Compact Representation of Polynomials for Algorithms for Computing Gröbner and Involutive Bases

D. A. Yanovich
Joint Institute for Nuclear Research, ul. Zhukov Kyuri 6, Dubna, Moscow oblast, 141980 Russia
E-mail: yan@jinr.ru
Received October 13, 2014

Abstract—In the computation of involutive and Gröbner bases with rational coefficients, the major part of the memory is occupied by precision numbers; however, in the case of modular operations (especially, in the computation of Gröbner bases), of most importance is the problem of compact representation of monomials composing polynomials of the system. For this purpose, for example, ZDD diagrams and other structures are used, which make execution of typical operations—multiplication by a monomial and reduction of polynomials—more complicated. In this paper, an attempt is made to develop convenient (in the sense of computation of bases) and compact representation of polynomials that is based on hash-tables. Results of test runs are presented.

DOI: 10.1134/S0361768815020097

1. INTRODUCTION

Involutive and Gröbner Bases

We use the following notation:

$X$ is a set of variables, $X = \{x_1, \ldots, x_n\}$;
$\mathbb{R} = \mathbb{K}[X]$ is a ring of polynomials over a field $\mathbb{K}$ of zero characteristic;
$f, g, h, q, r$ are polynomials in $\mathbb{R}$;
$a, b, c$ are elements of $\mathbb{K}$;
$F, G,$ and $H$ are finite subsets in $\mathbb{R}$,
$\langle F \rangle$ is an ideal in $\mathbb{R}$ generated by $F$;
$\mathbb{Z}_{\geq 0}$ is the set of nonnegative numbers;
$\mathbb{M} = \{x_1^{d_1} \ldots x_n^{d_n} | d_i \in \mathbb{Z}_{\geq 0}\}$ is the set of monomials in $\mathbb{R}$;
$u, v, w, s,$ and $t$ is the set of monomials or terms;
$U, V,$ and $W$ are finite subsets in $\mathbb{M}$;
$\deg(u)$ is the degree of variable $x_i$ in $u$;
$\deg(u)$ is the total degree of monomial $u$;
$>\text{ is an admissible ordering with the order of variables } x_1 > \ldots > x_n$;
$\text{lt}(f)$ is the leading term with respect to the ordering $>\text{;
}\text{lm}(f)$ is the leading monomial in $f$;
$\text{lm}(F)$ is the set of leading monomials in $F$;
$\text{lcm}(F)$ is the least common multiple of the set of monomials in $\text{lm}(F)$;
$u|v$ means that monomial $u$ divides monomial $v$.

The definition and algorithm for constructing Gröbner bases were proposed by B. Buchberger [1]. We briefly recall the basic concepts.

Definition 1 [1, 2]. Let an admissible monomial ordering be given. A finite subset $G = \{g_1, \ldots, g_m\}$ of elements of an ideal $I$ is called its Gröbner basis if

$$\langle \text{lm}(g_1), \ldots, \text{lm}(g_m) \rangle = \langle \text{lm}(I) \rangle.$$

The central concept in the algorithm of the computation of a Gröbner basis is the $S$-polynomial.

Definition 2 [1, 2]. An $S$-polynomial of $f, g \in \mathbb{R}$ is the combination

$$\text{Spoly}(f, g) = \frac{\text{lcm}(\text{lm}(f), \text{lm}(g))}{\text{lt}(f)} f - \frac{\text{lcm}(\text{lm}(f), \text{lm}(g))}{\text{lt}(f)} g.$$

Table

<table>
<thead>
<tr>
<th>Example</th>
<th>$t_{\text{hash}}/t_{\text{nohash}}$</th>
<th>$m_{\text{hash}}/m_{\text{nohash}}$</th>
<th>$M_{\text{saved}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>cyclic8</td>
<td>0.83</td>
<td>0.61</td>
<td>93.98</td>
</tr>
<tr>
<td>dl</td>
<td>1.02</td>
<td>0.61</td>
<td>40.50</td>
</tr>
<tr>
<td>eco11</td>
<td>0.96</td>
<td>0.60</td>
<td>106.75</td>
</tr>
<tr>
<td>eco12</td>
<td>1.13</td>
<td>0.60</td>
<td>700.36</td>
</tr>
<tr>
<td>f855</td>
<td>1.41</td>
<td>0.62</td>
<td>15.82</td>
</tr>
<tr>
<td>h cyclic8</td>
<td>1.07</td>
<td>0.62</td>
<td>61.33</td>
</tr>
<tr>
<td>katsura10</td>
<td>1.09</td>
<td>0.60</td>
<td>65.88</td>
</tr>
<tr>
<td>noon8</td>
<td>1.72</td>
<td>0.63</td>
<td>32.23</td>
</tr>
<tr>
<td>noon9</td>
<td>1.77</td>
<td>0.61</td>
<td>214.16</td>
</tr>
<tr>
<td>red cyc7</td>
<td>1.02</td>
<td>0.63</td>
<td>6.14</td>
</tr>
<tr>
<td>redeco11</td>
<td>0.82</td>
<td>0.61</td>
<td>16.98</td>
</tr>
<tr>
<td>redeco12</td>
<td>1.09</td>
<td>0.61</td>
<td>73.15</td>
</tr>
<tr>
<td>reimer7</td>
<td>0.74</td>
<td>0.61</td>
<td>50.22</td>
</tr>
</tbody>
</table>
Theorem 1 [1, 2]. Let \( I \) be some polynomial ideal. The basis \( G = \{g_1, \ldots, g_m\} \) of the ideal \( I \) is a Gröbner basis if and only if, for all pairs \( i \neq j \), NormalForm(Spoly(\( g_i, G \)) = 0.

If, in some self-consistent way, we forbid the division by certain (called non-multiplicative) variables and permit it by the other (called multiplicative) variables, we obtain a contraction of the division operation—an involutive division [3, 4]. Let us give basic definitions.

Definition 3 [3, 4]. We will say that a set of monomials \( \mathbb{M} \) has an involutive division \( L \) if, for any finite non-empty subset \( U \subset \mathbb{M} \) and any \( u \in U \), an \( M_L(u, U) \subseteq X \) is given that generates a submonoid \( L(u, U) \subseteq M \) consisting of monomials with variables from \( M_L(u, U) \) and satisfying the following conditions:

(a) from \( u, v \in U \) and \( uL(u, U) \cap vL(v, U) \neq \emptyset \), it follows that \( u \in vL(v, U) \) or \( v \in uL(u, U) \);

(b) from \( v \in U \) and \( uL(u, U) \), it follows that \( L(v, U) \subseteq L(u, U) \);

(c) from \( u \in V \) and \( V \subseteq U \), it follows that \( L(u, U) \subseteq L(u, V) \) for all \( u \in V \).

The elements of \( M_L(u, U) \) are called \( L \)-multiplicative for \( u \), and elements of \( NL_L(u, U) = X \setminus M_L(u, U) \) are called \( L \)-nonmultiplicative. If \( w \in uL(u, U) \), then \( u \) is called an \( (L-) \)involution divisor of \( w \) and denoted as \( u \mid L w \). In turn, the monomial \( w \) is called \( (L-) \)multiplication of \( u \).

Definition 4 [3, 4]. The set of monomials is called \( L \)-involutional if

\[
(\forall u \in U) \quad (\forall w \in \mathbb{M}) \quad (\exists v \in U) \quad [v \parallel u w].
\]

Using a constructive and Noetherian division (for example, the Janet division), we can build an involutive basis algorithmically, by supplementing a partially involutive set.

Definition 5 [3]. An autoreduced set \( F \) is called partially involutive up to a monomial \( v \) in an admissible monomial ordering \( \succ \) if the following condition is satisfied:

\[
(\forall f \in F) (\forall u \in \mathbb{M}) (\text{NormalForm}_L(fu, F) = 0),
\]

where NormalForm\(_L\) denotes an involutive normal form defined by the replacement of the ordinary division by the involutive division.

Thus, having added nonzero involutive normal forms of nonmultiplicative extensions of the set \( F \) to a partial basis, we obtain the desired involutive basis.

Below is a simplified algorithm for computing involutive bases that does not use criteria for avoiding reductions. The full version of the algorithm is presented in [3].

Input: \( F, L, \prec \)

Output: \( T \) — involutive basis of \( F \)

1: \( T := \emptyset \quad Q := F \)
2: while \( Q \neq \emptyset \) do
3: \( P := \{ p \in Q | \text{lm(pol}(p)) \text{== min}(\text{lm}(Q)), \}
4: \text{NF}_L(\text{pol}(p), T) \neq 0 \}
5: \( T := T \cup P, Q := Q \setminus P \)
6: for all \( p \in P \) do
7: \( Q := Q \cup \{ p \cdot \text{NM}_L(p, T) \} \)
8: if \( \text{lm}(\text{pol}(p)) \text{== anc}(p) \) then
9: for all \( r \in T \) do
10: if \( \text{lm}(\text{pol}(r)) \equiv \text{lm}(\text{pol}(p)) \) then
11: \( Q := Q \setminus \{ r \} \)
12: \( T := T \setminus \{ r \} \)
13: fi
14: od
15: fi
16: od
17: od

The relation between involutive and Gröbner bases is established by the following theorem.

Theorem 2 [3]. An involutive basis that is autoreduced in the sense of ordinary division is a reduced Gröbner basis.

2. MOTIVATION

When developing a modular parallel implementation of the algorithm for computing involutive bases [6], many examples were found where calculations failed because of lack of computer memory. Since the coefficients do not play a significant role in modular calculations, the cause of the extremely fast increase of the required memory is that much of it is spent on the internal representation of monomials and polynomials.

To verify this hypothesis, we carried out experiments with examples from cryptography. It turned out, for example, that the consumption of memory when running test HFE25–96 [7] reached 16Gb after several seconds of computation. Having studied the set of monomials in the polynomials arising in the course of computation, we made a decision to try to use more complicated data structures for representation of polynomials, thus sacrificing speed to please memory saving.

3. POLYNOMIAL REPRESENTATION

3.1. Survey of Modern Ways of Polynomial Representations

Currently, there are several ways to represent polynomials: