An. Şt. Univ. Ovidius Constanţa Vol. 15(1), 2007, 129–138

ON MODULAR COMPUTATION OF STANDARD BASIS

Gerhard Pfister

Abstract

In this article I want to report about modular methods to compute standard bases and the implementation in SINGULAR.

1 Introduction

It is well known that the computation of standard bases in a polynomial ring over the rational number $\mathbb{Q}$ is much more difficult than in a polynomial ring over a finite field $\mathbb{F}_p = \mathbb{Z}/p$. The reason is the enormous growth of the coefficients during the computation even if the result may have relatively small coefficients. To avoid these problems one can try to compute the standard bases over $\mathbb{F}_p$ for one suitable prime p (resp. several suitable primes) and use Hensel lifting as proposed in [10] (resp. Chinese remainder theorem) as proposed in [9] to lift the coefficients to $\mathbb{Z}$. Then Farey fractions (cf. [6], [7]) can be used to obtain the "correct" coefficients over $\mathbb{Q}$. This approach has been discussed since a long time (cf. [1],[2],[5],[8],[9],[10]). Here we follow the approach using Chinese remainder theorem. After the lifting there are two problems to be solved. It has to be checked whether the lifting of the standard bases to characteristic zero remains a standard basis and that it generates the ideal we started with. For the case of Gröbner bases (the monomial ordering is a global, i.e. a well–ordering) and homogeneous ideals a reasonable solution can be found for instance in the paper of Arnold (cf. [1]). It turns out that this method can also be used for standard basis with respect to local orderings.

Key Words: Standard bases, modular methods
2000 Mathematical Subject Classification: 13P10
Received: January, 2007

The case of mixed orderings or global orderings and non–homogeneous ideals is more complicated. With the same methods as in the homogeneous resp. local case one just obtains a standard basis generating an ideal containing the ideal we started with. For experiments this is already interesting, for proofs this is not enough.

2 Theoretical background

Let $I \subseteq \mathbb{Q}[x]$ be an ideal, $x = (x_1, \ldots, x_n)$, and $>$ be a monomial ordering. Let $I_0 = I \cap \mathbb{Z}[x]$ and $I_p = I_0\mathbb{Z}/p[x]$ for a prime p. Let $L(I)$ (resp. $L(I_p)$) be the minimal generating set of leading monomials of I (resp. I_p). The prime p is called lucky for I with respect to $>$ if $L(I) = L(I_p)$.

Remark 2.1. Let $g_1, \ldots, g_s \in \mathbb{Q}[x]$ be a standard basis* of $I\mathbb{Q}[x]_>$ and assume that the g_i are monic and let $u_{ij} \cdot \text{spoly}(g_i, g_j) = \sum h_{ijk}g_k$ be a standard representation for suitable $u_{ij}, h_{ijk} \in \mathbb{Q}[x], u_{ij}$ unit in $\mathbb{Q}[x]_>$. Then all primes not dividing a denominator of the coefficients of the g_i, u_{ij} and h_{ijk} are lucky. Especially randomly chosen primes are lucky.

The following proposition is the basis to find lucky primes without knowing a standard basis of $I\mathbb{Q}[x]_>$.

Proposition 2.2. *Let either I be homogeneous or $>$ be a local ordering. Let H_I (reps. H_{I_p}) be the Hilbert function (in case of I being homogeneous) or the Hilbert–Samuel function (in case of a local ordering) of $\mathbb{Q}[x]_>/I\mathbb{Q}[x]_>$ (resp. $\mathbb{F}_p[x]_>/I_p\mathbb{F}_p[x]_>$). Then $H_I(n) \leq H_{I_p}(n)$ for all n. If $H_I = H_{I_p}$ and $L(I) = \{f_1, \ldots, f_s\}, L(I_p) = \{m_1, \ldots, m_k\}$ such that $f_i < f_{i+1}$ and $m_i < m_{i+1}$ for all i and $f_1 = m_1, \ldots, f_{l-1} = m_{l-1}$ for $1 \leq l \leq \min\{s, k\}$ then $m_l \leq f_l$.*

Proof. The proof is not difficult and can be found for the global case in [1]. □

Corollary 2.3. *With the assumptions of proposition 2.2 let J be an ideal with the following properties:*

(1) $I \subset J$

(2) in case of a non–local ordering J is homogeneous

(3) $H_{I_p} = H_J$ for some prime p

Then $I = J$.

*For definitions and properties cf. [3], *Proof.* $H_{I_p}(n) = H_J(n) \leq H_I(n) \leq H_{I_p}(n)$. □

Corollary 2.4. *Let p, q be two primes. Assume one of the two following assumptions.*

(1) $H_{I_p}(n) = H_{I_q}(n)$ for $n < n_o$ and $H_{I_p}(n_0) < H_{I_q}(n_0)$.

(2) $H_I = H_{I_p}$ and $L(I_p) = \{f_1, \ldots, f_s\}, L(I_q) = \{m_1, \ldots, m_k\}$ such that $f_i < f_{i+1}$ and $m_i < m_{i+1}$ for all i and $f_1 = m_1, \ldots, f_{l-1} = m_{l-1}, m_l < f_l$ for $1 \leq l \leq \min\{s, k\}$.

Then q is not lucky.

Proof. q being lucky would imply $L(I) = L(I_q)$. This implies $H_I = H_{I_q}$. This is not possible because of proposition 2.2. □

The following procedure finds in a given list unlucky primes.

```
proc deleteUnluckyPrimes(list T,list L)
{
  int j,k;
  intvec hl,hc;
  ideal cT,lT;

  lT=lead(T[size(T)]);
  attrib(lT,"isSB",1);
  hl=hilb(lT,1);
  for (j=1;j<size(T);j++)
  {
     cT=lead(T[j]);
     attrib(cT,"isSB",1);
     hc=hilb(cT,1);
     if(hl==hc)
     {
        for(k=1;k<=size(lT);k++)
        {
           if(lT[k]<cT[k]){lT=cT;break;}
           if(lT[k]>cT[k]){break;}
        }
     }
     else
     {
```

```
        if(hc<hl){lT=cT;hl=hilb(lT,1);}
     }
  }
  j=1;
  attrib(lT,"isSB",1);
  while(j<=size(T))
  {
     cT=lead(T[j]);
     attrib(cT,"isSB",1);
     if((size(reduce(cT,lT))!=0)||(size(reduce(lT,cT))!=0))
     {
        T=delete(T,j);
        L=delete(L,j);
        j--;
     }
     j++;
  }
  return(list(T,L,lT));
}
```

Remark 2.5. Let HP_{I_p} resp. HP_{I_q} be the Hilbert–Poincaré series corresponding to H_{I_p} resp. H_{I_q}. Then $HP_{I_p}(t) = \frac{Q_{I_p}(t)}{(1-t)^n}$ and $HP_{I_q}\frac{Q_{I_q}(t)}{(1-t)^n}$ for suitable polynomials† $Q_{I_p}, Q_{I_q} \in \mathbb{Z}[t]$. Let $Q_{I_p} = \sum_{i=0}^{s_p} v_i t^i$ and $Q_{I_q} = \sum_{i=0}^{s_p} w_i t^i$. Then $v_n = w_n$ for $n < n_o$ and $v_{n_0} < q_{n_0}$ hold iff $H_{I_p}(n) = H_{I_q}(n)$ for $n < n_0$ and $H_{I_p}(n_0) < H_{I_q}(n_0)$. This implies that unlucky primes can be detected comparing the vectors of coefficients of the Hilbert series lexicographically.

Now we may assume that $p_1, \ldots, p_r$ are different lucky primes for I with respect to $>$. Let $G_{p_1}, \ldots, G_{p_r}$ be standard basis of $I_{p_1}, \ldots, I_{p_r}$ with the following properties:

(1) G_{p_i} is minimal for all i.

(2) the elements of G_{p_i} are monic for all i.

(3) G_{p_i} is uniquely determined by a fixed algorithm to compute it‡.

†They are also called the first Hilbert series and can be computed in Singular using the comment hilb$(I, 1)$.

‡In case of a global ordering one may choose a reduced Gröbner basis. In the local case reduced standard bases exist only for zero–dimensional ideals. Therefore we need to fix an algorithm to obtain uniqueness.

(4) Let $G_{p_i} = \{f_1^{(p_i)}, \ldots, f_l^{(p_i)}\}$ then lead $(f_k^{(p_i)}) < \mathrm{lead}(f_{k+1}^{(p_i)})$.

Using Chinese remainder theorem and Farey[§] fractions we obtain $G = \{f_1, \ldots, f_l\}$ with the following properties:

(1) $f_i \in \mathbb{Q}[x]$ and monic.

(2) There is an integer d such that $df_i \in \mathbb{Z}[x]$ for all i and $p_k \nmid d$ for all k.

(3) $df_i \mod p_k\mathbb{Z}[x] = (d \mod p_k) \cdot f_i^{(p_k)}$

Proposition 2.6. *For a random choice of $p_1, \ldots, p_r$ with r big enough G is a standard basis of $I\mathbb{Q}[x]_>$.*

Proof. Let $G = \{f_1, \ldots, f_l\} \subseteq \mathbb{Q}[x]$ be a standard basis of $I\mathbb{Q}[x]_>$ having the properties (1)...(4) similar to the G_{p_i} and let $u_{ij} \cdot \mathrm{spoly}(f_i, f_j) = \sum h_{ijk} f_k$ be standard representations as in Remark 2.1. We may assume that there is $d \in \mathbb{Z}$ not divisible by $p_1, \ldots, p_r$ such that du_{ij}, dh_{ijk}, df_i are in $\mathbb{Z}[x]$. Choose a bound m for the absolute value of the nominators and denominators occurring in the coefficients of $f_1, \ldots, f_l$. Enlarging the set of primes we may assume that $2m^2 < p_1 \cdot \ldots \cdot p_r$. Then the coefficients of the f_i are m–th Farey fractions and they map injectively to $\mathbb{Z}/p_1 \cdot \ldots \cdot p_r = \mathbb{F}_{p_1} \times \ldots \times \mathbb{F}_{p_r}$.

The standard representations of the $\mathrm{spoly}(f_i, f_j)$ induce standard representations of the spoly's of $f_i \mod p_k$ and $f_j \mod p_k$ and therefore $\{f_i \mod p_k\}_{i=1,\ldots,l}$ is a standard basis of $I_{p_k}\mathbb{F}_{p_k}[x]_>$ having the properties (1)...(4). This set mus be G_{p_k}. □

Remark 2.7. There is no efficient bound of the nominators and denominators of a minimal standard basis known in terms of generators of an ideal. Therefore the number of primes needed has to be found by trial and error. We use the fact that Buchberger's algorithm applied to a system of polynomials which is already a standard basis is usually less expensive than applied to a system of generators which is not a standard basis.

In Singular[¶] the following algorithm is implemented.

`modstd` (S)

[§]The set $F_m := \{\frac{a}{b} \in \mathbb{Q} \mid \gcd(a,b) = \gcd(b,N) = 1\}$ and $f_N := \mathbb{Q}_N \to \mathbb{Z}/N$ the canonical map defined by $f_N(\frac{a}{b}) = (a \mod N)(b \mod N)^{-1}$. It is not difficult to see that if $2m^2 < N$ the restriction of f_N to $\mathbb{Q}_N \cap F_m$ is injective. For given $q \in f_N(\mathbb{Q}_N \cap F_m)$ a variant of the Euclidian algorithm computes the uniquely determined $\frac{a}{b} \in F_n$ such that $f_N(\frac{a}{b}) = q$.

[¶]The corresponding algorithms are implemented in the library `modstd.lib`.

Input $S \subseteq \mathbb{Z}[x]$ a finite set of polynomials

Output $G \subseteq \mathbb{Q}[x]$ a minimal standard basis of $\langle S\rangle\mathbb{Q}[x]_>, >$ a fixed ordering

(1) $L = \emptyset$, $T = \emptyset$ (list of primes, list of standard bases)
$M = \emptyset$, $K = \emptyset$ (result, test set)

(2) while $\#L < 5$ do

- insert randomly chosen primes to L such that $\#L = 5$
- For $p \in L$ compute a standard basis M_p of $S\mathbb{F}_p[x]_>$ satisfying the properties described before and insert it to T.
- delete unlucky primes in L and the corresponding standard bases in T.

(3) Use Chinese remainder theorem and Farey fractions to lift the standard bases M_p for p in L to a system M of polynomials of $\mathbb{Q}[x]$.

(4) while $M \neq K$

- choose randomly a prime p with $p \notin L$
- insert p to L and compute the corresponding standard basis M_p of $S\mathbb{F}_p[x]_>$ and insert it to T
- delete unlucky primes in L and the corresponding standard bases in T.
- if more than one prime was deleted go to (2)
- if no prime was deleted $M = K$ go to (3).

(5) Use Buchberger's algorithm to compute a standard basis of M satisfying the properties described before. If it is different from M go to (1).

(6) Reduce the input S with respect to M. If the result is different from 0 then $K := \emptyset$ go to (4)

(7) return (M).

Remark 2.8. If the ordering $>$ is not local or if the ideal generated by the polynomials in S is not homogeneous then the test in (6) of the algorithm just implies $\langle S\rangle\mathbb{Q}[x]_> \subseteq \langle M\rangle\mathbb{Q}[x]_>$ because Corollary 2.3 does not hold in general. To obtain equality one could lift together with the standard bases M_p of T

also the relations[‖] expressing $\langle S\rangle\mathbb{F}_p[x]_> = \langle M_p\rangle\mathbb{F}_p[x]_>$. Experiments showed that this is much more expensive.

3 Examples

We will consider here only examples for local orderings and zero–dimensional ideals. In this case usually the reduced standard basis is relatively simple compared to polynomials occurring during the computations. Therefore the modular method including the verification is very efficient.

The examples are obtained studying singularities during the computation of Milnor numbers and Tjurina numbers.

We consider the ring $Q[x, y, z]$ with the local ordering ds defined by the matrix $\begin{pmatrix} -1 & -1 & -1 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \end{pmatrix}$.

Example 1

$f = x^6 + y^8 + z^{10} + x^5 + x^3y^2 + x^2yz^2 + xy(y^2 + x)^2$

$I = \langle \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z} \rangle$

Example 2

$f = xyz(x + y + z)^2 + (x + y + z)^3 + x^{10} + y^{10} + z^{10}$

$I = \langle \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z} \rangle$

Example 3

$f = x^{25} + y^{25} + z^{15} + x^7y^4 + x^4y^4z^3 + x^3y^5(y^2 + x)^2$

$I = \langle f, \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z} \rangle$

Example 4

$f = x^{16} + y^{15} + z^{12} + x^6y^3 + x^3y^3z^3 + x^2y^4(y^2 + x)^2$

$I = \langle \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z} \rangle$

[‖] In Singular the command `liftstd` (I, M) computes a standard basis $\{g_1, \ldots, g_s\}$ of the ideal $I = \langle f_1, \ldots, f_m\rangle$ together with a matrix M such that $\begin{pmatrix} g_1 \\ \vdots \\ g_s \end{pmatrix} = M \begin{pmatrix} f_1 \\ \vdots \\ f_m \end{pmatrix}$.

Timings (in seconds):

	modStd	std	memory used for std
1	3	258	1.3 GB
2	41	-	14 GB
3	5	11211	2.1 GB
4	10	35	5.6 MB

The examples are computed with Singular 3-0-3 on a Linux PC with AMD Athlon (tm) 64 Processor 2800+ with 1.8 GHz. `modStd` is the modular standard basis computation including verification, `std` is the usual standard basis computation implemented in Singular. In the second example the computation was stopped after 3 hours.

References

[1] Arnold, E.A., *Modular algorithms for computing Gröbner bases*, J. of Symbolic Computations, **35**(2003), 403-419.

[2] Ebert, G.L., *Some comments on the modular approach to Gröbner bases*, ACM SIGSAM Bulletin, **17**(1983), 28-32.

[3] Greuel, G.-M.; Pfister, G., *A singular Introduction to commutative Algebra*, Springer, 2002.

[4] Greuel, G.-M.; Pfister, G.; Schönemann, H., Singular - A Computer Algebra System for Polynomial Computations. Free software under the GNU General Public Licence (1990 – to date).

[5] Gräbe, H., *On lucky primes*, Journal of Symbolic Computation, **15**(1994), 199-209.

[6] Hardy, G.H.; Wright, E.M., *An Introduction to the Theory of Numbers*, Oxford University Press, 1954.

[7] Kornerup, P.; Gregory, R., *Mapping integers and Hensel codes onto Farey fractions*, Bit 23 (1983), 9-20.

[8] Pauer, F., *On lucky ideals for Gröbner bases computations*, Journal of Symbolic Computation, **14**(1992), 471–482.

[9] Sasaki, T.; Takeshima, T., *A modular method for Gröbner–basis construction over* $\mathbb{Q}$ *and solving system of algebraic equations*, Journal of Information Processing, **12**(1989), 371–379.

[10] Winkler, F., *A p–adic approach to the computation of Gröbner bases*, Journal of Symbolic Computation, **6**(1987), 287–304.

Department of Mathematics
University of Kaiserslautern
P.O. Box 3049 67653 Kaiserslautern
Germany
E-mail: pfister@mathematik.uni-kl.de