J. Symbolic Computation (2002) **33**, 43–55
doi:10.1006/jsco.2001.0478
Available online at http://www.idealibrary.com on IDEAL®

Continuously Parameterized Symmetries and Buchberger's Algorithm

RALF HEMMECKE†

Research Institute for Symbolic Computation, Johannes Kepler University, A-4040 Linz, Austria

Systems of polynomial equations often have symmetries. In solving such a system using Buchberger's algorithm, the symmetries are neglected. Incorporating symmetries into the solution process enables us to solve larger problems than with Buchberger's algorithm alone. This paper presents a method that shows how this can be achieved and also gives an algorithm that brings together continuously parameterized symmetries with Buchberger's algorithm.

1. Introduction

The Gröbner basis technique has proved to be an indispensable tool in many areas. For the basic concepts, we refer to Buchberger (1985) and Becker and Weispfenning (1993). Although it has been refined in various ways, many problems which are solvable in principle by the Gröbner basis technique remain practically intractable due to the double-exponential worst-case complexity of Buchberger's algorithm. One way to try and tackle such problems is to use additional information which is normally neglected during a Gröbner basis computation.

In this paper, we investigate problems which are additionally invariant under a linear group action. To describe our aim in detail, let K be an algebraically closed field and $B = \{b_1, \ldots, b_p\} \subset K[A_1, \ldots, A_n]$. Furthermore, let Γ be a group parameterized by the affine space $\mathcal{X} = K^m$: i.e. an element γ of Γ can be written in the form $\gamma = \gamma(x)$ with $x \in \mathcal{X}$. We assume that Γ acts on $\mathcal{A} = K^n$, and this action can be described by polynomials $\gamma_1, \ldots, \gamma_n \in K[X_1, \ldots, X_m, A_1, \ldots, A_n]$, such that

$$(\forall x \in \mathcal{X})(\forall a \in \mathcal{A}) \quad \gamma(x) \cdot a = (\gamma_1(x, a), \ldots, \gamma_n(x, a)) \in \mathcal{A}. \tag{1}$$

In addition, we require that the **set of common zeros**

$$\mathcal{Z}(B) := \{a \in \mathcal{A} \mid b_1(a) = \cdots = b_p(a) = 0\}$$

is invariant under the group action of Γ, and we are looking for the description of $\mathcal{Z}(B)$.

The main idea to simplify the original problem is to determine a subset $Z \subset \mathcal{Z}(B)$ which is easy to compute and such that $\mathrm{Orb}(Z)$, the Γ-orbits of $z \in Z$, covers $\mathcal{Z}(B)$. Here, we try to take Z as the intersection of $\mathcal{Z}(B)$ with appropriate coordinate-hyperplanes. For such a choice, however, $\mathrm{Orb}(Z)$ need not cover $\mathcal{Z}(B)$. Hence, the problem is divided

†E-mail: ralf@hemmecke.de

0747–7171/02/010043 + 13 $35.00/0

into a description of Z and of the "exceptional set" $Z' = \mathcal{Z}(B) \setminus \mathrm{Orb}(Z)$. Both may be found with a variant of the Gröbner factorizer algorithm.

It turns out that we can weaken the assumptions about Γ: we can take a polynomially parameterized set of transformations of $\mathcal{A}$ leaving $\mathcal{Z}(B)$ invariant, which may be inverted effectively in each point of $\mathcal{Z}(B)$ to recover the orbits.

Further investigation has to be done to choose Z in such a way that Z' inherits a "nice" structure, since the decomposition of this "exceptional set" consumes most of the time of the whole computation.

2. Preliminaries

In this paper, K will be an algebraically closed field. $K[A_1, \ldots, A_n]$ is abbreviated by $K[A]$. R denotes a Euclidean ring (though this condition can be weakened in some places), and Q its field of fractions. In many cases, we take $R = K[A]$. We write $R[X]$ for the polynomial ring $R[X_1, \ldots, X_m]$. For the natural numbers m and n, the affine spaces K^m and K^n are denoted by $\mathcal{X}$ and $\mathcal{A}$, respectively.

We follow the notation in Becker and Weispfenning (1993) and denote by $C(f)$, $T(f)$, and $M(f)$ the set of coefficients, terms, and monomials of f, respectively. These sets are empty if $f = 0$. $T(X_1, \ldots, X_m)$, abbreviated by $T(X)$, is the set of all terms in these indeterminates, i.e. monomials with coefficient 1.

Let us fix an arbitrary Noetherian term order throughout the paper. We use LC, LT, and LM as usual, and denote for $F \subseteq R[X]$ the set $\{\mathrm{LT}(f) \mid 0 \neq f \in F\}$ by $\mathrm{LT}(F)$.

$\mathcal{I}(B)$ is the ideal generated by the elements of B. It will always follow from the context in which polynomial ring this ideal is generated.

Definition 2.1. Let $f, \tilde{f}, f^*, g \in R[X]$, and $G = \{g_1, \ldots, g_k\} \subset R[X]$.

(i) f is **pseudo-reducible modulo** g **to** $\tilde{f}$ (written: $f \to_g \tilde{f}$) if $f, g \neq 0$, and there exists $r_0 t_0 \in M(f)$ such that $\mathrm{LT}(g) | t_0$, and

$$\tilde{f} = \tilde{r} f - \tilde{r}_0 t g$$

where

$$\varrho = \gcd(r_0, \mathrm{LC}(g)), \qquad \tilde{r} = \frac{\mathrm{LC}(g)}{\varrho}, \qquad \tilde{r}_0 = \frac{r_0}{\varrho}, \qquad t = \frac{t_0}{\mathrm{LT}(g)}.$$

(ii) f is **pseudo-reducible modulo** G **to** $\tilde{f}$ (written: $f \to_G \tilde{f}$) if $f \to_g \tilde{f}$ for some $g \in G$.

(iii) f is called **pseudo-reducible modulo** G if there is some $\tilde{f} \in R[X]$, such that f is pseudo-reducible modulo G to $\tilde{f}$.

(iv) f^* is a **pseudo-normal form of** f **modulo** G iff f^* is not pseudo-reducible modulo G and $f \to_G^* f^*$ holds.

By $\to_G^*$ we denote the reflexive transitive closure of $\to_G$.

Recall that, when $R = Q$ is a field, the normal form of a polynomial w.r.t. pseudo-reduction differs only in a constant factor (from R) from the normal form w.r.t. the usual concept of reduction as, for example, described in Becker and Weispfenning (1993).

In particular, reducibility and pseudo-reducibility modulo G coincide. Hence, 0 is a normal form of a polynomial f modulo G iff 0 is a pseudo-normal form of f modulo G.

Pseudo-normal forms can be computed with the algorithm **pseudoNormalForm** given below. It incorporates the obvious changes to the usual normal form algorithm (as, for example, presented in Becker and Weispfenning (1993)) necessary for denominator-free computations.

A Gröbner basis G of an ideal I of $Q[X]$ is called **denominator-free** (w.r.t. R) if $G \subset R[X]$.

For $g_1, g_2 \in R[X] \setminus \{0\}$ we define the **S-polynomial** in a denominator-free way by

$$\mathrm{spol}(g_1, g_2) := \frac{\mathrm{LM}(g_2)\, g_1 - \mathrm{LM}(g_1)\, g_2}{\gcd(\mathrm{LT}(g_1), \mathrm{LT}(g_2))} \in R[X].$$

For later reference, we now present a pseudo-normal form algorithm and Buchberger's algorithm in a denominator-free form.

Algorithm pseudoNormalForm
Input:
$f \in R[X]$
$G = \{g_1, \ldots, g_k\} \subset R[X]$
Output:
$f^* \in R[X]$, such that f^* is a pseudo-normal form of f modulo G.
$r \in R$ and $H = \{h_1, \ldots, h_k\} \subset R[X]$, such that $\mathrm{LT}(h_i g_i) \leq \mathrm{LT}(f)$ holds for all $1 \leq i \leq k$ if the corresponding leading terms are defined, and

$$rf = f^* + \sum_{i=1}^{k} h_i g_i.$$

begin
$f^* := f$
$r := 1$
Let $h_i := 0$ for all $i = 1, \ldots, k$.
while f^* (pseudo-)reducible modulo G do
  Choose $1 \leq i_0 \leq k$ such that $f^* \to_g \tilde{f}$ where $g = g_{i_0}$.
  If $r_0 t_0 \in M(f^*)$ is the monomial that will be replaced,
  choose $\tilde{r}$, $\tilde{r}_0$, t as in Definition 2.1, i.e.
    $\varrho := \gcd(r_0, \mathrm{LC}(g))$
    $\tilde{r} := \mathrm{LC}(g)/\varrho$
    $\tilde{r}_0 := r_0/\varrho$
    $t := t_0/\mathrm{LT}(g)$
  Update
    $f^* := \tilde{r} f^* - \tilde{r}_0 t g$
    $h_{i_0} := \tilde{r} h_{i_0} + \tilde{r}_0 t$
    $h_i := \tilde{r} h_i$ for $1 \leq i \leq k$, $i \neq i_0$
    $r := \tilde{r} r$
end while
return (f^*, r, H)
end pseudoNormalForm

Algorithm buchberger
Input:
B = finite subset of $R[X]$
Output:
G = finite subset of $R[X]$, such that G is a denominator-free Gröbner basis of $\mathcal{I}(B)$ in $Q[X]$.

begin
$G := B$
$P := \{(g_1, g_2) \mid g_1, g_2 \in G,\ g_1 \neq g_2\}$
while $P \neq \emptyset$ do
  Choose (g_1, g_2) from P
  $P := P \setminus \{(g_1, g_2)\}$
  $g := \mathrm{spol}(g_1, g_2)$
  $(g, r, H) :=$ **pseudoNormalForm**(g, G)
  if $g \neq 0$ then
    $g := \mathrm{primitivePart}(g) \in R[X]$
    $P := P \cup \{(\bar{g}, g) \mid \bar{g} \in G\}$
    $G := G \cup \{g\}$
  end if
end while
return G
end buchberger

Buchberger's algorithm has been refined in various ways. One method is to factor the normal forms of produced S-polynomials and split the corresponding problem according to these factors into several "simpler" branches. Gräbe presents in Gräbe (1995b) a form of the Buchberger-algorithm with factorization (called **FGB**) which will be used here.

Given a set $B = \{b_1, \ldots, b_p\} \subset K[A]$ and a set $C = \{c_1, \ldots, c_q\} \subset K[A]$ of "constraints", the algorithm **FGB** determines the set $\mathcal{Z}(B, C)$ of common zeros by returning a certain number of Gröbner bases B_i and corresponding "constraints" C_i such that $\mathcal{Z}(B, C) = \bigcup_i \mathcal{Z}(B_i, C_i)$ holds. Here $\mathcal{Z}(B, C) = \mathcal{Z}(B) \setminus \mathcal{Z}(c)$ with $c = \prod_{f \in C} f$ is the **relative set of common zeros of** B **w.r.t.** C **in** $\mathcal{A}$.

3. Gröber Bases and Specialization

Let $R = K[A]$, fix some elements $a_1, \ldots, a_n \in K$, and let $\sigma : R[X] \to K[X]$ be the **specialization** induced by $A_i \mapsto a_i$. For $B = \{b_1, \ldots, b_p\}$ the set $\{b_1^\sigma, \ldots, b_p^\sigma\}$ is denoted by B^σ.

Lemma 3.1. *Let $G = \{g_1, \ldots, g_k\} \subset R[X]$ be a denominator-free Gröbner basis of $\mathcal{I}(G)$ in $Q[X]$, σ a specialization, and $f \in \mathcal{I}(G) \cap R[X]$. If $\mathrm{LC}(g)^\sigma \neq 0$ for all $g \in G$ then $f^\sigma \in \mathcal{I}(G^\sigma) \subset K[X]$.*

Proof. Since $f \in \mathcal{I}(G)$ and G is a Gröbner basis, we have $f \to_G^* 0$. Now consider the algorithm **pseudoNormalForm** with input f and G. The algorithm yields a relation

$$rf = \sum_{i=1}^{k} h_i g_i \tag{2}$$

where $h_1, \ldots, h_k \in R[X]$ and $r \in R$ divides a product of $\{\mathrm{LC}(g) \mid g \in G\}$. Hence, $r^\sigma \neq 0$ is K-invertible and $f^\sigma \in \mathcal{I}(G^\sigma)$. □

THEOREM 3.2. *Let $B = \{b_1, \ldots, b_p\} \subset R[X]$, and let σ be a specialization. Let B in $Q[X]$ generate the proper ideal $\mathcal{I}(B)$, and let $G = \{g_1, \ldots, g_k\}$ be a denominator-free Gröbner basis of $\mathcal{I}(B)$.*

If $\mathrm{LC}(g_i)^\sigma \neq 0$, i.e. $\mathrm{LC}(g_i)^\sigma = \mathrm{LC}(g_i^\sigma)$ and $\mathrm{LT}(g_i) = \mathrm{LT}(g_i^\sigma)$, for all $i = 1, \ldots, k$, then G^σ generates a proper ideal in $K[X]$, and we have

$$\mathcal{I}(B^\sigma) \subseteq \mathcal{I}(G^\sigma) \qquad (\subset K[X]). \tag{3}$$

PROOF. $\mathcal{I}(G^\sigma)$ is a proper ideal of $K[X]$, since $\mathrm{LT}(G) = \mathrm{LT}(G^\sigma)$ and G is a Gröbner basis. For each $b \in B$ we have $b \in \mathcal{I}(G)$, since G is a Gröbner basis of $\mathcal{I}(B)$. From Lemma 3.1, it follows that $b^\sigma \in \mathcal{I}(G^\sigma)$, and hence also $\mathcal{I}(B^\sigma) \subseteq \mathcal{I}(G^\sigma)$. □

For a proper ideal I in $K[X]$ the Hilbert Nullstellensatz yields

$$\emptyset \neq \mathcal{Z}(I) \subseteq \mathcal{X}.$$

Therefore, we can state:

COROLLARY 3.3. *Under the same assumptions as in Theorem* 3.2*, we have*

$$\emptyset \neq \mathcal{Z}(G^\sigma) \subseteq \mathcal{Z}(B^\sigma) \subseteq \mathcal{X},$$

i.e. the set of zeros of B^σ is non-empty.

4. Un Nouvel Algorithme

4.1. ORIGIN

The motivating origin of the investigation presented here is the complete solution of the constant quantum Yang–Baxter equation in the two-dimensional case in Hietarinta (1992). Our paper formalizes and generalizes the method used by Hietarinta, and presents an algorithm which can then also be used to attack similar problems.

Hietarinta applied the Gröbner basis technique with factorization for his solution. Using the problem formulation introduced at the beginning of this paper, Hietarinta's method to employ inherent "continuous" symmetries can be formalized in the following way.

Choose a subset of $\{1, \ldots, n\}$, w.l.o.g. we take $\{1, \ldots, l\}$, and assume that for $a \in \mathcal{A}$ there is a $\gamma \in \Gamma$ such that the first l coordinates of $\gamma \cdot a$ vanish. For a fixed $a \in \mathcal{A}$, this can be decided by investigating the solvability of the system $\gamma_1(X, a) = \cdots = \gamma_l(X, a) = 0$.

Thus, the problem splits in the following way:

(i) Describe the set $Z = \mathcal{Z}(B) \cap \{a \in \mathcal{A} \mid a_1 = \cdots = a_l = 0\}$.
(ii) Describe the set $Z' = \mathcal{Z}(B) \setminus \mathrm{Orb}(\{a \in \mathcal{A} \mid a_1 = \cdots = a_l = 0\})$.

The first set provides whole Γ-orbits for the solution set. To each point $z \in Z$ corresponds a whole orbit $\mathrm{Orb}(z) \subset \mathcal{Z}(B)$ which can be determined by the parametrization of the Γ-action.

The second set is given by the system B and the unsolvability condition of $\Delta(a) := \{\gamma_1(X, a), \ldots, \gamma_l(X, a)\}$, i.e. may be described as $\mathcal{Z}(B) \cap \{a \in \mathcal{A} \mid \mathcal{I}(\Delta(a)) = K[X]\}$.

4.2. THE ALGORITHM GAMMA

In this section we present an algorithm for the desired task.

Algorithm gamma
Input:
$B = \{b_1, \ldots, b_p\} \subset R = K[A]$
l = number of vanishing coordinates as described above
Γ = group parameterized by $\mathcal{X}$, acting on $\mathcal{A}$ and leaving $\mathcal{Z}(B)$ invariant. The group action is given by $\gamma_1, \ldots, \gamma_n \in R[X]$, i.e. relation (1) holds.
Output:
$S_\nu = \{(B_i, C_i)\}_{i \in I_\nu}$ Gröbner bases with corresponding constraints for certain finite index sets I_0 and I_1 where $I_0 \cap I_1 = \emptyset$, such that

$$\mathcal{Z}(B) = \bigcup_{i \in I_0} \mathcal{Z}(B_i, C_i) \cup \bigcup_{i \in I_1} \mathrm{Orb}(\mathcal{Z}(B_i, C_i)).$$

All B_i and C_i are finite subsets of $K[A]$, and the orbit is taken w.r.t. Γ.

begin
$\Delta := \{\gamma_1, \ldots, \gamma_l\}$
$G :=$ **buchberger**(Δ), (*use* $R = K[A]$)
(G is now a denominator-free Gröbner basis.)
if $\mathcal{I}(G) = Q[X]$ then
(No usage of the group action is possible, and thus no problem reduction.)
$S_0 :=$ **FGB**$(B, \emptyset)$
$S_1 := \emptyset$
else
$r_1^{\nu_1} \cdot \cdots \cdot r_u^{\nu_u} = \prod_{g \in G} \mathrm{LC}(g)$ (Z1)
(decomposition into (irreducible) factors in $R = K[A]$*)*
$S_0 := \bigcup_{i=1}^{u}$ **FGB**$(B \cup \{r_i\}, \{r_1, \ldots, r_{i-1}\})$
$S_1 :=$ **FGB**$(B \cup \{A_1, \ldots, A_l\}, \{r_1, \ldots, r_u\})$
end if
return (S_0, S_1)
end gamma

Since we have m parameters, namely $x_1, \ldots, x_m$, $l = m$ would be an optimum for the input in **gamma**. However, this may not be possible as the example below will show.

For $a \in \mathcal{A}$ we try to find out, whether there exists an $x \in \mathcal{X}$ such that $\gamma_1(x, a) = \cdots = \gamma_l(x, a) = 0$. For a fixed a, such an x exists iff the set $\Delta(a)$ generates a proper ideal in $K[X]$. In order to avoid a Gröbner basis computation for each single a, we determine a Gröbner basis of the ideal of $Q[X]$ generated by Δ. Theorem 3.2 gives sufficient conditions that after a specialization σ of Δ, the set $\Delta^\sigma = \Delta(a)$ generates a proper ideal in $K[X]$.

Theorem 3.2 yields even more. A naive algorithm would have to collect all the leading coefficients of the polynomials occurring during the Gröbner basis computation of Δ to ensure that no information is lost, when, in the algorithm **buchberger**, a polynomial g is replaced by its primitive part primitivePart(g), and thus that no wrong decision about the existence of an x for a certain a is made. Such a collection is unnecessary. As follows from Theorem 3.2, it is sufficient to consider the leading coefficients of a minimal Gröbner basis only.

The set S_1 (together with the group action of Γ) describes the part of $\mathcal{Z}(B)$ where no leading coefficient of G vanishes. In other words, if $(B_0, C_0) \in S_1$, $a \in \mathcal{Z}(B_0, C_0)$, and $\sigma : R[X] \to K[X]$ is induced by $A_i \mapsto a_i$ $(i = 1, \ldots, n)$, then $\mathrm{LC}(g)^\sigma \neq 0$ holds for all $g \in G$. In this case, Γ can successfully be used for the reduction of the problem. Otherwise, we extend the set B by an additional polynomial r_i and thus (hopefully) decrease the dimension.

LEMMA 4.1. *The algorithm* **gamma** *terminates and meets its specification.*

PROOF. Termination is obvious. Correctness follows from the correctness of **FGB**. Consider the else branch. S_ν is a set of pairs. Assume $S_\nu = \{(B_i, C_i)\}_{i \in I_\nu}$ $(\nu = 0, 1)$ for certain finite index sets I_0 and I_1 with $I_0 \cap I_1 = \emptyset$. We have to prove

$$\mathcal{Z}(B) = \bigcup_{i \in I_0} \mathcal{Z}(B_i, C_i) \cup \bigcup_{i \in I_1} \mathrm{Orb}(\mathcal{Z}(B_i, C_i)). \tag{4}$$

By specification of **FGB**, we have the identities

$$\bigcup_{i \in I_0} \mathcal{Z}(B_i, C_i) = \bigcup_{i=1}^{u} \mathcal{Z}(B \cup \{r_i\}, \{r_1, \ldots, r_{i-1}\}), \tag{5}$$

$$\bigcup_{i \in I_1} \mathcal{Z}(B_i, C_i) = \mathcal{Z}(B \cup \{A_1, \ldots, A_l\}, \{r_1, \ldots, r_u\}), \tag{6}$$

which correspond to the last two lines in the else branch.

The inclusion "$\supseteq$" follows from (5) and (6) and the Γ-invariance of $\mathcal{Z}(B)$.

For the opposite inclusion, it is sufficient to describe an element $a \in \mathcal{Z}(B)$ in terms of S_0 or S_1. If $r_j(a) = 0$ for some $j \in \{1, \ldots, u\}$, we can easily see that by (5) this a is covered by S_0. Consider the line (Z1) in **gamma** and suppose $r_i(a) \neq 0$ for all $i = 1, \ldots, u$. For the specialization σ induced by $A_i \mapsto a_i$ $(i = 1, \ldots, n)$, we have $\mathrm{LC}(g)^\sigma \neq 0$ for all $g \in G$. From Corollary 3.3 we get $\gamma_1(x, a) = \cdots = \gamma_l(x, a) = 0$ for some $x \in \mathcal{X}$. Therefore, a is covered by the set S_1. □

4.3. GENERALIZATION

If we look at the algorithm **gamma** more closely, we observe that the invertibility of an element of Γ is only needed to deduce the full set of solutions from S_1. Therefore, we can weaken the assumptions about Γ. Instead of requiring a group to describe the symmetries, we now only assume that we have a set of polynomials $\gamma_1, \ldots, \gamma_n \in K[X, A]$ such that

$$(\forall x \in \mathcal{X})(\forall a \in \mathcal{Z}(B)) \quad \gamma(x, a) := (\gamma_1(x, a), \ldots, \gamma_n(x, a)) \in \mathcal{Z}(B). \tag{7}$$

These polynomials are used as before.

In order to be able to deduce the complete solution space from S_1, we additionally require a set $\gamma'_1, \ldots, \gamma'_n \in K[X, A]$ of polynomials which reverses the effect of the transformation of the γ_i, i.e.

$$(\forall x \in \mathcal{X})(\forall a \in \mathcal{Z}(B))(\forall i \in \{1, \ldots n\}) \quad \gamma'_i(x, \gamma(x, a)) = a_i. \tag{8}$$

We now replace in the specification of the algorithm **gamma** the input parameter Γ by

- $\Gamma = (\gamma_1, \ldots, \gamma_n) \in K[X, A]^n$, n-tuple of polynomials such that (7) holds;

- $\Gamma' = (\gamma'_1, \ldots, \gamma'_n) \in K[X, A]^n$, n-tuple of polynomials such that (8) holds.

The output specification is replaced by

- $S_\nu = \{(B_i, C_i)\}_{i \in I_\nu}$ Gröbner bases with corresponding constraints for certain index sets I_0 and I_1 with $I_0 \cap I_1 = \emptyset$ such that

$$\mathcal{Z}(B) = \bigcup_{i \in I_0} \mathcal{Z}(B_i, C_i) \cup \bigcup_{i \in I_1} (\mathcal{Z}(B_i, C_i))^{\Gamma'}.$$

  All B_i and C_i are finite subsets of $K[A]$, and for a subset $Z \subseteq \mathcal{A}$ let

$$Z^{\Gamma'} := \{(\gamma'_1(x, a), \ldots, \gamma'_n(x, a)) \mid x \in \mathcal{X}, a \in Z\}.$$

We call this changed algorithm **gamma**, too. Correctness of this algorithm follows from the proof of Lemma 4.1, when its reference to the group action is replaced by the transformations Γ and Γ'.

5. Example

In this section we ask for all automorphisms of a given Lie algebra. We show how we can obtain a context which enables us to apply **gamma**. With the help of an example, we shall demonstrate how the algorithm works.

5.1. automorphisms of Lie algebras

For better legibility we use Einstein notation, i.e. summation over doubly occurring indices from 1 to N. Additionally, we use concepts from the theory of Lie algebras which can be found, for example, in Humphreys (1980).

Let V be an N-dimensional Lie algebra over K with vector space basis $\{e_1, \ldots, e_N\}$, whose Lie bracket is given by structural constants such that for $1 \le j, k \le N$ we have $[e_j, e_k] = c^l_{jk} e_l$. An endomorphism of this Lie algebra is a linear transformation $\hat{A} : V \to V$ such that $\hat{A}[v, v'] = [\hat{A}v, \hat{A}v']$. Let $A = (A^k_j)$ be the corresponding matrix of $\hat{A}$. By linear algebra, and due to the antisymmetry of the Lie bracket, we deduce that the endomorphisms correspond to the set of zeros of

$$F := \{A^l_m c^m_{jk} - A^p_j A^q_k c^l_{pq} \mid 1 \le j, k, l \le N, j < k\}. \tag{9}$$

To solve this system, we can use the fact that the composition of an endo- and an automorphism is again an endomorphism, and that we can recover the original endomorphism in an analogous way.

From the theory of Lie algebras, it is known that for each $v \in V$ the map $\mathrm{ad}(v) : V \to V$, $v' \mapsto [v, v']$ is a derivation. Let us only regard $\mathrm{char} K = 0$, and assume that $\mathrm{ad}(v)^s = 0$ for some $s > 0$. An argument from Humphreys (1980, p. 9) proves $\exp \mathrm{ad}(v)$ to be a Lie algebra automorphism.

Without loss of generality, we assume that $\mathrm{ad}(e_i)$ is nilpotent for $e_1, \ldots, e_m$. Denote the matrix which corresponds to $\exp(X_i, \mathrm{ad}(e_i))$ by $\Gamma_i(X_i)$.

Let the matrices Γ and Γ' be defined as follows:

$$\Gamma := \begin{pmatrix} \gamma^1_1 & \cdots & \gamma^1_N \\ \vdots & & \vdots \\ \gamma^N_1 & \cdots & \gamma^N_N \end{pmatrix} := A \cdot \Gamma_1(X_1) \cdot \cdots \cdot \Gamma_m(X_m),$$

$$\Gamma' := (\gamma'^j_k) := A \cdot (\Gamma_1(X_1) \cdot \cdots \cdot \Gamma_m(X_m))^{-1} = A \cdot \Gamma_m(-X_m) \cdot \cdots \cdot \Gamma_1(-X_1).$$

Putting $n := N^2$ and fixing an order of the index pairs, we obtain polynomials $\gamma_1, \ldots, \gamma_n$ and $\gamma'_1, \ldots, \gamma'_n$ in $A_1, \ldots, A_n, X_1, \ldots, X_m$. By this construction, it is clear that relation (8) is fulfilled.

Patera *et al.* (1976) classifies low-dimensional Lie algebras. We pick out one of them to demonstrate the behaviour of the algorithm **gamma**. A detailed treatment of another example can be found in Hemmecke (1996).

The indeterminates A^i_j are ordered w.r.t. the relation $\prec$. For $i, j, k, l = 1, \ldots, N$ let $A^i_j \prec A^k_l$ iff one of the following conditions is fulfilled:

(i) $|i-j| < |k-l|$
(ii) $|i-j| = |k-l|$ and $i+j < k+l$
(iii) $|i-j| = |k-l|$ and $i+j = k+l$ and $i < k$.

This corresponds to the intention to order the indices in such a way that the indeterminates of the main diagonal are less than the others.

We use this order on the variables to define the lexicographical term order on the terms $T(A^i_j : 1 \le i, j \le N)$. From now on we shall always use such a term order.

For our calculations we use routines from the REDUCE[†] package CALI[‡]. By employing some functions of CALI, we implemented the algorithm **gamma**[§] in REDUCE.

5.2. the Lie algebra $A_{4,7}$

In this section, K denotes the complex numbers. We examine the four-dimensional complex Lie algebra V ($A_{4,7}$ in Patera's notation) which is given by the non-zero commutator relations

$$[e_2, e_3] = e_1, \qquad [e_1, e_4] = 2e_1, \qquad [e_2, e_4] = e_2, \qquad [e_3, e_4] = e_2 + e_3$$

between the basis vectors.

The set F corresponding to (9) contains 22 polynomials. After autoreducing in a denominator-free way, we obtain

$$\begin{aligned} F' := \{&A^2_1, A^3_1, A^4_1, A^4_2, A^4_3, \quad A^4_4A^1_1 - A^1_1, \quad A^3_2A^4_4 - A^3_2, \quad A^3_2 + A^4_4A^2_2 - A^2_2, \\ &A^3_2A^2_3 - A^3_3A^2_2 + A^1_1, \quad -A^3_2 + A^4_4A^3_3 - A^3_3, \quad A^2_4A^3_2 - A^3_4A^2_2 - 2A^1_2A^4_4 + A^1_2, \\ &A^3_2 + A^2_3A^4_4 - A^2_3 + A^3_3 - A^2_2, \quad -A^2_4A^3_3 + 2A^1_3A^4_4 - A^1_3 + A^3_4A^2_3 - A^1_2\}. \end{aligned}$$

Although the indeterminate A^1_4 occurs in F, it does not in F'. We remove this indeterminate from further examination; and we remove the variables A^2_1, A^3_1, A^4_1, A^4_2, A^4_3, since from F' it can easily be deduced that their corresponding coordinates vanish. Hence, it is sufficient to describe the set of zeros of $B := F' \setminus \{A^2_1, A^4_3, A^4_1, A^3_1, A^4_2\}$.

We now try to determine l and the corresponding coordinates for the algorithm **gamma**. The derivations $\mathrm{ad}(e_i)$, ($i = 1, 2, 3$) are nilpotent. We denote the matrices

$$\begin{pmatrix} 1 & 0 & 0 & 2x \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \qquad \begin{pmatrix} 1 & 0 & x & 0 \\ 0 & 1 & 0 & x \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \qquad \begin{pmatrix} 1 & -x & 0 & -\frac{1}{2}x^2 \\ 0 & 1 & 0 & x \\ 0 & 0 & 1 & x \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

corresponding to $\exp(x,\ \mathrm{ad}\ e_i)$ by $M_1(x)$, $M_2(x)$, and $M_3(x)$, respectively.

[†]For documentation of the used version 3.4.1 see Hearn and Melenk (1991).
[‡]We use version 2.2.1. See Gräbe (1995a).
[§]The implementation is described in Hemmecke (1996).

Since the indeterminate A_4^1 does not occur in B, there is no advantage in examining its corresponding coordinate further. The variable x, however, occurs only in position $(1,4)$ of the matrix $A \cdot M_1(x)$. For that reason, the matrix $M_1(x)$ has no effect.

We set $\Gamma_1(X_1) := M_2(X_1)$ and $\Gamma_2(X_2) := M_3(X_2)$, and define with $m = 2$ the matrices Γ and Γ' as in the previous section.

Considering the fact that certain coordinates vanish, namely those corresponding to the indeterminates A_1^2, A_1^3, A_1^4, A_2^4, and A_3^4, we obtain from Γ the matrix

$$\begin{pmatrix} A_1^1 & -A_1^1X_2 + A_2^1 & A_1^1X_1 + A_3^1 & \gamma_4^1 \\ 0 & A_2^2 & A_3^2 & A_2^2X_1 + (A_2^2 + A_3^2)X_2 + A_4^2 \\ 0 & A_2^3 & A_3^3 & A_2^3X_1 + (A_2^3 + A_3^3)X_2 + A_4^3 \\ 0 & 0 & 0 & A_4^4 \end{pmatrix},$$

by replacing the above indeterminates by zero.

Since we have two parameters, namely X_1 and X_2, we can choose two of the polynomials γ_j^i. So let $l = 2$ for **gamma**, and let A_2^1 and A_3^1 be the first two coordinates, i.e. $\gamma_1 := \gamma_2^1$, $\gamma_2 := \gamma_3^1$.

We compare the algorithms **gamma** and **FGB** (implemented in CALI by the procedure `groebfactor`).

`groebfactor` returns for the input set B three sets of polynomials. While two of them describe non-invertible endomorphisms, from the third we get

$$\begin{pmatrix} a_1^2 & -a_1a_5 & a_1a_4 - a_1a_5 - a_2a_5 & a_3 \\ 0 & a_1 & a_2 & a_4 \\ 0 & 0 & a_1 & a_5 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad (a_1, \ldots, a_5 \in K,\ a_1 \neq 0).$$

It is clear how Γ and Γ' are put into a suitable form for the algorithm **gamma**. (Remember: $\gamma_1 = \gamma_2^1$ and $\gamma_2 = \gamma_3^1$.) We start this algorithm with the arguments B, $l = 2$, Γ and Γ'. First, there will be a Gröbner basis computation of $\{\gamma_1, \gamma_2\}$ in the ring $Q[X_1, X_2]$ where $R = K[A]$ is the polynomial ring in the remaining indeterminates, and Q its quotient field. The minimal denominator-free Gröbner basis from this computation is: $\{A_1^1 X_2 - A_2^1, A_1^1 X_1 + A_3^1\}$.

Referring to the notations of algorithm **gamma**, we now have $u = 1$, $r_1 = A_1^1$, and $\nu_1 = 2$. That is, for all $a \in \mathcal{Z}(B)$ with coordinate $a_1^1 \neq 0$ there are $x_1, x_2 \in K$ such that $\gamma_2^1(x_1, x_2; a) = \gamma_3^1(x_1, x_2; a) = 0$. In order to cover such an a, therefore, it is sufficient to look only for elements of $\mathcal{Z}(B)$ with vanishing coordinates in positions $(1,2)$ and $(1,3)$, and later to apply the transformation given by Γ'. In this way, S_1 will be determined.

It remains to describe the elements $a \in \mathcal{Z}(B)$ for which $a_1^1 = 0$. This condition is incorporated by determining a Gröbner basis of the enlarged set $B \cup \{A_1^1\}$. Thus, we end up with a problem reduction in this case, too, since, practically, the Gröbner basis computation is started with a smaller number of variables. In this way, the set S_0 will be calculated.

The result of **gamma** in our example is the following sets:

$$\begin{aligned} S_0 = \{&(\{A_1^1, A_2^2, A_2^3, A_2^1, A_3^3, A_3^2, 2A_4^4 - 1\}, \{A_3^1, A_4^4 - 1, A_4^4\}), \\ &(\{A_3^3, A_1^1, A_2^2, A_2^3, A_2^1, A_3^1 + A_4^3A_3^2, A_4^4 - 1\}, \{A_4^4\}), \\ &(\{A_1^1, A_2^2, A_2^3, A_2^1, A_3^3, A_3^2A_3^1\}, \emptyset)\}, \\ S_1 = \{&(\{A_2^1, A_3^1, A_2^3, A_4^3, A_4^2, (A_2^2)^2 - A_1^1, A_3^3 - A_2^2, A_4^4 - 1\}, \{A_1^1\})\}. \end{aligned}$$

The elements of S_0 only give rise to non-invertible matrices, and thus do not describe automorphisms, whereas S_1 leads to the matrix

$$\begin{pmatrix} a_1^2 & 0 & 0 & a_3 \\ 0 & a_1 & a_2 & 0 \\ 0 & 0 & a_1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad (a_1, a_2, a_3 \in K,\ a_1 \neq 0).$$

Using Γ' to determine the complete solution yields

$$\begin{pmatrix} a_1^2 & a_1^2 x_2 & -a_1^2 x_1 & -a_1^2 x_1 x_2 - \frac{1}{2} a_1^2 x_2^2 + a_3 \\ 0 & a_1 & a_2 & -a_1 x_1 - a_1 x_2 - a_2 x_2 \\ 0 & 0 & a_1 & -a_1 x_2 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

depending on $a_1, a_2, a_3, x_1, x_2 \in K$ with $a_1 \neq 0$.

Although we obtained a different representation here than we did with `groebfactor`, both representations can be transformed into each other.

Comparing both representations, we observe that with the first method four parameters, namely a_1, a_2, a_4, a_5, have to be extracted from the Gröbner basis. With the second method, two parameters, namely x_1 and x_2, arise quite naturally from the transformation Γ'. The parameter a_3 comes from the fact that A_4^1 does not occur in F'.

6. Efficiency

In this section we present some other examples, but we mainly concentrate on the running times for different inputs. We continue to examine the class of low-dimensional Lie algebras and adopt the notation in Patera *et al.* (1976). However, we consider the complexification of these Lie algebras.

In order to apply the algorithm as in the last example, we have to state which coordinates we regard to be the first ones. The matrices Γ and Γ' are for each Lie algebra constructed in the same way as before. Hence, it is sufficient to indicate the row and column indices of the entries of Γ for the first l coordinates:

$A_{4,7}$: (1,2), (1,3)	$A'_{4,7}$: (2,4), (3,4)
$A_{4,8}$: (1,2), (1,3)	$A_{4,10}$: (1,2), (1,3)
$A_{4,12}$: (1,3), (1,4)	$A_{5,2}$: (2,5), (3,4), (3,5)
$A'_{5,2}$: (1,2), (2,5), (3,5)	$A_{5,3}$: (3,4), (3,5)
$A_{5,5}$: (2,3), (2,5)	$A'_{5,5}$: (1,2), (2,5)
$A_{5,6}$: (1,2), (2,5)	$A_{5,22}$: (1,2), (2,5)
$A_{5,37}$: (1,2), (1,3), (1,4)	$A_{5,40}$: (5,4), (4,2), (4,3), (4,5)
$A_{6,1}$: (3,1), (3,2)	$A_{6,2}$: (3,1), (3,2), (4,1), (5,1)
$A_{6,22}$: (3,1), (3,2), (4,2), (5,2)	

Some of the Lie algebras are written with a prime to express that we use the same Lie algebra, but different coordinates.

The running times[†] for computing the sets S_0 and S_1 by the algorithm **gamma** (implemented in REDUCE), and the time for solving the same problem by means of `groebfactor` are put together in the following table.

[†]in milliseconds as reported by REDUCE

$A_{4,7}$	40	360	330	730	450
$A'_{4,7}$	70	340	150	560	440
$A_{4,8}$	30	380	210	620	260
$A_{4,10}$	40	310	630	980	1080
$A_{4,12}$	70	6060	4810	10940	11460
$A_{5,2}$	50	600	250	900	2620
$A'_{5,2}$	50	630	320	1000	2650
$A_{5,3}$	40	90	530	660	4100
$A_{5,5}$	50	1170	460	1680	2110
$A'_{5,5}$	60	1590	340	1990	2100
$A_{5,6}$	60	1600	330	1990	2110
$A_{5,22}$	60	1570	330	1960	2100
$A_{5,37}$	160	166570	2590	169320	175730
$A_{5,40}$	860	199610	7150	207620	224850
$A_{6,1}$	80	22290	700	23070	21830
$A_{6,2}$	140	4340	390	4870	2730
$A_{6,22}$	180	10600	850	11630	12760

The second to fifth columns refer to the algorithm **gamma**, and the sixth column contains the running time of `groebfactor`.

The second column denotes the time for the decision whether or not $\gamma_1, \ldots, \gamma_l$ could be used for a problem reduction. The third column shows the computation time for the set S_0, and the fourth for S_1. In the fifth column the total running time of **gamma** is shown, i.e. the sum of the columns 2, 3, and 4.

Comparing the last two columns, we observe that in many cases the algorithm **gamma** is faster. However, even when the computation time is not noticeably better, **gamma** has a certain advantage. Namely, in case we look for a presentation of the variety by parameters, the applicability of the transformation Γ already yields a part of the usable parameters in a natural way.

7. Conclusion

The algorithm **gamma** can readily be extended to use an additional set of polynomials to be used to describe constraints. In fact, an examination of the chosen examples has also been done with such an algorithm. This results (not surprisingly) in a faster algorithm. However, no advantage to `groebfactor` can be seen if this procedure is invoked with this additional parameter.

Therefore, and because of the fact that this paper is dedicated to the principal applicability of symmetries depending on free parameters in solving systems of polynomial equations, we presented **gamma** only in its simple form.

In some cases transformations arise which do not depend on parameters in a polynomial way. Despite this, for some of these transformations we can arrive at a splitting of the problem. The idea is to investigate "parameters" which are constrained by polynomial conditions. In this way, we try to increase the number of usable parameters. For example, the transformation is given by polynomials in $\sin x$ and $\cos x$, which is no polynomial situation w.r.t. x. The parameter x could not be used. After replacing $\sin x$ by s, $\cos x$ by c, where s and c are considered parameters, and adding the condition $c^2 + s^2 = 1$, everything is polynomial again. However, the polynomial $c^2 + s^2 - 1$ has to be taken into

account when the Gröbner basis of Δ is computed in **gamma**, since s and c are not *free* parameters.

A combinatorial difficulty is choosing which coordinates should vanish after the transformation has been applied. This difficulty has not been taken into consideration in this article. In our examples the coordinates could be determined without much effort by hand. For larger problems, however, there is need for an efficient algorithm which finds such coordinates automatically. Such an algorithm could, for example, factor the polynomials of B to divide the problem into smaller ones, and then look for relevant coordinates in the sub-problems.

Looking more closely at the examples shows also another method. Namely, it turns out that most of **gamma**'s time is spent computing the set S_0. It therefore seems reasonable to look for a possibility to apply the transformation also to this part of $\mathcal{Z}(B)$ which, in our consideration, appears to be an exceptional case.

References

Becker, T., Weispfenning, V. (1993). *Gröbner Bases. A Computational Approach to Commutative Algebra*, volume 141 of Graduate Texts in Mathematics. New York, Springer.

Buchberger, B. (1985). Gröbner bases: An algorithmic method in polynomial ideal theory. In Bose, N. K. ed., *Recent Trends in Multidimensional Systems Theory*, chapter 6, pp. 184–232. Dordrecht, Reidel.

Gräbe, H.-G. (June 1995a). *CALI—A REDUCE Package for Commutative Algebra. Version 2.2.1, June 1995*, Available via WWW from `http://www.informatik.uni-leipzig.de/~compalg/software/cali`.

Gräbe, H.-G. (1995b). On factorized Gröbner bases. In Fleischer, J., Grabmeier, J., Hehl, F. W., Küchlin, W. eds, *Computer Algebra in Science and Engineering: 28–31 August 1994. Bielefeld, Germany*, pp. 77–89. Singapore, World Scientific Publishing.

Hearn, A. C., Melenk, H. (1991). *REDUCE User's Manual, Version 3.4*. Santa Monica, The RAND Corporation.

Hemmecke, R. (1996). Lösen von Gleichungssystemen mit kontinuierlichen Symmetrien. Diplomarbeit, Universität Leipzig, Augustusplatz 10–11, 04109 Leipzig, Germany, February 1996. (in German).

Hietarinta, J. (1992). Solving the constant quantum Yang-Baxter equation in 2 dimensions with massive use of factorizing Gröbner basis computations. In Wang, P. S. ed., *ISSAC'92: Proceedings of the 1992 International Symposium on Symbolic and Algebraic Computation, July, Berkeley, California*, pp. 350–357. New York, NY 10036, USA, ACM Press.

Humphreys, J. E. (1980). *Introduction to Lie Algebras and Representation Theory,* Number 9 in Graduate Texts in Mathematics, 3rd edn, New York, Springer.

Patera, J., Sharp, R. T., Winternitz, P., Zassenhaus, H. (June 1976). Invariants of real low dimension Lie algebras. *J. Math. Phys.*, **17**, 986–994.

Received 5 September 1996
Accepted 13 June 2001