lambda',\rho',C,R)$ iff

there are no $(\lambda'',\rho'') \in R$, $q_o,q_1 \in \mathbb{N}^*$ such that

$p=q_1 \cdot q_o$,

$(\lambda,\rho,q_o,\lambda'',\rho'') \in CP(R)-C$, $(\lambda'',\rho'',q_1,\lambda',\rho') \in CP(R)-C$,

$\sigma(\lambda,q_o,\rho'') \preccurlyeq \sigma(\lambda,p,\lambda')_{/q_1}$, $\sigma(\lambda'',q_1,\lambda') \preccurlyeq \sigma(\lambda,p,\lambda')$.

For every tuple $(\lambda,\rho,p,\lambda',\rho')$ selected in the K n u t h - B e n d i x algorithm we can check this criterion and skip the reduction of the associated critical pair if it is satisfied.

5. EXAMPLE FOR THE KNUTH-BENDIX ALGORITHM WITH CRITERION

Let Σ consist of the three operator symbols f,g,e with arities $2,1,0$ and weights $0,0,1$, respectively. Let V be an infinite set of variables, which we denote $x,y,z,u,\ldots$. Let the terms T over Σ and V be ordered by the ordering $">_t"$ given in [KB67].

Now consider the reduction system R consisting of the following 7 reductions

(1) $f(g(f(x,e)),x) \rightarrow e$

(2) $g(f(x,y)) \rightarrow f(g(y),g(x))$

(3) $f(g(x),x) \rightarrow e$

(4) $f(f(x,g(y)),y) \rightarrow x$

(5) $f(g(x),f(x,y)) \rightarrow y$

(6) $f(f(x,y),z) \rightarrow f(x,f(y,z))$

(7) $g(g(x)) \rightarrow x$.

During the execution of the algorithm on R the following new reductions are added to the reduction system as reduction results of the specified superpositions:

(8) $g(e) \rightarrow e$ (λ_3 on λ_2 at 1)

(9) $f(e,x) \rightarrow x$ (λ_3 on λ_4 at 1)

(10) $f(x,e) \rightarrow x$ (λ_6 on λ_4 at Λ)

(11) $f(x,g(x)) \rightarrow e$ (λ_7 on λ_3 at 1)

(12) $f(x,f(g(x),y)) \rightarrow y$ (λ_{11} on λ_6 at 1).

69 superpositions have to be considered, 12 of which can be recognized as unnecessary by the criterion "crit", for instance

$\sigma(\lambda_1,1.1,\lambda_1) = f(g(f(g(f(e,e)),e)),g(f(e,e)))$

$\sigma(\lambda_1,1,\lambda_2) = g(f(g(f(x,e)),x)) \preccurlyeq \sigma(\lambda_1,1.1,\lambda_1)_{/1}$

$\sigma(\lambda_2,1,\lambda_1) = f(g(f(x,e)),x) \preccurlyeq \sigma(\lambda_1,1.1,\lambda_1)$.

In accordance with the experience in the case of polynomial ideals, [Bu79], we conjecture that the applicability of the criterion increases drastically with the complexity of the examples, in particular the "depth" of nesting in the terms. Computer experiments for studying this phenomenon are planned for the near future.

REFERENCES

[Ba82] D. B a y e r, The Division Algorithm and the Hilbert Scheme, Ph.D. thesis, *Harvard Univ., Math.Dept., Cambridge Mass.*, 1982.

[Bu65] B. B u c h b e r g e r, Ein Algorithmus zum Auffinden der Basiselemente des Restklassenringes nach einem nulldimensionalen Polynomideal, doctoral dissertation, *Universität Innsbruck, Austria*, 1965.

[Bu70] B. B u c h b e r g e r, *Ein algorithmisches Kriterium für die Lösbarkeit eines algebraischen Gleichungssystems*, Aequationes math. 4/3, 374-383, 1970.

[Bu76] B. B u c h b e r g e r, *A Theoretical Basis for the Reduction of Polynomials to Canonical Forms*, SIGSAM Bull. 10/3, 19-29, 1976.

[Bu79] B. Buchberger, A Criterion for Detecting Unnecessary Reductions in the Construction of Gröbner-Bases, *Proc. of the European Symposium on Symbolic and Algebraic Manipulation, Marseille, 1979, E.W.Ng, ed.*, Lecture Notes in Computer Science 72, 3-21, Springer-Verlag, 1979.

[BL82] B. Buchberger, R. Loos, Algebraic Simplification, in: *Computer Algebra - Symbolic and Algebraic Computation, B. Buchberger, G.E. Collins and R.Loos, eds.*, Computing Supplementum 4, 11-43, Springer-Verlag, 1982.

[Gu82] J. Guiver, Contributions to Two-Dimensional Systems Theory, Ph.D. thesis, *Univ. of Pittsburgh, Fac. of Arts and Sciences*, 1982.

[Hu80] G. Huet, *Confluent Reductions: Abstract Properties and Applications to Term Rewriting Systems*, J.ACM 27, 797-821, 1980.

[HO80] G. Huet, D. C. Oppen, Equations and Rewrite Rules - A survey, in: *Formal Language Theory, R.V. Book, ed.*, Academic Press, 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.

[Ll83] R. Llopis de Trias, Canonical Forms for Residue Classes of Polynomial Ideals and Term Rewriting Systems, *Univ. Aut. de Madrid, Division de Matematicas*, submitted to publication, 1983.

[Lo81] R. Loos, Term Reduction Systems and Algebraic Algorithms, *Proc. 5th GI Workshop on AI, Bad Honnef, 1981*, Informatik Fachberichte 47, 214-234, Springer-Verlag, 1981.

[MM83] F. Mora, H. M. Möller, The Computation of the Hilbert Function, *Proc. EUROCAL 83 Conference, London, March 1983*, Lecture Notes in Computer Science 162, 157-167, Springer-Verlag, 1983.

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

[PS81] G. E. Peterson, M. E. Stickel, *Complete Sets of Reductions for some Equational Theories*, J. ACM 28/2, 233-264, 1981.

[VW67] B. L. van der Waerden, Algebra II. 5th ed., *Springer-Verlag*, Berlin-Heidelberg-New York, 1967.

[Wi82] F. Winkler, Attempts to Establish a General Critical Pair - Completion Algorithm, *Univ. Linz. Inst. f. Math., CAMP 82-2.0*, 1982.

[Wi83] F. Winkler, A Criterion for Eliminating Unnecessary Reductions in the Knuth-Bendix Algorithm, *Univ, Linz, Inst. f. Math., CAMP 83-14.1*, 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., Univ. Linz, Austria*, 1984.

[WBLR81] F. Winkler, B. Buchberger, F. Lichtenberger, H. Rolletschek, An Algorithm for Constructing Canonical Bases (Groeber-Bases) for Polynomial Ideals, *Univ. Linz, Inst. f. Math., CAMP 81-10.0*, 1981.

[WBLR85] F. Winkler, B. Buchberger, F. Lichtenberger, H. Rolletschek, *An Algorithm for Constructing Canonical Bases of Polynomial Ideals*, to appear in ACM Trans. on Math. Software 11/1, 1985.

F. WINKLER and B. BUCHBERGER

Johannes Kepler Universität
Institut für Mathematik
Altenberger Str. 69.
4040 Linz
Austria