Incomplete Gröbner basis as a preconditioner for polynomial systems

Yang Sun, Yu-Hui Tao, Feng-Shan Bai *

Department of Mathematical Sciences, Tsinghua University, Beijing, 100084, PR China

A R T I C L E   I N F O

Article history:
Received 28 June 2007
Received in revised form 15 December 2007

MSC:
65H20
65H10

Keywords:
Incomplete Gröbner basis (IGB)
Homotopy continuation method
Subtraction polynomial (S-polynomial)
Precondition

A B S T R A C T

Precondition plays a critical role in the numerical methods for large and sparse linear systems. It is also true for nonlinear algebraic systems. In this paper incomplete Gröbner basis (IGB) is proposed as a preconditioner of homotopy methods for polynomial systems of equations, which transforms a deficient system into a system with the same finite solutions, but smaller degree. The reduced system can thus be solved faster. Numerical results show the efficiency of the preconditioner.

© 2008 Elsevier B.V. All rights reserved.

1. Introduction

Consider polynomial systems of equations

\[ P(x) = \begin{pmatrix} p_1(x) = p_1(x_1, \ldots, x_n) = 0 \\ p_2(x) = p_2(x_1, \ldots, x_n) = 0 \\ \vdots \\ p_n(x) = p_n(x_1, \ldots, x_n) = 0 \end{pmatrix} \]

(1)

where \( p_k(x) \) \( (k = 1, 2, \ldots, n) \) are polynomials of \( x = (x_1, \ldots, x_n)^T \) with degree \( d_k \). They arise in many applications. It is well known that the number of isolated solutions of \( P(x) = 0 \) is bounded above by the total degree \( TD = \prod_{k=1}^n d_k \). When the system becomes large, it would normally become sparse. This is similar to the case of linear systems. The sparsity for nonlinear algebraic system often gives deficiency. Namely the number of the isolated solutions of \( P(x) = 0 \) may be far less than the total degree \( TD \). A simple example is the matrix eigenvalue problem, where \( TD = 2^n \) and the number of the isolated solutions is only \( n \). For large and sparse linear systems of equations, precondition plays an important role in the numerical methods. Here in this paper we will show that it is also true for nonlinear algebraic systems.

Morgan \[9,10\] develops linear reduction as an automatic reformulation, where the symbolic reduction can be regarded as a simple precondition for polynomial continuation methods. Verschelde \[17,18,16\] extends the idea to nonlinear reduction using the subtraction polynomial, a key concept in Gröbner basis \[3\], and a mathematical software PHCpack (PHC) \[19\] is developed. However, in practice, many polynomial systems may not satisfy the conditions for the reductions in PHC. Note that the Gröbner basis can be considered as an extension of the Gaussian elimination for nonlinear algebraic system. Here we propose another reduction method based on the incomplete Gröbner basis (IGB), which can naturally be regarded as a nonlinear version of the incomplete LU decomposition.

* Corresponding author.
E-mail address: fbai@math.tsinghua.edu.cn (F.-S. Bai).

0377-0427/$ – see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2008.05.054
This paper is organized as follows. In the second section we review the concept of linear reduction and the basic idea of Verschelde’s methods. The nonlinear reduction method by incomplete Gröbner basis is proposed in Section 3. The computational results by the method with IGB are presented in Section 4. The last section gives conclusions and some discussions on the future researches.

2. Gröbner bases

There are different term orders for multivariate polynomials. There is not such an issue for single variable polynomials. For the purpose of elimination, the graded lexicographic ordering is commonly used to order the monomials in Gröbner basis computation and the ground field, on which the polynomial rings are defined, is assumed to be real.

Definition 2.1. Let \( d_i = (d_{i1}, d_{i2}, \ldots, d_{im}) \). The lexicographic ordering \(<_L\) for monomials is defined as

\[ X^{d_1} <_L X^{d_2} \iff \exists k \leq n \text{ such that } \forall m < k \ d_{1m} = d_{2m} \text{ and } d_{1k} < d_{2k} .\]

where \( X^{d_i} \) denotes the monomial \( x_1^{d_{i1}} x_2^{d_{i2}} \ldots x_n^{d_{in}} \) for simplicity.

Definition 2.2. Let \( d_i = (d_{i1}, d_{i2}, \ldots, d_{im}) \). The graded lexicographic ordering \(<_{GL}\) is defined as

\[ X^{d_1} <_{GL} X^{d_2} \iff \sum_i d_{1i} < \sum_i d_{2i}, \text{ or } \sum_i d_{1i} = \sum_i d_{2i}, \text{ but } X^{d_1} <_L X^{d_2} .\]

Definition 2.3. \( \ell(f) \) denotes the leading term of polynomial \( f \), \( lc(f) \) denotes the coefficient of \( \ell(f) \) and \( \ell m(f) \) means the leading monomial of \( f \). The least common multiple of the polynomials \( f_1 \) and \( f_2 \) is denoted by \( \text{lcm}(f_1, f_2) \). The Subtraction polynomial \((S\text{-polynomial})\) of \( f_1, f_2 \) is defined by

\[ S(f_1, f_2) = \ell c(f_2) * \frac{\text{lcm}(\ell(t), \ell(f_2))}{\ell(t)} \star f_1 - \ell c(f_1) * \frac{\text{lcm}(\ell(f_1), \ell(f_2))}{\ell(f_2)} \star f_2 .\]

Example 2.1. Consider a nonlinear algebraic system \( F(x) = 0 \), where

\[ F(x) = \begin{cases} f_1(x) = x^3 - x_1 = 0 \\ f_2(x) = x_1^3 x_2 + 1 = 0 . \end{cases} \]

The total degree of this system equals nine. There are only two solutions for the system \( F(x) = 0 \). Replace \( f_1 \) by the \( S\text{-polynomial} \( S(f_1, f_2) = -x_1 x_2 - x_1 \), one obtains

\[ \tilde{F}(x) = \begin{cases} \tilde{f}_1(x) = -x_1 x_2 - x_1 = 0 \\ \tilde{f}_2(x) = x_1^3 x_2 + 1 = 0. \end{cases} \]

The total degree of the reduced system \( \tilde{F} \) equals six. Instead, if \( f_2 \) in \( F \) is replaced by the \( S\text{-polynomial} \( S(f_1, f_2) \), one obtains

\[ \underline{\tilde{F}}(x) = \begin{cases} \underline{\tilde{f}}_1(x) = x_1^3 - x_1 = 0 \\ \underline{\tilde{f}}_2(x) = -x_1 x_2 - x_1 = 0. \end{cases} \]

The infinite set \( S = \{ x \in (0, a), \ a \in \mathbb{C} \} \) solves the system \( \underline{\tilde{F}}(x) = 0 \). This shows that replacing a polynomial directly by an \( S\text{-polynomial} \) may not work sometimes.

Definition 2.4. Two polynomial systems \( F(x) = 0 \) and \( \tilde{F}(x) = 0 \) are said to be equivalent if and only if they have the same finite solutions with the same multiplicities.

By using varieties and ideals as in [3], this question can be rewritten as follows. Let \( \langle F \rangle \) denotes the ideal generated by the polynomials in \( F = F(x) \).

Theorem 2.1. \( F \) and \( \tilde{F} \) are equivalent if and only if \( \langle F \rangle = \langle \tilde{F} \rangle \).

Corollary 2.2. Consider \( F = \{ f_1, f_2, \ldots \} \) and \( \tilde{F} = \{ S(f_1, f_2), f_2, \ldots \} \). If \( f_1 \in \langle S(f_1, f_2), f_2 \rangle \), then \( F \) and \( \tilde{F} \) are equivalent.

Corollary 2.3. Let \( F \) and \( \tilde{F} \) be defined as Corollary 2.2. If \( \text{lcm}(\ell(f_1), \ell(f_2)) = \ell(f_1) \), then \( F \) and \( \tilde{F} \) are equivalent.

In the Example 2.1 one can see that \( f_1 = -x_1^2 S(f_1, f_2) - x_1 f_2 \) does not satisfy Corollary 2.2. However, the \( \tilde{F} \), which is obtained by replacing \( f_2 \) in \( F \), is not equivalent to \( F \) because \( f_2 \notin \langle S(f_1, f_2), f_1 \rangle \).

A condition under which a polynomial can be replaced by an \( S\text{-polynomial} \) without changing the finite solutions is called a replacement criterion. Verschelde [17] gives two replacement criteria.
Definition 2.5. The Residue-polynomial (or R-polynomial) \( R(f_1, f_2) \) of the polynomials \( f_1 \) and \( f_2 \) is defined by

\[
R(f_1, f_2) = f_1 - f_2 * \text{lcm}(f_1) * \text{lt}(f_1)/\text{Rt}(f_2),
\]

where \( \text{Rt}(f_2) \) is defined as the largest term in the polynomial \( f_2 \) so that \( \text{Rt}(f_2)|\text{lt}(f_1) \).

Observe that if \( \text{Rt}(f_2) = 0 \), then \( R(f_1, f_2) = f_1 \). Therefore \( R(f_1, f_2) \) eliminates \( \text{lt}(f_1) \) whenever \( \text{Rt}(f_2) \neq 0 \).

Theorem 2.4. If \( R(R(f_1, f_2), S(f_1, f_2)) = 0 \) or \( R(R(f_1, S(f_1, f_2)), f_2) = 0 \), then

\[
f_1 \in \langle S(f_1, f_2), f_2 \rangle.
\]

Theorem 2.5. If \( \text{lcm}(\text{lt}(f_1), \text{lt}(f_2)) = \text{lt}(f_1) \), then

\[
R(R(f_1, f_2), S(f_1, f_2)) = 0.
\]

The nonlinear reductions are constructed by the two criteria above and used in PHC as preconditioners for homotopy methods. There are three additional parameters in the stopping criteria for the algorithms in PHC. Moreover, in practice not so many systems could satisfy the replacing criteria.

3. Preconditioning with incomplete Gröbner basis

The nonlinear reduction algorithm provides an efficient preprocessor for solving polynomial systems with homotopy continuation methods. However, due to the computational complexity of the Gröbner basis, it is unpractical to try for every \( S \)-polynomial. We propose an algorithm which can stop automatically without any additional criteria and give the resulting reduction system to most of polynomial systems in polynomial time. Analogously, by applying \( S \)-polynomial, we can eliminate those terms with the highest degree and consequently reduce the degree of a polynomial.

Proposition 3.1. There exists monomial \( X^d \) such that

\[
\langle X^d f_1, f_2 \rangle = \langle S(f_1, f_2), f_2 \rangle.
\]

Proof. By the definitions in Section 2, one has

\[
S(f_1, f_2) = \text{lc}(f_2) * \frac{\text{lcm}(\text{lt}(f_1), \text{lt}(f_2))}{\text{lt}(f_1)} * f_1 - \text{lc}(f_1) * \frac{\text{lcm}(\text{lt}(f_1), \text{lt}(f_2))}{\text{lt}(f_2)} * f_2
\]

where

\[
X^{d_1} = \frac{\text{lcm}(\text{lt}(f_1), \text{lt}(f_2))}{\text{lt}(f_1)}, \quad X^{d_2} = \frac{\text{lcm}(\text{lt}(f_1), \text{lt}(f_2))}{\text{lt}(f_2)}.
\]

This clearly gives the result. \( \square \)

The zero set of polynomial system \( F = 0 \) is denoted by \( V(F) \). [3] gives the following.

Theorem 3.2. Let \( F = V(f_1, \ldots, f_s), G = V(g_1, \ldots, g_t) \). Then

\[
F \cap G = V(f_1, \ldots, f_s, g_1, \ldots, g_t).
\]

\[
F \cup G = V(f_i g_j; i = 1 \ldots s, j = 1 \ldots t).
\]

Proposition 3.3. There exists a monomial \( X^d \) such that

\[
V(S(f_1, f_2), f_2) = V(X^d, f_2) \cup V(f_1, f_2).
\]

Proof. By Proposition 3.1 and Theorem 3.2 we have

\[
V(S(f_1, f_2), f_2) = V(X^d f_1, f_2) = V(X^d f_1) \cap V(f_2)
\]

\[
= (V(X^d) \cup V(f_1)) \cap V(f_2)
\]

\[
= (V(X^d) \cap V(f_2)) \cup (V(f_1)) \cap V(f_2)
\]

\[
= V(X^d, f_2) \cup V(f_1, f_2).
\]

Hence the result follows. \( \square \)

Proposition 3.3 shows that the resulting system obtained in replacing the \( f_1 \) by the \( S \)-polynomial with lower degree contains the root set of the initial system. That is to say there may be some additional roots in the new system, but some components of these additional roots are zero. So we just need to verify the root with zero components by evaluating the initial system. A straightforward idea is to select the replacing system from the Gröbner basis. Buchberger Algorithm for Gröbner
Given the admissible term order, for all $f \in I = \langle f_1, \ldots, f_n \rangle$, there exist $h_i \in P$, such that $f = \sum_{i=1}^{n} h_i g_{n-i}$, where $(n_1, \ldots, n_n)$ is the permutation of $(1, \ldots, n$). $lm(h_i g_{n-i}) \geq lm(h_{i+1} g_{n-i+1})$, $i = 1, \ldots, n - 1$. This is the so-called descending combination with respect to $(g_1, \ldots, g_n)$. If $lm(h_i g_{n-i}) > lm(h_{i+1} g_{n-i+1})$, it is called strict descending combination of $f$.

**Assumption A.** Assume $F = \langle f_1, f_2, \ldots, f_l \rangle$. Let $G = \langle g_1, \ldots, g_n \rangle$ be Incomplete Gröbner Basis of $F$, satisfying $deg(S(g_i, g_i)) \leq D$ for all $(i, j)$.

The Assumption A is assumed throughout the rest of this section.

**Theorem 3.5.** For all $f \in (F)$, $deg(f) \leq D$, there exist strict descending combination of $f$ with respect to $(g_1, \ldots, g_n)$.

**Proof.** Since $G$ is a generating set, so there exists a descending combination of $f$ with respect to $(g_1, \ldots, g_n)$. Denote

$$f = \sum_{i=1}^{n} h_i g_i, \quad \text{where } lm(h_i g_i) \geq lm(h_{i+1} g_{i+1}), i = 1, \ldots, n - 1.$$ 

If $lm(h_i g_i) = lm(h_{i+1} g_{i+1})$, then there is a monomial $t$, such that

$$tlcm(lm(g_i), lm(g_{i+1})) = lm(h_i g_i).$$

$$tS(g_i, g_{i+1}) = t \left( \frac{Lcm(lm(g_i))}{Lcm(lm(g_{i+1}))} g_{i+1} - \frac{Lcm(lm(g_{i+1}))}{Lcm(lm(g_i))} g_i \right) = lm(h_i) g_{i+1} - lm(h_{i+1}) g_i.$$ 

By Proposition 3.4, $S(g_i, g_{i+1}) \rightarrow 0$, i.e.

$$S(g_i, g_{i+1}) = \sum_{i=1}^{n} k_i g_i, \quad \text{where } lm(k_i g_i) \leq lm(S(g_i, g_{i+1})).$$ 

Hence

$$h_i g_i = lt(h_i) g_i + h'_i g_i = lc(h_i)(lm(h_{i+1}) g_{i+1} + tS(g_i, g_{i+1})) + h'_i g_i$$

$$= lc(h_i) \left( lm(h_{i+1}) g_{i+1} + t \sum_{i=1}^{n} k_i g_i \right) + h'_i g_i.$$
Substituting it for $h_i g_i$, we have a new combination
\[ f = \sum_{j=1}^{n} h'_j g_j, \]
where
\[ h'_j = \begin{cases} h_j + tk_j & j \neq i, i + 1 \\ h'_j + tk_j & j = i \\ (lc(h_j) + lc(h_j))lm(h_j) + tk_j & j = i + 1. \end{cases} \]
For $j < i$, we have $lm(tk_j) \leq lm(S(g_i, g_{i+1})) < lm(h_i g_i) \leq lm(h_i g_i)$, that is $lm(h'_j g_j) = lm(h_i g_i)$. So in the descending combination constructed by the new combination the order of first $i - 1$ terms do not change. As $lm(h'_j g_j) < lm(h'_{i+1} g_{i+1})$, we decrease the number of equalities. Thus by finite steps, we get strict descending combination. \(\square\)

**Corollary 3.6.** For all $f \in \langle F \rangle$, $\deg(f) \leq D$, then $f$ reduce to zero via $G$. That is to say
\[ \exists h_i \in \mathcal{P}, \text{ such that } f = \sum_{i=1}^{n} h_i g_i, \text{ where } \deg(h_i g_i) \leq \deg(f). \]

**Corollary 3.7.** Let $I = \langle F \rangle$. Then for all $h \in \mathcal{P}$ with $\deg(h) \leq D$, $\text{Can}(h, I)$ can be computed by Buchberger reduction of $h$ via $G$. Here $\text{Can}(h, I)$ means canonical form or normal form of $h$ with respect to $I$.

**Theorem 3.8.** Let $G'$ be the reduced Gröbner basis of $I$, $G_D$ be the polynomial set of degree less than $D$ in $G'$. Then $G = G_D$.

**Proof.** Let $G = \{g_1, \ldots, g_i\}$, $G' = \{f_1, \ldots, f_s\}$. By the result of Corollary 3.6, for any $f \in G_D$ we have
\[ f = \sum_{i=1}^{s} h_i g_i, \quad g_i \in G, \quad \deg(h_i g_i) \leq \deg(f) \leq D. \]
So there exist $g_i$ such that $lt(g_i) | lt(f)$. As $G'$ is the reduced Gröbner basis of $I$, and $g_i \in I$, there exist $f_j \in G'$ such that $lt(f_j) | lt(g_i) | lt(f)$. So $f_j = f$, $lt(g_i) = lt(f)$. If $lt(f - g_i)$ is in $f$, there must exist $f_k \in G'$ such as $lt(f_k) | lt(f - g_i)$. It is contradicting to that $G'$ is reduced. If $lt(f - g_i)$ is in $g_i$, it is in contradiction with that $G$ is reduced. So we have $f = g_i \in G$, which gives $G \supseteq G_D$. \(\square\)

Analogously, we can prove that $G_D \supseteq G$. Hence the result follows.

Here we give a rule that can be used to check the output of incomplete Gröbner basis is a subset of the reduced Gröbner base with the maximal degree $D$, that is all $(i, j)$ satisfy $\text{deg}(S(g_i, g_j)) \leq D$. For generic polynomial system, Theorem 3.8 may not always be true. [4,1] give other results.

**Theorem 3.9.** Let $F = \{g_1, \ldots, g_i\}$, $I = \langle F \rangle$ and let $h$ satisfy $\deg(h) \leq D$, $h - \text{Can}(h, I) = \sum_{i=1}^{l} p_i g_i$ with $\deg(p_i g_i) \leq D$. Let $G$ be the Incomplete Gröbner Basis of $F$. Then $\text{Can}(h, I)$ can be computed by Buchberger reduction of $h$ via $G$.

Hence the selecting criteria from incomplete Gröbner bases are as follows.

1. Contain all the information of the polynomials in the initial system
2. Keep the degrees lower, as possible
3. Preserve the sparse structure, as possible

For the polynomial system in the Example 2.1, the $S$-polynomials generated by the algorithm are
\[ x_1^3 - x_1, \quad x_1^2 x_2 + 1, \quad -x_1 x_2 + x_1, \quad -x - 1^3 - x_1 x_2, \quad -x_1^2 + 1, x_2 + 1, \]
and the incomplete Gröbner basis is
\[ -x_1^2 + 1, \quad x_2 + 1. \]
Thus the total degree of the resulting system equals to 2 which is the same as the numbers of the root.

**4. Numerical results**

In the following examples, results by IGB and PHC are both given. All experiments are conducted on a 1.5 GHz Intel processor.
Example 4.1. Consider a system of polynomials $F(w_1, w_2, x_1, x_2) = 0$,
\[
F = \begin{cases}
    f_1 = w_1 + w_2 - 1 \\
    f_2 = w_1 x_1 + w_2 x_2 - 1 \\
    f_3 = w_1 x_1^2 + w_2 x_2^2 + 1 \\
    f_4 = w_1 x_1^3 + w_2 x_2^3 - 1.
\end{cases}
\]
This is a Gaussian quadrature formula with 2 knots and 2 weights from data gaukwa2. The total degree equals 24, and the number of the roots is 2.
Using PHC we obtain the following reduction system
\[
\tilde{F} = \begin{cases}
    \tilde{f}_1 = x_1 + x_2 + 1 \\
    \tilde{f}_2 = x_2^2 + x_2 \\
    \tilde{f}_3 = w_1 + w_2 - 1 \\
    \tilde{f}_4 = 2 w_2 x_2 + w_2 - x_2 - 2.
\end{cases}
\]
The total degree of $\tilde{F}$ is 4.
By Algorithm IGB we obtain the following reduction system
\[
\hat{F} = \begin{cases}
    \hat{f}_1 = x_1 + x_2 + 1 \\
    \hat{f}_2 = x_2^2 + x_2 \\
    \hat{f}_3 = 3 x_2 + w_1 + 1 \\
    \hat{f}_4 = 3 x_2 - w_2 + 2.
\end{cases}
\]
The total degree of $\hat{F}$ is 2. We can further verify that the full reduced Gröbner basis is exactly $\{\tilde{f}_1, \tilde{f}_2, \tilde{f}_3, \tilde{f}_4\}$.
Comparing with the results by PHC, the method by IGB gives the reduction system with lower degree. Moreover, the method with IGB can work for some systems that PHC does not work.

Example 4.2. Consider a system $F(a, b, c, d, t, u, v, w) = 0$,
\[
F = \begin{cases}
    f_1 = a + b - 1 \\
    f_2 = c + d + 1 \\
    f_3 = t a + u b - v c - w d + 1 \\
    f_4 = v a + w b + t c + u d - 2 \\
    f_5 = a t^2 - a v^2 - 2 c t v + b u^2 - b w^2 - 2 d u w - 1 \\
    f_6 = c t^2 - c v^2 + 2 a t v + d u^2 - d w^2 + 2 b u w + 1 \\
    f_7 = a t^3 - 3 a t^2 v + c v^3 - 3 c v t^2 + b u^3 - 3 b u w^2 + d u^3 - 3 d u w^2 - b w^3 + 3 b w u^2 - 1.
\end{cases}
\]
This is a problem from the data 'heart' [11,13,14]. The total degree of this system equals 576, but there are only 4 solutions.
PHC does not work for this problem. By the algorithm with IGB we obtain the following reduction system $\hat{F}$.
\[
\hat{F} = \begin{cases}
    \hat{f}_1 = 13 v + 13 w + 1 \\
    \hat{f}_2 = 13 t + 13 u + 5 \\
    \hat{f}_3 = 250 d - 572 u + 104 w + 19 \\
    \hat{f}_4 = 250 c + 572 u - 104 w + 231 \\
    \hat{f}_5 = 250 b + 104 u + 572 w - 83 \\
    \hat{f}_6 = 250 a - 104 u - 572 w - 167 \\
    \hat{f}_7 = 26 u w + u + 5 w - 1 \\
    \hat{f}_8 = 13 u^2 - 13 w^2 + 5 u - w - 5.
\end{cases}
\]
The total degree of $\hat{F}$ is reduced to 4. We can verify that both of the system $F$ and $\tilde{F}$ have the same roots, and
\[
\{\hat{f}_1, \hat{f}_2, \hat{f}_3, \hat{f}_4, \hat{f}_5, \hat{f}_6, \hat{f}_7, \hat{f}_8, 676 w^3 + 78 w^2 - 31 u + 287 w + 5\}
\]
gives the full reduced Gröbner basis.
Examples above show the efficiency of the algorithm by the incomplete Gröbner basis. In Table 1, more numerical results by PHC and IGB are presented. Most of the problems computed are concrete problems in science and engineering. It shows the performance of the two algorithms on not only the total degree, but also the multi-homogeneous Bezout number and the mixed volume.
Table 1
Numerical results

<table>
<thead>
<tr>
<th>Data</th>
<th>Dim</th>
<th>Res</th>
<th>No_reduced</th>
<th>PHC</th>
<th>IGB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overcon</td>
<td>2</td>
<td>3</td>
<td>12/5/5/3</td>
<td>4/3/3/3</td>
<td>4/3/3/3</td>
</tr>
<tr>
<td>Wood</td>
<td>4</td>
<td>9</td>
<td>36/25/16/9</td>
<td>16/16/9/9</td>
<td>16/9/9/9</td>
</tr>
<tr>
<td>Quadfor2</td>
<td>4</td>
<td>2</td>
<td>24/11/11/4</td>
<td>2/2/2/2</td>
<td>2/2/2/2</td>
</tr>
<tr>
<td>Quadgrid</td>
<td>5</td>
<td>5</td>
<td>120/10/10/10</td>
<td>8/8/8/8</td>
<td>8/8/6/6</td>
</tr>
<tr>
<td>Eco5</td>
<td>5</td>
<td>8</td>
<td>54/20/16/8</td>
<td>24/10/14/8</td>
<td>24/10/8/8</td>
</tr>
<tr>
<td>Gaukwa2</td>
<td>4</td>
<td>2</td>
<td>24/11/11/5</td>
<td>4/2/2/2</td>
<td>2/2/2/2</td>
</tr>
<tr>
<td>Gaukwa3</td>
<td>6</td>
<td>6</td>
<td>720/225/225/49</td>
<td>360/111/85/35</td>
<td>16/6/9/6</td>
</tr>
<tr>
<td>Boon</td>
<td>6</td>
<td>8</td>
<td>1024/344/216/20</td>
<td>512/224/144/24</td>
<td>16/8/8/9</td>
</tr>
<tr>
<td>Cohn3</td>
<td>8</td>
<td>161</td>
<td>1080/484/358/213</td>
<td>1080/484/358/213</td>
<td>1080/432/335/209</td>
</tr>
<tr>
<td>Butcher</td>
<td>8</td>
<td>7</td>
<td>4608/1361/605/24</td>
<td>4608/1113/813/22</td>
<td>1296/578/287/22</td>
</tr>
</tbody>
</table>

The first column in Table 1 indicates the name of each problem. The detailed description refers to [15,2,19]. The second and the third columns list the number of variables and the number of isolated solutions, respectively. The column with title “No_reduced” gives the results without precondition; that with title “PHC” and “IGB” present the corresponding results with PHCpack and the incomplete Gröbner basis. There are four numbers in the columns of No_reduced, PHC and IGB, which respectively list:

- total degree,
- mutli-homogeneous Bezout number [12,20,8,7],
- general linear-product Bezout number [9,10],
- mixed volume [8,5,19].

The smaller the numbers, the better the performance, since the tighter bound for the isolated solutions are obtained. Results in Table 1 clearly shows that the method by IGB is at least as good as that of PHC. Sometimes, for example the problems gaukwa, Boon and heart, IGB gives much better results than PHC.

5. Conclusions and discussion

The reduction algorithm based on the incomplete Gröbner basis provides an efficient preconditioner for the homotopy continuation methods in solving polynomial systems. It transforms a deficient polynomial system into a polynomial system having the same amount of finite complex solutions in polynomial time. The concept of $S$-polynomial plays an important role in the construction process of the incomplete Gröbner bases. The resulting system has a lower degree (in not only the total degree or the so called classical Bezout number, but also the mutli-homogeneous Bezout number and the mixed volume), thus can be solved faster by the continuation methods.

Nonlinear reductions are used extensively as preconditioners in mathematical software for polynomial systems, such as HOMpack [12] and HPCpack [19]. It is always an important issue to balance the costs spent on the precondition process and the resulting effects. Both of the nonlinear reductions in HPCpack and the method proposed in this paper rely on the computation of $S$-polynomials. By using the incomplete Gröbner basis in this paper, it should be simpler and quicker. The examples listed in the Table 1 can all be solved in several seconds by the algorithm with IGB implemented on MAPLE.

The nonlinear reduction algorithm by the incomplete Gröbner Bases is still at the very primary stage. To make it a more sophisticated preconditioner of the continuation methods for polynomial systems, much work needs to be done. For example, the mixed volume normally produces tight bound in homotopy continuation methods for polynomial systems of equations [8]. Preserving or even reducing the sparsity of polynomial systems would be a very important issue there.

Acknowledgments

The authors thank the anonymous referees for their helpful suggestions and comments. The work was supported by the National Science Foundation of China 10501030.

References