A Modular Method for Gröbner-basis Construction over Q and Solving System of Algebraic Equations

TATEAKI SASAKI* and TAKU TAKESHIMA**

A modular method for constructing Gröbner-basis of polynomial ideal over Q is described. Given a finite set of polynomials in $Z[x_1, \ldots, x_n]$, the method calculates Gröbner-bases over $Z/(p_i)$, $i=1, \ldots, k$, where $p_1, \ldots, p_k$ are distinct primes, then it constructs a Gröbner-basis over Q by using Chinese remainder algorithm and conversion of integers to rationals. By this method, we can calculate Gröbner-basis with large-sized coefficients efficiently by avoiding intermediate coefficient growth. We propose two algorithms, one is simple but probabilistic in that it may give a wrong basis such that ideal(*wrong basis*)$\supset$ideal(*true basis*) with an extremely small possibility, and the other is less simple but gives the correct basis. we also discuss solving system of algebraic equations by using the modular Gröbner-basis method.

1. Introduction

Gröbner-basis [2, 3] of polynomial ideal is currently one of the most important notions in algebraic computation, and we already know that many algebraic operations can be reduced to calculating Gröbner-bases. On the other hand, it is found that calculation of Gröbner-basis of moderate size or larger often causes strong coefficient growth [1, 5]. The coefficient growth is so common in the Gröbner-basis computation that it is the most serious problem in actual computation. Observing the coefficient growth carefully, we readily find that much larger coefficients are handled in the intermediate step of computation than in the final expressions (so-called intermediate coefficient growth), if we adopt the conventional computation method. We can avoid the intermediate coefficient growth by the use of a modular method.

So far, several authors have investigated the use of modular methods for Gröbner-basis construction. To our knowledge, the first literature on such study is Ebert [4], which discussed the "lucky" primes for modular construction of Gröbner-basis using Chinese remainder theorem. Ebert concluded that algorithmic determination of lucky prime was very difficult and he proposed no algorithm. Subsequently, Trinks [7, 8] proposed a p-adic method for solving a system of algebraic equations of special (simple) type, and Winkler [11] discussed the p-adic method for constructing Gröbner-bases in the general case. However, the problem of lucky primes was not solved and the complete modular algorithm for Gröbner-basis construction has not been presented so far. In this paper, we propose a complete modular method based on Chinese remainder theorem.

In our method, we do not check the luckiness of primes but check the unluckiness and finally we check the correctness of the basis constructed. In addition to this theoretical aspect, our algorithm seems to be quite useful practically because of its high efficiency.

One may think that, compared with p-adic approach, our method based on Chinese remainder algorithm is much less efficient. This is, however, not true. In the case of p-adic method, considerably complicated computation is necessary at each lifting step $p^i \to p^{i+1}$ (see [11] for details), but no such computation is necessary in our case. (The p-adic method in its full form has not been implemented yet, so we cannot say definite thing about its efficiency.) Furthermore, our method is favorable in that the possibility of handling unlucky case is extremely small (see **5.** for details).

In **2**, we summarize necessary notions and theorems on Gröbner-basis. Calculating Gröbner-basis with special term ordering is directly related with solving system of algebraic equations, and we also explain Gröbner-basis method for solving system of algebraic equations. In **3**, we consider the use of Chinese remainder algorithm to Gröbner-basis construction. In particular, we discuss treatment of unlucky primes. In **4**, we present a probabilistic algorithm, prove its termination property, and discuss the probabilistic nature. A complete algorithm is presented in **5**, as well as improvements of algorithm. **6.** describes some empirical study.

*The Institute of Physical and Chemical Research, 2-1 Hirosawa, Wako-shi, Saitama 351-01, Japan
**Fujitsu Limited, 140 Miyamoto, Numazu-shi, Shizuoka 410-03, Japan

2. Gröbner-basis and System of Algebraic Equations

We first define several notations. In the following, we assume that the polynomials are in $K[x_1, \ldots, x_n]$, with K a number field.

Order $>$. Let $a=(a_n, \ldots, a_1)$ and $b=(b_n, \ldots, b_1)$ be n-tuples of nonnegative integers. We define $a>b$ if $a_k>b_k$ for some integer k, $1\leq k\leq n$, and $a_j=b_j$ for all j such that $n\geq j\geq k+1$.

Term-order $\triangleright$. Let $T_i=c_i x_n^{e_{in}}\cdots x_1^{e_{i1}}$ be a monomial in $K[x_1, \ldots, x_n]$, where $c_i\in K$. We can order the terms T_i, $i=1, 2, \ldots,$ uniquely by the order $>$ on n-tuples $(e_{in}, \ldots, e_{i1})$, which we call the lexicographic order, or on $(n+1)$-tuples $(\Sigma_j e_{ij}, e_{in}, \ldots, e_{i1})$, which we call the total-degree order. If T_i is of higher order than T_j, we write $T_i\triangleright T_j$.

Head term, head coefficient and head power product (abbreviated to ht, hc and hpp, respectively). Let $F\in K[x_1, \ldots, x_n]$. The highest order monomial, with respect to $\triangleright$, of F is called the *head term* of F and written as $\mathrm{ht}(F)$. Let $\mathrm{ht}(F)=cx_n^{e_n}\cdots x_1^{e_1}$, with $c\in K$. The c is called the *head coefficient* of F and written as $\mathrm{hc}(F)$, and $x_n^{e_n}\cdots x_1^{e_1}$ is called the *head power product* and is written as $\mathrm{hpp}(F)$.

S-polynomial (abbreviated to Spol). Given polynomials F_1 and F_2, the *S-polynomial* of F_1 and F_2 is defined by

$$\mathrm{Spol}(F_1, F_2)=\mathrm{hc}(F_2)\frac{lcm}{\mathrm{hpp}(F_1)}F_1-\mathrm{hc}(F_1)\frac{lcm}{\mathrm{hpp}(F_2)}F_2, \quad (1)$$

where $lcm=\mathrm{LCM}(\mathrm{hpp}(F_1), \mathrm{hpp}(F_2))$.

M-reduction. Let F and G be polynomials over K, $\mathrm{ht}(G)=bV$, with $b\in K$ and V a monomial with coefficient 1. If F contains a term aU such that $U=U'V$, where $a\in K$ and U' is a monomial, then $F'=F-(a/b)U'G$ is called an *M-reduct* of F with respect to G and written as $F\rightarrow_G F'$. Constructing F' is a term-order reduction procedure and called the *M-reduction*. By $F\rightarrow\!\!\rightarrow_G\tilde{F}$, we denote that terms of F are M-reduced by G successively so that no term of $\tilde{F}$ is *M-reducible* w.r.t. G. Let Γ be a finite set of polynomials over K. By $F\rightarrow\!\!\rightarrow_\Gamma\tilde{F}$ we denote that terms of F are M-reduced by elements of Γ successively so that no term of $\tilde{F}$ is M-reducible with respect to Γ.

Polynomial ideal. Let $F_1, \ldots, F_r$ be elements of $K[x_1, \ldots, x_n]$. The *ideal* (polynomial ideal) generated by $F_1, \ldots, F_r$ is a set of all the polynomials of the form

$$F=h_1F_1+\cdots+h_rF_r,\ h_i\in K[x_1, \ldots, x_n],\ i=1, \ldots, r.$$

The ideal is written as $(F_1, \ldots, F_r)$ and the set $\{F_1, \ldots, F_r\}$ is called a *basis* of the ideal.

Gröbner-basis. Let $\Gamma=\{G_1, \ldots, G_s\}$ be a set of polynomials over K and $I=(F_1, \ldots, F_r)$ be an ideal in $K[x_1, \ldots, x_n]$. The Γ is a *Gröbner-basis* of I if and only if $(F_1, \ldots, F_r)=(G_1, \ldots, G_s)$ and $F\rightarrow\!\!\rightarrow_\Gamma 0$ for any element F in I.

Reduced normalized Gröbner-basis. Let $\Gamma=\{G_1, \ldots, G_s\}$ be a Gröbner-basis, and put $\Gamma_{(i)}=\Gamma-\{G_i\}$. The Γ is called a *reduced Gröbner-basis* if and only if $G_i\rightarrow\!\!\rightarrow_{\Gamma_{(i)}}G_i$ for every $i=1, \ldots, s$. The Γ is called a normalized basis if $\mathrm{hc}(G_i)=1$, $i=1, \ldots, s$.

For the construction and properties of Gröbner-basis, see [2]. Here, we give an algorithmic and simple criterion of Gröbner-basis.

Theorem 1 (Buchberger [2]). *Let $\Gamma=\{G_1, \ldots, G_s\}$ be a set of polynomials over K. The Γ is a Gröbner-basis of the ideal $(G_1, \ldots, G_s)$ if and only if $\mathrm{Spol}(G_i, G_j)\rightarrow\!\!\rightarrow_\Gamma 0$ for all the pairs $\langle G_i, G_j\rangle$, $1\leq i\neq j\leq s$, in Γ.* □

In addition to Theorem 1, the following well-known theorems are crucially important in constructing modular algorithms for Gröbner-bases.

Theorem 2 (Buchberger [2]). *The reduced normalized Gröbner-basis $\{G_1, \ldots, G_s\}$ of polynomial ideal $(F_1, \ldots, F_r)$ is unique.* □

Theorem 3 (Winkler [12]). *Let Γ and Γ' be Gröbner-bases in $K[x_1, \ldots, x_n]$, $\mathrm{hpp}(\Gamma)=\mathrm{hpp}(\Gamma')$ and $\Gamma\subseteq\mathrm{ideal}(\Gamma')$, where $\mathrm{hpp}(\Gamma)=$set of $\mathrm{hpp}($element of $\Gamma)$. Then, $\mathrm{ideal}(\Gamma)=\mathrm{ideal}(\Gamma')$.* □

Next, we consider solving a system of algebraic equations

$$\{F_1(x_1, \ldots, x_n)=0, \ldots, F_r(x_1, \ldots, x_n)=0\}, \quad (2)$$

where $F_i\in \boldsymbol{Q}[x_1 \ldots, x_n]$, $i=1, \ldots, r$. We assume that the ideal $(F_1, \ldots, F_r)$ is zero-dimensional, *i.e.*, the system (2) has finitely many solutions in $\boldsymbol{C}$(field of complex numbers).

We define the term-order for $x_1, \ldots, x_n$ as

$$x_n\triangleright\cdots\triangleright x_2\triangleright x_1. \quad (3)$$

It is well-known that the reduced Gröbner-basis of ideal $(F_1, \ldots, F_r)$ with respect to the lexicographic order (3) becomes

$$\{G_1(x_1), G_{21}(x_1, x_2), \ldots, G_{31}(x_1, x_2, x_3), \ldots, G_{n1}(x_1, \ldots, x_{n-1}, x_n), \ldots\}, \quad (4)$$

where

$$G_{ki}\in \boldsymbol{Q}[x_1, \ldots, x_k],\ k=1, 2, \ldots, n,\ i=1, \ldots,$$

$$\mathrm{ht}(G_1)\lhd\mathrm{ht}(G_{21})\lhd\cdots\lhd\mathrm{ht}(G_{31})\lhd\cdots\lhd\mathrm{ht}(G_{n1}\lhd\cdots,$$

and

$$\mathrm{ht}(G_{k1})=c_k x_k^{m_k}, c_k\in \boldsymbol{Q},\ m_k\in N.$$

Furthermore, all the roots of (2) are given by the roots of

$$\{G_1(x_1)=0, G_{21}(x_1, x_2)=0, \ldots, G_{n1}(x_1, \ldots, x_{n-1}, x_n)=0, \ldots\}. \quad (5)$$

Note that, if (2) has no multiple roots then (4) often has the following form.

$$\{G_1(x_1), x_2-\bar{G}_2(x_1), \ldots, x_n-\bar{G}_n(x_1)\}. \quad (4')$$

Correspondingly, system (5) often has the following

form.

$$\{G_1(x_1)=0, x_2-\tilde{G}_2(x_1)=0, \ldots, x_n-\tilde{G}_n(x_1)=0\}. \quad (5')$$

The system (5) is easy to solve numerically, so the main task of solving (2) by algebraic method is the reduction of system (2) to (5). The reduction can be made, in principle, by calculating a reduced Gröbner-basis of $(F_1, \ldots, F_r)$ with respect to the lexicographic order, but it often requires tremendous amount of computation time as we have noted in **1**. Using the modular algorithm to be described in this paper, we can derive the system (5) (correctly speaking, system (5) or its subsystem) efficiently.

Our method for constructing Gröbner-basis is quite simple in principle. We first describe our method in an algorithm form and explain it in detail in **3**. The complete algorithm as well as termination proof are given in **4**.

3. Basic Considerations

As we have seen in **2**, our problem is to construct a Gröbner-basis $\Gamma=\{G_1, \ldots, G_s\}$, with respect to some term-order, of a given ideal $(F_1, \ldots, F_r)$ in $Q[x_1, \ldots, x_n]$. In this section, we consider calculating Γ by a modular method using Chinese remainder algorithm.

Throughout this paper, by $p_1, \ldots, p_k$, we denote distinct primes of word size and we assume that

$$\{F_1, \ldots, F_r\} \in Z[x_1, \ldots, x_n].$$

(If some coefficients of F_i are rationals, we multiply LCM of the denominators.) We first discuss luckiness of the prime p.

Definition [lucky prime]: A prime p is *lucky* if and only if

$$\Gamma' \equiv \Gamma (\mathrm{mod}\, p),$$

where Γ and Γ' are reduced and normalized Gröbner-basis over Q and $Z/(p)$, respectively. □

Let Γ be the reduced and normalized Gröbner-basis over Q, as above, and consider the mapping from Q onto $Z/(p)$. There exists an image of Γ, let it be Γ^*, if and only if every coefficient of Γ has its image in $Z/(p)$, or in other words, if and only if p divides no denominator of the coefficients of Γ. By this and the uniqueness of reduced and normalized Gröbner-basis, we see $\Gamma'=\Gamma^* \equiv \Gamma (\mathrm{mod}\, p)$ if and only if p divides no denominator of coefficients of Γ. Hence, we have the following proposition.

Proposition 1. The number of unlucky primes is finite. □

Unfortunately, the above definition of luckiness is useless for actual computation, because we do not know Γ before the computation. Therefore, we introduce the concept of "procedurally unlucky prime" which can be checked easily but may be dependent on the procedure of Gröbner-basis construction.

Definition [procedurally unlucky prime]. Assume that all the polynomials concerned are normalized. A prime p is unlucky if it divides some denominator of coefficients of input polynomials or some denominator of normalized S-polynomials constructed over Q. □

The initial polynomials are in $Z[x_1, \ldots, x_n]$. The definitions of M-reduction and S-polynomial tell us that the denominators of Γ are composed of only factors of head coefficients of F_i, $i=1, \ldots, r$, and S-polynomials constructed which are M-reduced but not normalized yet.

In order to define practically useful "unlucky primes", we assume that the Gröbner-basis construction over $Z/(p_i)$, $i=1, 2, \ldots$, is done by the same procedure as that for Gröbner-basis over Q. In particular, we assume the following three conditions on the construction.

(I) When a polynomial F is M-reduced by elements of a set of polynomials Γ', elements of Γ' are selected according to a definite rule which is independent of the coefficients of Γ'.

(II) When an S-polynomial is constructed by selecting two elements from Γ', the selection is done according to a definite rule which is independent of the coefficients of Γ'.

(III) Normalization of a polynomial is done according to a definite rule which is independent of the coefficients of Γ'.

Let $Sp^{(i,j)}$ be an S-polynomial constructed j-th over $Z/(p_i)$, and let $\mathrm{HIST}^{(i)}$ be a list of $\mathrm{ht}(Sp^{(i,j)})$, $j=1, 2, \ldots$, i.e.,

$$\mathrm{HIST}^{(i)}=(\mathrm{ht}(Sp^{(i,1)}), \mathrm{ht}(Sp^{(i,2)}), \ldots).$$

We call $\mathrm{HIST}^{(i)}$ the ht-history for p_i.

Definition [weak (procedurally) unlucky prime]. A prime $p_{i'}$ is weak unlucky compared with p_i if

$$\mathrm{HIST}^{(i)} \triangleright \mathrm{HIST}^{(i')}$$

where $\mathrm{HIST}^{(i)} \triangleright \mathrm{HIST}^{(i')}$ if and only if the following (i) or (ii) holds:

(i) $\mathrm{hpp}(Sp^{(i,j')})=\mathrm{hpp}(Sp^{(i',j')})$, $j'=1, \ldots, j-1$, and $\mathrm{hpp}(Sp^{(i,j)}) \triangleright \mathrm{hpp}(Sp^{(i',j)})$ for some integer $j>0$,

(ii) $\mathrm{hpp}(Sp^{(i,j')})=\mathrm{hpp}(Sp^{(i',j')})$, $j'=1, \ldots, j-1$, and $\mathrm{length}(\mathrm{HIST}^{(i)})>\mathrm{length}(\mathrm{HIST}^{(i')})=j-1$, for some integer $j>0$.

Modular construction of the Gröbner-basis Γ over Q is done in an orthodox way as follows (the full algorithm will be given later).

Modular Gröbner-basis method [in a gross form]:
Input: a set of polynomials $\{F_1, \ldots, F_r\} \in Z[x_1, \ldots, x_n]$;
a set of distinct primes $\{p_1, \ldots, p_k\}$;
Output: a reduced Gröbner-basis $\Gamma=\{G_1, \ldots, G_s\}$;

Step A [Gröbner-basis in $Z/(p_i)[x_1, \ldots, x_n]$, $i=1, \ldots, k$]:
For sufficiently many primes $p_1, \ldots, p_k$, calculate a reduced Gröbner-basis
$\Gamma^{(i)}=\{G_1^{(i)}, \ldots, G_s^{(i)}\}$ in $Z/(p_i)[x_1, \ldots, x_n]$, $i=1, \ldots, k$,
where the normalization is made as $\mathrm{hc}(G_j^{(i)})=1$, $j=1, \ldots, s$;

Step B [Gröbner-basis in $Z/(p_1 \cdots p_k)$ $[x_1, \ldots, x_n]$]:
By applying the Chinese remainder algorithm, construct a Gröbner-basis
$\Gamma^{(0)}=\{G_1^{(0)}, \ldots, G_s^{(0)}\}$ such that $\Gamma^{(0)} \equiv \Gamma^{(i)} \pmod{p_i}$, $i=1, \ldots, k$;

Step C [Gröbner-basis in $Q/(p_1 \cdots p_k)$ $[x_1, \ldots, x_n]$]:
Convert the integer coefficients in $\Gamma^{(0)}$ in such a way that an integer c is converted to a rational a/b satisfying $c \equiv a/b \pmod{p_1 \cdots p_k}$ and $0<|a|<\sqrt{p_1 \cdots p_k/2}$, and $0<b<\sqrt{p_1 \cdots p_k/2}$;

Step D: Check that the basis constructed in Step C actually is equal to Γ, the true basis over Q, or Γ' such that ideal$(\Gamma') \supseteq$ ideal(Γ).
We explain the above method in details.

3.1 The Number of Primes

We denote the Gröbner-basis constructed in the above Step C, *i.e.*, a Gröbner-basis in $Q/(p_1 \cdots p_k)[x_1, \ldots, x_n]$, by $\Gamma_{(p_1 \cdots p_k)}$. We note that we now have no method of knowing the value of k, the number of necessary primes, in advance of the computation, so we estimate k by checking the Gröbner-basis constructed. The estimation is done as follows: if $\Gamma_{(p_1 \cdots p_k)}=\Gamma_{(p_1 \cdots p_k p_{k+1})}$ then k is a candidate value. (This *does not* mean that k is a *sufficient* value.) Therefore, we calculate $\Gamma_{(p_1 \cdots p_i)}$ after the construction of each $\Gamma^{(i)}$.

3.2 Integer Interpolation

There are two algorithms for performing the above Step B: Newton's interpolation and Lagrange's interpolation, see [6], for example. Since we calculate $\Gamma^{(0)}$ defined in the above Step B after the construction of each basis $\Gamma^{(i)}$, Newton's algorithm is apparently suited for our computation. The algorithm, in the form for iterative use, is as follows.

Algorithm Newton's-INTERPOL [to be used iteratively.]
Input: a set of distinct primes $(p_1, \ldots, p_{i-1}, p_i)$;
a set of interpolation coefficients $(v_1, \ldots, v_{i-1})$;
an $(i-1)$st interpolation u and the i-th residue u_i;
Output: the i-th interpolation u such that $u \equiv u_j \pmod{p_j}$, $j=1, \ldots, i$;
the set of interpolation coefficients $(v_1, \ldots, v_{i-1}, v_i)$;
(I1) **if** $i=1$ **then** $u:=v_1:=u_1$; **return** u and (v_1);
(I2) Calculate integers w_j, $j=1, \ldots, i-1$, such that $w_j p_j \equiv 1 \pmod{p_i}$;
(I3) Calculate v_i as $v_i=(\cdots((u_i-v_1)w_1-v_2)w_2-\cdots-v_{i-1})w_{i-1} \pmod{p_i}$;
(I4) **return** $u:=u+v_i(p_1 \cdots p_{i-1})$ and $(v_1, \ldots, v_{i-1}, v_i)$. □

3.3 Conversion to Rationals

The conversion of integer to rational number modulo $p_1 \cdots p_k$ is based on the following well-known theorem:

Theorem [well-known]. *Let m, A, B be elements of N, satisfying $2AB<m$. For any $u \in Z$, there exists at most one rational number a/b such that*

$$-A \le a \le A \text{ and } 1 \le b \le B,$$
$$bu \equiv a \pmod{m}, \ \gcd(a, b)=1. \quad \square \qquad (6)$$

We use the theorem by setting $m=p_1 \cdots p_k$ and $A=B=[\sqrt{m/2}]$.

Given positive integers $m(=p_1 \cdots p_k)$ and u, the following algorithm calculates a rational number a/b such that $a/b \equiv u \pmod{m}$ [9, 10]; below, by $\bar{a}$ we mean (a_1, a_2, a_3).

Algorithm CONV: INT2RAT.
Input: u, modulus m, and $sqm=\sqrt{m/2}$, where $0<u<m$;
Output: integers a and b such that $a/b \equiv u \pmod{m}$, $-sqm<a<sqm$, $0<b<sqm$;
(R1) $\bar{a}:=(1, 0, m)$; $\bar{b}:=(0, 1, u)$;
(R2) **while** $b_3 \ge sqm$ **do**
$(q:=a_3/b_3$; $\bar{r}:=\bar{a}-q\bar{b}$; $\bar{a}:=\bar{b}$; $\bar{b}:=\bar{r})$;
(R3) **if** $b_3=0$ **then return** NIL*)
else if $|b_2|<sqm$
then return $(a:=\mathrm{sign}(b_2)b_3$, $b:=|b_2|)$
else return NIL*). □

*) If NIL is returned by **CONV:INT2RAT** then it means that there is no rational a/b satisfying (6) for modulus m. Such a case may happen when $\gcd(m, u) \neq 1$. For a given rational a/b, however, if $ub \equiv a \pmod{m}$ then we can recover a/b from u so long as m is sufficiently large [10].

In the above algorithm, the following relations always hold.

$$a_1 m+a_2 u=a_3, \ b_1 m+b_2 u=b_3. \qquad (7)$$

In the **while** loop of (R2), the values of a_3 and b_3 decrease steadily and the values of $|a_2|$ and $|b_2|$ increase steadily. After the **while** loop, we have $b_3<\sqrt{m/2}$ and if $|b_2|<\sqrt{m/2}$ then we find the desired rational a/b.

3.4 Final Check in $Q[x_1, \ldots, x_n]$

The Step D of the above gross algorithm is composed of two checks.

Check 1. Assure that $\Gamma_{(p_1 \cdots p_k)}$ is actually a Gröbner-basis over Q, by using Theorem 1;

Check 2. Assure that ideal$(\Gamma_{(p_1 \cdots p_k)})=(F_1, \ldots, F_r)$.

Note that if $\Gamma_{(p_1 \cdots p_k)}$ is of the form (4′) then the Check 1 is unnecessary, because of the well-known property:

if $\gcd(\mathrm{hpp}(G_i), \mathrm{hpp}(G_j))=1$ then $\mathrm{Spol}(G_i, G_j) \to\to_{G_i, G_j} 0$.

The Check 2 is rather complicated, and the method is described in 5. On the other hand, it is quite easy to check that ideal($\Gamma_{(p_1\cdots p_k)}$)$\supseteq(F_1,\ldots,F_r)$: we have only to check that every F_i, $1\leq i\leq r$, is M-reduced to 0 by $\Gamma_{(p_1\cdots p_k)}$. Depending on which check is adopted, a complete check or an incomplete but simple one, we have a complete algorithm or a probabilistic one. The complete algorithm is described in 5, and the probabilistic one in 4.

4. Probabilistic Algorithm

Summarizing the above considerations, we obtain modular algorithms for Gröbner-basis construction. In this section, we describe a probabilistic version of such an algorithm and prove the termination.

Algorithm Modular-GBASIS (I) (probabilistic version)
Input: a set of polynomials $\{F_1,\ldots,F_r\}\in Z[x_1,\ldots,x_n]$;
a set of distinct primes $\{p_1,p_2,\ldots\}$ of word-size;
Output: a Gröbner-basis $\Gamma=\{G_1,\ldots,G_s\}$ w.r.t. a given term order;

Step 0 [initialize]: $P:=1$; $i:=0$; $\Gamma^{(0)}:=$**nil**;

Step 1 [Gröbner-basis over $Z/(p_i)$]: $i:=i+1$;
Calculate $\Gamma^{(i)}$=a reduced normalized Gröbner-basis $\{G_1^{(i)},\ldots,G_s^{(i)}\}$ in $Z/(p_i)$ $[x_1,\ldots,x_n]$;

Step 2 [check the weak unlucky prime]:
if $P=1$ **then goto Step 3**
else if p_i is weak unlucky*1) **then goto Step 1**
else if all the previous primes are unlucky*1)
then $P:=1$ and $\Gamma^{(0)}:=$**nil**;

Step 3 [Gröbner-basis over $Z/(P\cdot p_i)$]:
if $P=1$ **then** $\Gamma^{(0)}:=\Gamma^{(i)}$ and **goto Step 4**;
By applying algorithm **Newton's-INTERPOL**,
update $\Gamma^{(0)}=\{G_1^{(0)},\ldots,G_s^{(0)}\}$ so that $\Gamma^{(0)}\equiv\Gamma^{(i)}$ (mod p_i);*2)

Step 4 [Gröbner-basis over $Q/(P\cdot p_i)$]:
Convert the integer coefficients in $\Gamma^{(0)}$ so that an integer c is converted to a rational a/b satisfying $c\equiv a/b$(mod $P\cdot p_i$), $b>0$ and $|a|$, $b<\sqrt{P\cdot p_i/2}$;*3)
Let $\Gamma_{(P\cdot p_i)}=G_1',\ldots,G_s'$ denote the Gröbner-basis obtained;

Step 5 **[check the termination]:**
if $P=1$ or $\Gamma_{(P)}\neq\Gamma_{(P\cdot p_i)}$ **then** $P:=P\cdot p_i$ and **goto Step 1**;
if Spol$(G_i',G_j')\to\to_{\Gamma_{(P)}}$ non-zero for any i and j, $1\leq i<j\leq r$, **then goto Step 1**
if for each $j=1,\ldots,r$, $F_j\to\to_{\Gamma_{(P)}}0$ **then** return $\Gamma_{(P)}$
else $P:=P\cdot p_i$ and **goto Step 1**.

*1) For the weak unluckiness check, see the beginning of 3.
*2) This is nothing but the Chinese remainder algorithm, hence $\Gamma^{(0)}$ is now a Gröbner-basis over $Z/(P\cdot p_i)$.
*3) There may not exist a rational a/b satisfying the condition. However, such a rational exists surely if P is enough large.

In the rest of this section, we show that the above algorithm calculates the correct Gröbner-basis almost always. Furthermore, even if the calculated Gröbner-basis Γ' is not equal to the Gröbner-basis Γ over Q, Γ' is still useful in many cases.

4.1 Termination of the Algorithm

Proposition 2. *Algorithm* Modular-GBASIS (I) *terminates.*

(Proof). We first note that the number of unlucky primes is finite. Now, suppose that some prime among $p_1,\ldots,p_k$ is lucky, then the check in Step 2 discards all the unlucky primes. Hence, without loss of generality, we have only to consider two cases, i) all the primes $p_1,\ldots,p_k,p_{k+1}$ are unlucky, and ii) all the primes $p_1,\ldots,p_k,p_{k+1}$ are lucky. First, consider the case i). Since the number of unlucky primes is finite, the case i) with $k\to\infty$ does not occur. Next, consider the case ii). Let M be the maximum-magnitude integer in the coefficients in Γ. Then, we can recover Γ from $\Gamma_{(p_1\cdots p_k)}$ so long as $p_1\cdots p_k\geq 2M^2$. Since $|M|$ is finite, the algorithm terminates in this case also. □

4.2 Probabilistic Nature of the Algorithm

Let us analyze the above cases i) and ii), given in **4.1**, in detail.

Case i): All the primes $p_1,\ldots,p_k,p_{k+1}$ are unlucky.
Modular-GBASIS (I) cannot exclude this case although the case is extremely rare to happen: it is quite rare that a word-sized prime is unlucky for the ideal $(F_1,\ldots,F_r)$, whose coefficients are moderate-sized, hence the possibility that all the primes $p_1,\ldots,p_{k+1}$ are unlucky is extremely small. Let $\Gamma_{(P)}=\{G_1',\ldots,G_s'\}$, then after the check in Step 5 we have

$$F_i=\sum u_{ij}G_j',\quad i=1,\ldots,r. \tag{8}$$

That is, $(F_1,\ldots,F_r)\subseteq(G_1',\ldots,G_s')$. This means that, when our problem is to solve the system of algebraic equations, every root of $\{G_1'=0,\ldots,G_s'=0\}$ is a root of the original system $\{F_1=0,\ldots,F_r=0\}$. Hence, even if all the primes $p_1,\ldots,p_{k+1}$ are unlucky, the resulting basis $\Gamma_{(P)}$ is useful in some cases.

Case ii): All the primes $p_1,\ldots,p_k$ are lucky.
Note that the condition $\Gamma_{(p_1\cdots p_k)}=\Gamma_{(p_1\cdots p_kp_{k+1})}$ is not enough to show that $\Gamma=\Gamma'$ over Q. This can be seen as follows. Suppose a rational coefficient a'/b' in Γ' corresponds to a coefficient a/b in Γ, then $\Gamma_{(p_1\cdots p_k)}=\Gamma_{(p_1\cdots p_kp_{k+1})}$ means $a/b\equiv a'/b'$ (mod $p_1\cdots p_kp_{k+1}$). The case $a/b\neq a'/b'$ may happen, for example, when $b=b'$ and $a=a'+c\times p_1\cdots p_{k+1}$. Therefore, the check in Step 5 is necessary.

If, however, the check in Step 5 is passed then the resulting basis is the required Gröbner-basis Γ as the following proposition proves.

Proposition 3. *If $p_1, \ldots, p_k$ are lucky then Γ' constructed by algorithm* Modular-GBASIS (I) *is identical to Γ, the reduced normalized Gröbner-basis over $\boldsymbol{Q}$.*
(Proof) The check in Step 5 assures that $\Gamma \subseteq \text{ideal}(\Gamma')$. Since $p_1, \ldots, p_k$ are lucky, we have $\text{ht}(G_i)=\text{ht}(G_i')$, $i=1, \ldots, s$. Therefore, Theorem 3 insists that $\text{ideal}(\Gamma)=\text{ideal}(\Gamma')$. Since the head coefficients are normalized, we have $\Gamma=\Gamma'$ by Theorem 2. □

5. Improvement and Completion of Algorithm

In this section, we improve algorithm Modular-GBASIS (I) in two points: one is to avoid zero-reduced S-polynomial construction in Step 1, and the other is to execute Step 4 efficiently. Furthermore, we present a complete algorithm which constructs a Gröbner-basis over $\boldsymbol{Q}$ successfully.

5.1 Avoiding Zero-reduced S-polynomial Construction

As we have mentioned in **1**, in the construction of Gröbner-basis of many elements, most computation time is spent to construct S-polynomials which are reduced to zero immediately. By preserving the history of basis construction process, we can avoid such zero-reduced S-polynomial construction for many primes, reducing the computation time largely. The history is the following list which is a slight modification of the ht-history defined in **3**:

$$\begin{aligned}\text{HIST}:=((&\#11\ \#12\ \text{ht}(\text{Spol}(F_{\#11}, F_{\#12}))),\\ &(\#21\ \#22\ \text{ht}(\text{Spol}(F_{\#21}, F_{\#22}))),\\ &\cdots\cdots\cdots\cdots\ ).\end{aligned} \tag{8}$$

Here, $\text{Spol}(F_{\#11}, F_{\#12})$ is a non-zero S-polynomial constructed first, $\text{Spol}(F_{\#21}, F_{\#22})$ is a non-zero S-polynomial constructed second, and so on.

We construct HIST by the basis construction process for the first several primes, say p_1, p_2 and p_3. The HIST thus constructed will be almost valid and, for the rest primes ($p_4, p_5, \ldots$, in this case), we construct only the S-polynomials registered in HIST sequentially. Suppose, for the i-th prime p_i ($i>3$ in this case), we calculate $Sp^{(i,j)}=\text{Spol}(F_{\#j1}, F_{\#j2})$ which is a non-zero S-polynomial constructed j-th. Let $Sp^{(*,j)}$ be $\text{Spol}(F_{\#j1}, F_{\#j2})$ saved in HIST. If $\text{ht}(Sp^{(i,j)}) \lhd \text{ht}(Sp^{(*,j)})$ then p_i is an unlucky prime hence we discard the p_i. If $\text{ht}(Sp^{(i,j)}) \rhd \text{ht}(Sp^{(*,j)})$ then all the primes $p_1, \ldots, p_{i-1}$ are unlucky and we initialize HIST by the basis construction for the prime p_i.

5.2 When Solving System of Algebraic Equations

When the problem is to solve the system of algebraic equations, there are two very useful ideas which are applicable to the case that the reduced system is of the form (4′). The first idea is on the term order $\rhd$. We have mentioned in **2** that the lexicographic term-order is used for reducing the system (2) to (4). If, however, the reduced system is of the form (4′), we can use the following term-order as proved in [7].

$$\begin{cases} x_n, \ldots, x_2 \rhd x_1, \\ \text{total-degree order for } \{x_2, \ldots, x_n\}. \end{cases}$$

In the actual implementation, we had better test this order first and, if we find that the reduced system is not of the form (4′), we apply the lexicographic order.

The second idea is on the completeness check of algorithm Modular-GBASIS (I). The number of roots of the system (2), with m multiple roots counted as m, is bounded by the so-called Bezout bound defined as follows.

$$\text{Bezout-bound}(\{F_1, \ldots, F_r\})=\prod_{i=1}^{r} \text{tdeg}(F_i),$$

where $\text{tdeg}(F)$ is the total-degree of polynomial F. On the other hand, if the reduced system is of the form (4′), the number of roots of the system is $\deg(G_1)$. Therefore, according to the analysis in **(4.2)**, we see that $\Gamma=\{G_1, \ldots, G_s\}$, of the form (4′), computed by Modular-GBASIS (I) is the correct reduced system if

$$\text{Bezout-bound}(\{F_1, \ldots, F_r\})=\deg(G_1).$$

5.3 Conversion to Polynomials with Rational Coefficients

Among the steps in Modular-GBASIS (I), Step 1 (Gröbner-basis construction over $Z/(p_i)$, $i=1, 2, \ldots$) is the most time-consuming step, and Step 4 (integer to rational conversion step) is also a time-consuming step, if executed naively, because of the following two reasons. First, algorithm CONV:INT2RAT treats long numbers and it is practically much more expensive than algorithm Newton's-INTERPOL. Second, integer to rational conversion is done for all the coefficients in $\Gamma^{(0)}$ (Gröbner-basis over $Z/(p_1 \ldots p_i)$) after each construction of $\Gamma^{(i)}$ (Gröbner-basis over $Z/(p_i)$). However, Step 4 can be executed quite efficiently by using the following propositions.

Proposition 4. *Let $p_1, \ldots, p_{k-1}$, p_k be primes, u_{k-1}, u_k and v_k* be integers satisfying

$$u_k=u_{k-1}+v_k p_1 \cdots p_{k-1}. \tag{10}$$

If $u_k \equiv a/b \pmod{p_1 \cdots p_{k-1} p_k}$, where a/b is rational, then $u_{k-1} \equiv a/b \pmod{p_1 \cdots p_{k-1}}$.
(Proof). By assumption, there exists an integer c_k such that

$$bu_k+c_k p_1 \cdots p_{k-1} p_k=a.$$

By this and (10), we see $u_{k-1} \equiv a/b \pmod{p_1 \cdots p_{k-1}}$. □

Proposition 5. *Let $p_1, \ldots, p_k$, p_{k+1} be distinct primes, u_k, u_{k+1} and v_{k+1} be integers satisfying*

$$u_{k+1}=u_k+v_{k+1} p_1 \cdots p_k. \tag{11}$$

If $u_k \equiv a/b \pmod{p_1 \cdots p_k}$ *and* $u_{k+1} \equiv a/b \pmod{p_{k+1}}$, *where* a/b *is a rational, then* $u_{k+1} \equiv a/b \pmod{p_1 \cdots p_k p_{k+1}}$.

(Proof). By assumption, there exist integers c_k and c such that

$$bu_k + c_k p_1 \cdots p_k = a,\ bu_{k+1} + cp_{k+1} = a.$$

These equations as well as Eq. (11) allow us to rewrite $bu_{k+1} - a$ as

$$\begin{aligned} bu_{k+1} - a &= b(u_k + v_{k+1} p_1 \cdots p_k) - a \\ &= b(v_{k+1} p_1 \cdots p_k) + (bu_k - a) \\ &= (bv_{k+1} - c_k) p_1 \cdots p_k, \\ bu_{k+1} - a &= -cp_{k+1}. \end{aligned}$$

Since $p_1, \ldots, p_k$, p_{k+1} are distinct primes, these equalities mean $bu_{k+1} - a \propto p_1 \cdots p_k p_{k+1}$. □

Proposition 4 tells us that we may skip integer to rational conversion until $p_1 \cdots p_k$ becomes large enough; enough suppose we skip the conversion until the k-th prime p_k and find a rational a/b such that $a/b \equiv u_{k+1} \pmod{p_1 \cdots p_k p_{k+1}}$, where u_{k+1} is the $(k+1)$-st Newton's interpolation, then we can check the appropriateness of a/b for the number $p_1 \cdots p_k$ by checking inequalities $0 \leq |a| < \sqrt{p_1 \cdots p_k / 2}$, and $0 < b < \sqrt{p_1 \cdots p_k / 2}$.

Proposition 5 tells us that, if we have calculated a rational a/b such that $a/b \equiv u_k \pmod{p_1 \cdots p_k}$, then we can check the appropriateness of a/b for the number $p_1 \cdots p_k p_{k+1}$ by checking congruence $a/b \equiv u_{k+1} \pmod{p_{k+1}}$.

Thus, we can perform Step 4 efficiently as follows.

(1) In addition to $\Gamma^{(0)} = \{G_1^{(0)}, \ldots, G_s^{(0)}\}$, the Gröbner-basis over $Z/(p_1 \cdots p_i)$, we prepare $\Gamma' = \{G_1', \ldots, G_s'\}$, the Gröbner-basis over $Q/(p_1 \cdots p_i)$, where each coefficient in Γ' is set of NIL initially and replaced by a rational as the computation proceeds.

(2) After the construction of $\Gamma^{(0)}$, we convert the coefficients in $\Gamma^{(0)}$ to rationals and save them into the corresponding places in Γ'. This coefficient conversion is made in a special way as follows.

(3) We first convert the second coefficients of $G_l^{(0)}$, $l = 1, \ldots, s$, and if the conversion fails for some $G_l^{(0)}$ then we stop Step 4 and goto Step 1. If all the second coefficients are converted successfully, we try to convert other coefficients of $G_l^{(0)}$, $l = 1, \ldots, s$. The conversion procedure is, however, stopped and go to Step 1 if the conversion fails for some coefficient.

(4) In each conversion of an integer to a rational, we utilize Propositions 4 and 5 to avoid unnecessary computation. (The integer to rational conversion is made only when the corresponding coefficient in Γ' is NIL.)

Note that we need not convert the head coefficient of $G_l^{(0)}$ because $\mathrm{lc}(G_l^{(0)}) = 1$. We convert the second coefficient of $G_l^{(0)}$ first to see whether $p_1 \cdots p_i$ is enough large or not. Using the above-mentioned method, most coefficients in $\Gamma^{(0)}$ are not converted until $p_1 \cdots p_i$ becomes large enough and the conversion is made only once or several times for most coefficients.

5.4 Completion of the Algorithm

As we have mentioned in **4**, algorithm Modular-GBASIS (I) is probabilistic in that it may return $\Gamma' = \{G_1', \ldots, G_s'\}$ such that $\Gamma' \neq \Gamma$ and ideal(Γ) $\subset$ ideal(Γ') when all the $p_1, \ldots, p_k$ are unlucky. This incompletendess can be removed easily by checking the existence of polynomials $V_{\rho\sigma} \in Q[x_1, \ldots, x_n]$, $\rho = 1, \ldots, r$, $\sigma = 1, \ldots, s$, such that

$$G_\sigma' = \sum_{\rho=1}^{r} V_{\rho\sigma} F_\rho,\ \sigma = 1, \ldots, s. \tag{12}$$

If there exist such $V_{\rho\sigma}$'s satisfying Eq. (12) then we see ideal(Γ') $\subseteq$ ideal(Γ), hence $\Gamma = \Gamma'$.

Algorithmic construction of $V_{\rho\sigma}$'s is made by a modular method as follows. Tracing the S-polynomial construction procedure over $Z/(p_i)$, we can easily calculate $V_{\rho\sigma}^{(i)}$ such that

$$G_\sigma^{(i)} \equiv \sum_{\rho=1}^{r} V_{\rho\sigma}^{(i)} F_\rho \pmod{p_i},\ \sigma = 1, \ldots, s. \tag{13}$$

Therefore, we can construct $V_{\rho\sigma}$ from $V_{\rho\sigma}^{(i)}$, $i = 1, \ldots, k$, $k+1$, just as we construct G_l' from $G_l^{(i)}$, $i = 1, \ldots, k$, $k+1$.

Algorithm Modular-GBASIS (II) (complete version)
Input: a set of polynomials $\{F_1, \ldots, F_r\} \in Z[x_1, \ldots, x_n]$;
a set of distinct primes $\{p_1, p_2, \ldots\}$ of word-size;
Output: a Gröbner-basis $\Gamma = \{G_1, \ldots, G_s\}$ w.r.t. a given term-order;

Step 0 [initialize]: $P := 1$; $i := 0$; $\Gamma^{(0)} :=$ **nil**; $\Phi^{(0)} :=$ **nil**;

Step 1 {Gröbner-basis and transformation matrix over $Z/(p_i)$}: $i := i+1$;
$\Gamma^{(i)} :=$ a reduced normalized Gröbner-basis $\{G_1^{(i)}, \ldots, G_s^{(i)}\}$ in $Z/(p_i)[x_1, \ldots, x_n]$;
$\Phi^{(i)} :=$ a set of polynomials $V_{\rho\sigma}^{(i)}$, $\rho = 1, \ldots, r$, $\sigma = 1, \ldots, s$, such that $G_\sigma^{(i)} = \sum V_{\rho\sigma}^{(i)} F_\rho \pmod{p_i}$;

Step 2 [check the weak unlucky prime]: same as in Modular-GBASIS (I);

Step 3 [Gröbner-basis and transformation matrix over $Z/(P \cdot p_i)$]:
if $P = 1$ **then** $\Gamma^{(0)} := \Gamma^{(i)}$ and $\Phi^{(0)} := \Phi^{(i)}$ and **goto Step 4**;
By applying algorithm **Newton's-INTERPOL**,
update $\Gamma^{(0)} = \{G_1^{(0)}, \ldots, G_s^{(0)}\}$ and $\Phi^{(0)} = \{V_{\rho\sigma}^{(0)}\}$
so that $\Gamma^{(0)} \equiv \Gamma^{(i)} \pmod{p_i}$ and $\Phi^{(0)} \equiv \Phi^{(i)} \pmod{p_i}$;

Step 4 [Gröbner-basis and transformation matrix over $Q/(P \cdot p_i)$]:
Convert the integer coefficients in $\Gamma^{(0)}$ and $\Phi^{(0)}$ so that an integer c is converted to a rational a/b satisfying

$c \equiv a/b \pmod{P \cdot p_i}$, $b>0$ and $|a|$, $b<\sqrt{P \cdot p_i/2}$;

Let $\Gamma_{(P \cdot p_i)}=\{G'_1, \ldots, G'_s\}$ and $\Phi_{(P \cdot p_i)}=\{V'_{\rho\sigma} \mid 1 \le \rho \le r, 1 \le \sigma \le s\}$ denote the Gröbner-basis and transformation matrix obtained;

Step 5 [check the termination]:

if $P=1$ or $\Gamma_{(P)} \neq \Gamma_{(P \cdot p_i)}$ or $\Phi_{(P)} \neq \Phi_{(P \cdot p_i)}$ **then** $P:=P \cdot p_i$ and **goto Step 1**;

if $\mathrm{Spol}(G'_i, G'_j) \to \to_{\Gamma_{(P)}}$ nonzero for any i and j, $1 \le i < j \le r$, **then goto Step 1**;

if for each $j=1, \ldots, r$, $F_j \to \to_{\Gamma_{(P)}} 0$ and

for each $\sigma=1, \ldots, s$, $G'_\sigma=\sum_\rho V'_{\rho\sigma} F_\rho$ **then** return $\Gamma_{(P)}$

else $P:=P \cdot p_i$ and **goto Step 1**.

Although the complete modular algorithm mentioned above is simple in principle, actual implementation is complicated. We think the simpler probabilistic version will be almost sufficient for solving system of algebraic equations. The complete algorithm will be necessary only when the rigorously correct Gröbner-basis is required.

6. Empirical Study

We have implemented algorithm Modular-GBASIS (I) on the algebra system GAL and studied the effectiveness of the algorithm by the following three examples.

Example 1. (Klein's equations)

$$P_1=(x_1^6+x_2^6)+522(x_1^5x_2-x_1x_2^5) - 10005(x_1^4x_2^2+x_1^2x_2^4)-u_1=0,$$

$$P_2=-(x_1^4+x_2^4)+228(x_1^3x_2-x_1x_2^3)-494x_1^2x_2^2-u_2=0,$$

$$P_3=x_1x_2(x_1^2+11x_1x_2-x_2^2)^5-u_3=0.$$

We calculate the reduced Gröbner-basis of (P_1, P_2, P_3) with the ordering $x_1, x_2 \triangleright u_1, u_2, u_3$, where total-degree order is assumed for $\{x_1, x_2\}$ and $\{u_1, u_2, u_3\}$, respectively. By this, we want to derive a polynomial relation satisfied by u_1, u_2 and u_3, which is $u_1^2+u_2^3-1728u_3=0$.

Example 2. (Katsura's equation #3)

$$P_1=2(x_4^2+x_3^2+x_2^2)+x_1^2-x_1=0,$$

$$P_2=2(x_4x_3+x_3x_2+x_2x_1)-x_2=0,$$

$$P_3=2(x_4x_2+x_3x_1)+x_2^2-x_3=0,$$

$$P_4=2(x_4+x_3+x_2)+x_1-1=0.$$

This equation (as well as the next one) is derived by Katsura in his theory of spin-grass [13]. We calculate the reduced Gröbner-basis of (P_1, P_2, P_3, P_4) with the ordering $x_4, x_3, x_2 \triangleright x_1$, where the total-degree order is assumed for $\{x_2, x_3, x_4\}$. The result is of form (4').

Example 3. (Katsura's equation #4)

$$P_1=2(x_5^2+x_4^2+x_3^2+x_2^2)+x_1^2-x_1=0,$$

Table 1 Comparison of modular algorithm (M1 & M2) and conventional algorithm (C). Timing data are in milli-seconds on a FACOM-M780 computer. (k is the number of primes used in each computation.)

No.	*Algorithm* C (conventional)	*Algorithm* M1 (HIST=nil)	*Algorithm* M2 (use HIST)
Example 1	221	591 ($k=5$)	389 ($k=5$)
Example 2	347	385 ($k=5$)	386 ($k=5$)
Example 3	>600,000	28,593 ($k=16$)	28,709 ($k=16$)

$$P_2=2(x_5x_4+x_4x_3+x_3x_2+x_2x_1)-x_2=0,$$

$$P_3=2(x_5x_3+x_4x_2+x_3x_1)+x_2^2-x_3=0,$$

$$P_4=2(x_5x_2+x_4x_1+x_3x_2)-x_4=0,$$

$$P_5=2(x_5+x_4+x_3+x_2)+x_1-1=0.$$

We calculate the reduced Gröbner-basis of $(P_1, \ldots, P_5)$ similarly as Example 2. The result is of form (4').

Table 1 shows timing data of three algorithms: algorithm C is the conventional one based on the rational arithmetic; M1 and m2 are modular algorithms, where M1 does not utilize the HIST (history of *Spol* construction) while M2 does. We used primes of order 10^6, and we have encountered no unlucky primes in these examples.

The number k in Table 1 shows the size of coefficients (rationals in this case) of the basis polynomials obtained. Example 1 is a "small-sized" problem for which the intermediate coefficient growth is quite weak, and the modular method is not effective for this case. Example 2 causes "weak" intermediate coefficient growth, hence the modular method is not bad compared with the conventional method although the Gröbner-basis is calculated five times. Example 3 causes "strong" intermediate coefficient growth, and we find the modular method is actually quite effective for such problems. Comparison of algorithms M1 and M2 indicates that the technique for avoiding the construction of zero-reduced S-polynomials is not effective in our examples (it is slightly effective for Example 1), but we expect it to be quite effective for more large-sized problems.

Acknowledgement

The authors give a special thank to Mr. K. Yokoyama of IIAS-SIS, FUJITSU LIMITED for useful discussion and valuable comments on this article.

References

0. This paper is a revised version of "A Modular Gröbner Basis Method for Algebraic Equations" by the present authors, Report of RIKEN Symposium (informal document), Feb. 1988.

1. BÖGE, W., GEBAUER, R. and KREDEL, H. Some Examples for Solving Systems of Algebraic Equations by Calculating Gröbner-basis, *J. Symb. Comp.* **2** (1986), 83-98.

2. BUCHBERGER, B. An Algorithm for Finding a Basis for the

Residue Class Ring of a Zero-dimensional Polynomial Ideal (German.) Ph.D. Thesis, Math. Inst., Univ. of Insbruck, Austria (1965).
3. BUCHBERGER, B. Gröbner Bases: An Algorithmic Method in Polynomial Ideal Theory, in *Multidimensional Systems Theory*, N. K. Bose, ed., 184–232, D. Reidel Publ. Comp., 1985.
4. EBERT, G. Some Comments on the Modular Approach to Gröbner-bases, *SIGSAM Bulletin* **17** (1983), 28–32.
5. KOBAYASHI, H., MORITSUGU, S. and HOGAN, R. W. Solving Systems of Algebraic Equations, paper presented at ISSAC'88, *Lec. Notes Comp. Sci.* **358** (1989), 139–149.
6. SASAKI, T. Suushiki-shori (Formula Manipulation), published by IPSJ (sold by Ohm publ. Co. Tokyo), 1981.
7. SASAKI, T. Some Algebraic Algorithms based on Head Term Elimination over Polynomial Rings, paper presented at EUROCAL'87, *Lec. Notes Comp. Sci.* **378** (1989), 348–354.
8. TRINKS, W. On Improving Approximate Results of Buchberger's Algorithm by Newton's Method, *SIGSAM Bulletin* **18** (1984), 7–11.
9. TRINKS, W. L. Über Buchbergers Verfahren, Systeme algebraischer Gleichungen zu lösen, *J. Number Theory* **10** (1978), 475–488.
10. WANG, P. S. A *p*-adic Algorithm for Univariate Partial Fractions, *Proc. of* SYMSAC'81 (1981), 212–217.
11. WANG, P. S., GUY, M. J. T. and DAVENPORT, J. H. P-adic Reconstruction of Rational Numbers, *SIGSAM Bulletin* **16** (1982), 2–3.
12. WINKLER, F. *p*-adic Methods for the Computation of Gröbner-bases, *J. Symb. Comp.* **6** (1988), 287–304.
13. KATSURA, S. *Prog. Theor. Phys. Suppl.* No. 87 (1986), 139.

(Received October 31, 1988; revised August 8, 1989)