Hybrid Method for Solving Polynomial Equations

L.H. Zhi Y. Notake H. Kai M.-T. Noda
Department of Computer Science
Ehime University Japan
{lzhi,notake,kai,noda }@hpc.cs.ehime-u.ac.jp
K.I. Shiraishi
Department of Control Engineering
Takuma National College of Technology
siraisi@dc.takuma-ct.ac.jp

Abstract

We discuss how to decompose the zero set of a multivariate polynomial system with inexact coefficients to a sequence of zero sets of reduced triangular sets in a numerically stable way.

1 Introduction

Finding the solutions to a system of non-linear polynomial equations over a given field is a classical and fundamental problem in the computational literature. Many problems in robotics, computer vision, computational geometry, signal processing involve solving polynomial systems of equations. A number of symbolic, numeric and hybrid approaches have been proposed. Newton's algorithms and homotopy methods are two main numeric approaches for solving zero-dimensional polynomial systems. Newton's method works well only if we are given good initial guesses to the solutions and it is difficult for most practical problems. Since 1970's, the rapid advances in techniques for homotopy method have brought a great leap in the feasibility of solving numerical polynomial systems globally [8] [11]. However, it still suffers some problems such as path-crossing[12].

Most papers on symbolic or hybrid methods(combination of symbolic and numeric approaches) for polynomial solving concentrate on Gröbner basis and resultant method. It is well known that Gröbner basis method can not be applied safely with floating point arithmetic and requires to increase the precision of computation dramatically compared with input and output precision. Algorithm based on resultant method provides one of the most efficient solution method for small and medium-size zero-dimensional polynomial systems [3][18]. Different kinds of resultant matrices are used for constructing monomial bases, multiplication maps and, ultimately, reduce solving a polynomial system to an eigenvalue problem. On the other hand, as directly applying resultant for polynomial solving, Wu Wen-tsün developed the theory of subresultant for reducing a polynomial system to a family of triangular sets. The subresultant polynomial remainder sequence is well known as the best non-modular algorithm for computing GCD and resultant of sparse multivariate polynomials [1][2][4][5]. However, its application in polynomial solving is still relatively unexplored. In [13], Noda and Sasaki have used subresultant theory for computing approximate GCD of multivariate polynomials and then, applied it to solve ill-condition polynomial systems. But their purpose is to divide out the approximate GCD and transfer the system to well-condition problem.

In this paper, we combine Wu's symbolic elimination theory with Noda and Sasaki's approximate GCD computation to solve systems of polynomial equations with numeric coefficients. Our paper is organized as follows. In section 2 we describe Wu's method, followed in section 3 by generalizing it to polynomials with numerical coefficients. Section 4 compares the current approach with the Gröbner basis method.

2 Wu's Elimination Theory

2.1 Preliminaries

- Let K be a field of characteristic 0 and let $x_1, \ldots, x_n$ be a set of indeterminates with the order: $x_1 \prec x_2 \ldots \prec x_n$. $\mathsf{K}[\mathsf{x_1}, \ldots, \mathsf{x_n}]$ is the ring of polynomials in these variables.

- Let c be the greatest subscript such that x_c actually occurs in f. We define:

    1. $\text{cls}(f)$ = the *class* of f = c.
    2. $\text{lv}(f)$ = the *leading variable* of $f = x_c$.
    3. $\text{cdeg}(f)$ = the *class degree* of $f = \deg_{x_c} f$.
    4. $\text{ini}(f)$ = the *initial* of f with respect to $\text{lv}(f) = \text{coeff}(f, x_c, \text{cdeg}(f)$. Note that $\text{ini}(f)$ is a polynomial in $\mathsf{K}[\mathsf{x_1}, \ldots, \mathsf{x_{c-1}}]$.

- A polynomial g is said to be *reduced* with respect to f if $\deg_{x_c}(g) < cdeg(f)$.

- Let $PS = \{p_1, p_2, \ldots, p_s\}$ be a polynomial set in $\mathsf{K}[\mathsf{x_1}, \ldots, \mathsf{x_n}]$, PS is called a *triangular set* if either $s = 1$ and $p_1 \neq 0$, or $s > 1$ and $\text{cls}(p_1) < \text{cls}(p_2) < \cdots < \text{cls}(p_s)$. If $s > 1$ and p_j is reduced with respect to p_i for each pair $j > i$, then PS is called an *ascending set*. An ascending set is said to be *contradictory* if $s = 1$ and p_1 is a non-zero constant.

- For a non-empty polynomial set $PS \in \mathsf{K}[\mathsf{x}_1, \ldots, \mathsf{x}_\mathsf{n}]$, the greatest class c, if it exists, for which the number of corresponding polynomial is > 1, is called the *dominant class* of PS, the least degree of polynomials having class c is called the *dominant degree* of PS. In case, no such $c > 0$ exists then dominant class will be defined to be 0, while dominant degree will be left undefined.

- For polynomial sets PS and polynomial G. Zero(PS) denotes the zero set of PS, Zero(PS/G) for Zero(PS) − Zero(G).

2.2 Subresultant Chain

Let f and g be two multivariate polynomials in $\mathsf{K}[\mathsf{x}_1, \ldots, \mathsf{x}_\mathsf{n}]$. Suppose $\mathrm{lv}(f) = \mathrm{lv}(g) = x$ and $m = \mathrm{cdeg}(f) \geq \mathrm{cdeg}(g) = n$:

$$\begin{aligned} f &= f_m x^m + \cdots + f_0, \ f_m \neq 0. && (1) \\ g &= g_n x^n + \cdots + g_0, \ g_n \neq 0. && (2) \end{aligned}$$

According to [2][4], the subresultant chain is defined as

$$S_j(x) = \begin{vmatrix} f_m & f_{m-1} & & \cdots & \cdots & f_{2j-n+2} & x^{n-j-1}f \\ & \ddots & & & & \vdots & \vdots \\ & & f_m & f_{m-1} & \cdots & f_{j+1} & x^0 f \\ g_n & f_{n-1} & & \cdots & \cdots & g_{2j-m+2} & x^{m-j-1}f \\ & \ddots & & & & \vdots & \vdots \\ & & g_n & g_{n-1} & \cdots & g_{j+1} & x^0 g \end{vmatrix}, \quad (3)$$

where $f_k = g_k = 0$ if $k < 0$. Therefore

$$S_j(x) = U_j(x)\, f(x) + V_j(x)\, g(x).$$

where U_j is S_j except for the last column, which is top down

$$x^{n-j-1} \cdots 1 \ 0 \cdots 0$$

and V_j is S_j except for the last column, which is top down

$$0 \cdots 0 \ x^{m-j-1} \cdots 1,$$

hence, $\deg_x U_j \leq n - j - 1$ and $\deg_x V_j \leq m - j - 1$. It is clear that

$$\mathrm{Zero}(\{f, g\}) \subset \mathrm{Zero}(S_j), \ \text{for}\, 0 \leq j \leq n - 1.$$

Proposition 1 *The last subresultant S_{n-1} is equal up to a sign to the pseudo-remainder of f with respect to g, i.e., for some polynomial q,*

$$S_{n-1} = \pm\mathrm{prem}(f, g) = \pm g_n^{m-n+1}\, f + q\, g. \quad (4)$$

Proposition 2 *S_0 is the resultant of f and g, and the vanishing of S_0 is the necessary and sufficient condition for f and g to have a GCD of positive degree in x.*

Proposition 3 *If f and g have a non-trivial GCD of degree $d > 0$, then $S_j = 0$ for $0 \leq j < d$, and $\mathrm{GCD}(f, g)$ is equal to the primitive part of the first non-zero polynomial S_d.*

In the case $S_0 \neq 0$, the least integer e, if it exists, for which S_e has a positive degree in x, will be called *exponent* of f and g. The corresponding polynomial S_e will then be called the *eliminant* of f and g.

Wu's elimination method consists the following four replacement rules[19]. Here, we suppose PS is a non-empty polynomial set.

Rule 1. For any polynomial $p \in PS$, if $p = p_1 \cdot p_2$. We replace PS by polynomial sets PS_1 and PS_2 consisting of same polynomials as PS with p replaced by p_1 and p_2 respectively.

Rule 2. Suppose the dominant class of PS is $c > 0$. Let f be the polynomial with class c and $\mathrm{cdeg}(f) = d$ the dominant degree, g be any other polynomial in PS with $\mathrm{cls}(g) = c$, S_0 the resultant of f and g with respect to variable x_c. Replace PS by PS_1 consisting of same polynomials as PS but with f and g replaced according to the following rules.

2.1 If $S_0 = 0$ then replace f and g by S_d, where $d = \deg_{x_c} \mathrm{GCD}(f, g)$.

2.2 If $S_0 \neq 0$ and S_e be the eliminant of f and g, then replace f and g by S_0 and S_e.

2.3 If $S_0 \neq 0$ and the eliminant is non-existent, then replace f and g by f and S_0.

Applying Rule 2 to PS_1 again, until the dominant class is 0. We get a triangular set TS.

Rule 3. If the initial of some polynomial f_{i+1} in TS is not reduced with respect to the partial triangulated set TS_i, formed of polynomials in TS preceding f_{i+1}. Compute the pseudo-remainder r of f_{i+1} with respect to TS_i.

3.1 If $\mathrm{cls}(r) = \mathrm{cls}(f_{i+1})$, replace TS by TS' consisting of same polynomials as TS but with f_{i+1} replaced by r.

3.2 If $\mathrm{cls}(r) < \mathrm{cls}(f_{i+1})$, then apply replacement rules to $\{\{r\} \bigcup TS_i\}$ to find an ascending set AS_i. If AS_i is contradictory, then the zero set of PS is empty; otherwise, apply the replacement rules to PS again over the algebraic extension field generated by AS_i.

Rule 4. For each ascending set AS obtained by preceding rules applied to PS, compute the pseudo-remainder set RS of polynomials in PS with respect to AS, replace AS by $PS' = AS \bigcup RS$. Apply rules to PS' until the pseudo-remainder sets of PS with respect to AS are empty.

Applying replacement rule 1-4 whenever possible. Ultimately, we have the following theorem.

Zero Decomposition Theorem[19] *There is an algorithm so that for any polynomial set PS there will be a decomposition of the form*

$$\text{Zero}(PS) = \sum_k \text{Zero}(AS_k/J_k), \tag{5}$$

in which each AS_k is an ascending set while J_k is the product of all initials of polynomials in AS_k.

Example 1 $PS = \{f_1(x,y,z), f_2(x,y,z), f_3(x,y,z)\}$ *with* $x \prec y \prec z$ *and*

$$\begin{aligned} f_1 &= x^2 - xy + y^2 - 1, \\ f_2 &= 2xy + yz - 3z^2, \\ f_3 &= yz + x^2 - 2z^2. \end{aligned}$$

Step 1. Classify the polynomials in PS into two polynomial sets.

$$PS = [[f_1], [f_2, f_3]]$$

Step 2. Compute the subresultant chain of f_2, f_3, we get

$$\begin{aligned} S_1 &= yz - 4xy + 3x^2, \\ S_0 &= 17x^2y^2 - 2y^3x - 24yx^3 + 9x^4. \end{aligned}$$

Step 3. Since $S_0 \neq 0$ and $\deg_z(S_1) = 1 > 0$, by Rule 2.2, replace f_2, f_3 by S_0, S_1. Let

$$PS_1 = [[f_1, S_0], [S_1]],$$

Step 4. Compute the subresultant chain of f_1, S_0,

$$\begin{aligned} S_1' &= -2xy + 7yx^3 - 6x^4 + 15x^2, \\ S_0' &= 127x^8 - 294x^6 + 171x^4 - 4x^2. \end{aligned}$$

By Rule 2.2, replace f_1, S_0 by S_0', S_1'. Let

$$TS_1 = [[S_0'], [S_1'], [S_1]].$$

Step 5. Form the pseudo-remainder of S_1 with respect to ascending set $[S_0', S_1']$,

$$r = 6zx^4 - 15x^2z - 45x^5 + 54x^3.$$

By Rule 3, replace S_1 by r,

$$TS_1 = [S_0', S_1', r].$$

Step 6. The pseudo-remainder set of PS with respect to TS_1 is empty. Then

$$\text{Zero}(PS) = \text{Zero}(TS_1/(I_1 I_2)) + \sum_{i=1}^{2} \text{Zero}(PS + TS_1 + I_i),$$

where

$$\begin{aligned} I_1 &= \text{ini}(S_1') = -7x^3 - 2x = -x(7x^2+2) = -p_1 \cdot p_2, \\ I_2 &= \text{ini}(r) = 6x^4 - 15x^2 = 3x^2(2x^2-5) = 3p_1^2 \cdot p_3. \end{aligned}$$

Step 7. Apply Rule 1 to the factors of I_1 and I_2, we have

$$\sum_{i=1}^{2} \text{Zero}(PS + TS_1 + I_i) = \sum_{i=1}^{3} \text{Zero}(PS + TS_1 + \{p_i\}).$$

Step 8. For $i = 2, 3$, it is easy to check the zero sets are empty. For $i = 1$, repeat the preceding steps, we will get

$$\text{Zero}(PS + TS_1 + \{p_1\}) = \text{Zero}([x, y^2 - 1, yz]).$$

Finally,

$$\text{Zero}(PS) = \text{Zero}(TS_1/(I_1 I_2)) + \text{Zero}([x, y^2 - 1, yz]).$$

Now, it is easy to get all eight solutions of PS as

x	1	-1	-0.156	0.156	1.13	-1.13	0	0
y	1	-1	-1.06	1.06	0.747	-0.747	1	-1
z	1	-1	-0.556	0.556	-0.637	0.637	0	0

3 Polynomial System Solving

We consider $PS = \{p_1, \ldots, p_s\}$ be a polynomial set with $p_i \in \mathsf{C}[\mathsf{x}_1, \ldots, \mathsf{x}_n]$ whose coefficients have specified *numerical* values. Unlike most papers on polynomial solving, here we do not assume $s = n$ or the zero set of PS be zero-dimension. Our meaning of finding the common zeros of PS is to decompose the zero set as (5). If all coefficients in PS be assumed to be exact as rational numbers, implement Wu's elimination method in exact arithmetic, we could decompose the zero set of PS as in section 2. Otherwise, some of the coefficients in PS are only known to a specified level of accuracy. Then PS represents an *equivalent class* $\overline{PS}$ of polynomial sets $\widetilde{PS}$ and the members of $\overline{PS}$ cannot be distinguished in the given context. Thus, according to [15], the concept of a zero has to be widened to :

$$z \in \mathsf{C}^n \text{ is a pseudozero of } PS \Longleftrightarrow \exists \widetilde{PS} \in \overline{PS}, \quad |p_i(z)| \leq \epsilon, \quad p_i \in \widetilde{PS}.$$

for a specified small number $\epsilon > 0$. In the following, we define $\| p \|$ be the ∞-norm of the coefficient vector. We show how to stabilize Wu's elimination method in finite precision arithmetic.

3.1 Univariate Case

Let $PS = \{p_1, \ldots, p_s\}$ with $p_i \in \mathsf{C}[x]$. The zero decomposition in (5) is actually:

$$\mathrm{Zero}(PS) = \mathrm{Zero}(\mathrm{GCD}(p_1, \ldots, p_s)).$$

There are a lot of algorithms available for computing the GCD of univariate polynomials with inexactly known coefficients. [6][7][9][10][13][14]. Noda and Sasaki's scaled Euclidean algorithm is simple, efficient and stable. But it can produce answers slightly different than what we want. In the following, we present a new algorithm that modified Noda and Sasaki's method to avoid unsatisfactory results.

Algorithm A *(Approximate GCD of two univariate polynomials with accuracy ϵ).* Given nonzero polynomials f and g in $\mathsf{C}[x]$ with accuracy ϵ and $\deg_x(f) \geq \deg_x(g)$, this algorithm calculates an approximate GCD of f and g with accuracy ϵ.

A1. [Initialize] Set $p_1 \leftarrow f$, $p_2 \leftarrow g$.

A2. [Iteration] Compute the remainder r and quotient q of p_1 and p_2.

A3. [Finished?] If $\| r \| \geq \epsilon$, set $p_1 \leftarrow p_2$, $p_2 \leftarrow r/\max(1, \| q \|)$. Go back to A2.

Otherwise, compute the remainder r and quotient q of g and p_2.

If $\| r \| \geq \epsilon$ then set $p_1 \leftarrow g$, $p_2 \leftarrow r/\max(1, \| q \|)$. Go back to A2.

Otherwise, compute the remainder r and quotient q of f and p_2.

If $\| r \| \geq \epsilon$ then set $p_1 \leftarrow f$, $p_2 \leftarrow r/\max(1, \| q \|)$. Go back to A2.

Otherwise, the algorithm terminates, return $p_2/\mathrm{ini}(p_2)$. ■

Example 2

$$\begin{aligned} f &= 3.x^7 - 1.x + 3.x^6 - 1., \\ g &= x^5 + 4.x + 1.00001x^4 + 4.00004. \end{aligned}$$

Suppose $\epsilon = 10^{-4}$.

Numbering intermediate remainder in A2 of Algorithm A properly, we obtain a sequence of polynomials

$$\begin{aligned} p_3 &= -.333333x - .333373 - 4.00000x^3 - 4.00000x^2 - .100002 \cdot 10^{-4}x^4, \\ p_4 &= .333332x + .333373 + 4.00000x^2 + 3.99999x^3, \\ p_5 &= .250001 \cdot 10^{-9}x - .625003 \cdot 10^{-9}x^2, \end{aligned}$$

Since $\| p_5 \| \leq \epsilon$, compute the remainder and quotient of g and p_4:

$$\begin{aligned} r &= 4.00694x + 4.00698 - .295158 \cdot 10^{-5}x^2, \\ q &= .250000x^2 + .192706 \cdot 10^{-5}x - .208353 \cdot 10^{-1}. \end{aligned}$$

Since $\| r \| > \epsilon$, replace p_1 by g and p_2 by r. Repeat A2, we obtain

$$\begin{aligned} p_3 &= 4.00694x + 4.00698, \\ p_4 &= 0.186850 \cdot 10^{-8}. \end{aligned}$$

Check the termination, we will find the approximate GCD of f and g with accuracy 10^{-4} is $p_3/\mathrm{lcoeff}(p_3) = x + 1.00001$. We remark that algorithm in [13] stops after $\| p_5 \| \leq \epsilon$, and returns a degree-3 GCD which is completely spurious.

The normalization of the remainder is crucial in the algorithm. The analysis of numerical stability of the algorithm is similar to [13].

3.2 Multivariate Case

Let $PS = \{p_1, \ldots, p_s\}$ be a polynomial set with $p_i \in \mathsf{C}[\mathsf{x}_1, \ldots, \mathsf{x}_n]$. We can use (3) to compute the subresultant chain to find the pseudo-remainder, eliminant, resultant and GCD. But compute the determinant of a polynomial matrix is not easy. Actually, we have the following more efficient algorithm which modified [2][4] to numerical case.

Algorithm S *(Approximate subresultant polynomial remainder sequence of two multivariate polynomials with accuracy* ϵ*).* Given nonzero polynomials f and g in $[x_1, \ldots, x_n]$ with accuracy ϵ, $\mathrm{lv}(f) = \mathrm{lv}(g)$ and $\mathrm{cdeg}(f) \geq \mathrm{cdeg}(g)$, this algorithm calculates an approximate subresultant polynomial remainder sequence of f and g with accuracy ϵ.

S1. [Initialize] Set $L \leftarrow [g, f]$, $p_1 \leftarrow f$, $p_2 \leftarrow g$, $\gamma \leftarrow 1$, $\beta \leftarrow 1$, $i \leftarrow 3$.

S2. [Iteration] Set $d \leftarrow \mathrm{cdeg}(p_{i-2}) - \mathrm{cdeg}(p_{i-1})$, $r \leftarrow \mathrm{nprem}(p_{i-2}, p_{i-1})$.

If $\| r \| \leq \epsilon$ then go to S3.

Otherwise, set $p_i \leftarrow \text{nquo}(r, \text{normal}(\beta \cdot \gamma^d)))$, $L \leftarrow \text{CONS}(p_i, L)$,

$\beta \leftarrow \text{ini}(p_{i-1})$, $\gamma \leftarrow \gamma^{1-d}\beta^d$, $i \leftarrow i+1$.

S3. [Finished?] If $\| r \| \leq \epsilon$ or $\deg_x(r) = 0$ then set $L \leftarrow \text{INV}(L)$, return L.

Otherwise, go back to S2. ■

The function $\text{CONS}(p_i, L)$ appends p_i to the list L and INV(L) reverses the list L. Note that the division to get p_i in S2 is exact if the coefficients are exact rational numbers. Otherwise, we impose the similar normalization of quotient as in the case of univariate polynomials. If $\text{cls}(g) = 0$ then $\text{nquo}(f, g) = f/g$. Otherwise, suppose the pseudo-remainder r and quotient q of f and g with respect to $x = \text{lv}(g)$ be calculated by

$$r = \text{ini}(g)^d\, f - q\, g,\; d = \deg_x(f) - cdeg(g) + 1 > 0. \tag{6}$$

If $\| r \| \,/\, \| \text{ini}(g)^d \| \leq \epsilon$ then

$$\text{nquo}(f, g) = \text{nquo}\left(q, \text{ini}(g)^d\right).$$

Otherwise, return $f/\| g \|$ as the quotient. Since the class of divisor decreases, finally, we can stop to get a polynomial divided by a number. Let q, d be the same in (6), the normalizations of the pseudo-remainders and polynomials are

$$\begin{aligned} \text{nprem}(f, g) &= r/max(\| \text{ini}(g)^d \|, \| q \|), \\ \text{normal}(f) &= f/\| f \|. \end{aligned}$$

See [13] for the analysis of numerical stability of the algorithm.

Example 3 *Suppose* $\epsilon = 10^{-5}$,

$$\begin{aligned} p_1 &= 2y^5 + xy^4 + x^2y + 2x + 2xy^2 + 4y + y^4 + xy + 2, \\ p_2 &= 6.y^3x + 6.y^3 + x^4 + 3.x^2y^2 + 6.xy^2 + 1.00001x \\ &\quad +2.yx^3 + 2.00002y + x^3 + 3.y^2 + 1.00001. \end{aligned}$$

The subresultant polynomial remainder sequence(up to sign) of p_1 and p_2 computed by the above algorithm is

$$\begin{aligned} p_3 &= 432.x^4y^2 + 1296.x^2y^2 + 1296.x^3y^2 + 432.xy^2 + 216.x^5y + 2256.00x^3y \\ &\quad +2856.00xy + 48.x^7y + 3456.x^2y + 912.000y + 48.x^6y + 960.000x^4y \\ &\quad +1776.00x + 24.x^8 + 1776.00x^3 + 456.000 + 2616.00x^2 + 24.x^6 \end{aligned}$$

$$
\begin{aligned}
& +528.000x^4 + 48.0004x^5 + 48.x^7; \\
p_4 &= 21696.0x^8y + 173632.x^5y + 16533.3x^9y + 64.x^{13}y + 30912.1x^7y + 7701.34y \\
& +52288.0xy + 153472.x^2y + 254698.x^3y + 1600.00x^{11}y + 106.667x^{12}y \\
& +64.x^{14}y + 7552.00x^{10}y + 262336.x^4y + 76821.5x^6y + 21.3333x^{15}y \\
& +29994.7x + 19114.6x^9 + 853.335x^{12} + 102880.x^2 + 258517.x^4 \\
& +12042.6x^{10} + 53866.8x^7 + 125226.x^6 + 204085.x^3 + 3850.67 \\
& +217984.x^5 + 4576.00x^{11} + 42.6667x^{15} + 26304.0x^8 \\
& +64.x^{14} + 85.3337x^{13} + 10.6667x^{16},
\end{aligned}
$$

and $\| p_5 \| \leq \epsilon$. Actually, apply approximate GCD to the coefficients of p_4, we will find the primitive part of p_4 is

$$\text{primitive}(p_4) = 2.y + 1.x + 1.$$

4 Experimental Test

We report here on the results of our algorithms applied to two examples. The algorithms are implemented in Maple V.

Example 4 *This example is cited in [16]. Consider two ellipses which intersect with angles not far from* $90°$ *in four well-separated real points. The associated quadratic equations in* x, y *have real rational coefficients with nontrivial denominators and numerators.* p_1 *and* p_2 *are their decimal approximations to seven digits.*

$$
\begin{aligned}
p_1 &= 1.027748y^2 - .467871xy + 2.972252x^2 + .662026y + 0.0785252x - 3.888889, \\
p_2 &= 3.958378y^2 + .701807xy + 1.041622x^2 - 0.0785252y + .662026x - 3.888889.
\end{aligned}
$$

With lexicographic term order, $x \prec y$, the exact rational Gröbner basis of this system is (displayed to 7-digits)

$$
\begin{aligned}
g_1(x) &= x^4 - 0.134646x^3 - 2.107266x^2 + 0.242335x + 1.009172, \\
g_2(x, y) &= y - 1.355154 \cdot 10^{16}x^3 - 1.240075 \cdot 10^{16}x^2 + 1.553930 \cdot 10^{16}x + 1.302800 \cdot 10^{16}.
\end{aligned}
$$

It has been pointed in [16], if we compute the solutions of g_1 to accuracy less than 34 digits, there are no meaningful results for two $y-$components. By our methods, suppose $e = 10^{-5}$, we get the zero decomposition of p_1, p_2 as

$$\text{Zero}(\{p_1, p_2\}) = \text{Zero}(\{g_1, g_2\}/I_1) + \text{Zero}(\{f_1, f_2\}).$$

Where I_1 is the initial of g_2 and

$$
\begin{aligned}
g_1 &= 120.9999x^4 - 16.29212x^3 - 254.9791x^2 + 29.32250x + 122.1098, \\
g_2 &= -2.573291xy + 10.69477x^2 + 2.701253y - .3695634x - 11.39689.
\end{aligned}
$$

Solve g_1 for Digits $= 10$(the number of digits carried in floats),

$$x = -1.204415,\ -.7603909,\ 1.049726,\ 1.049726.$$

Substitute the first two zeros to g_2, the initial I_1 is nonzero, and we get

$$y = -.7865145,\ 1.058881.$$

which are exact to six digits. Evaluate I_1 at the last two zeros, we find it is less than 10^{-5}. Now, we consider another branch

$$\begin{aligned} f_1 &= 1.000000x - 1.049727, \\ f_2 &= 2.254914y^2 + .3749366y - 1.165602. \end{aligned}$$

There are two sets of solutions

$$\{x = 1.049727,\ y = -.8068975\},$$

$$\{x = 1.049727,\ y = .6406222\}.$$

Substitute the solutions to p_1, p_2, the error is less than 10^{-5}.

For this example, using Maple's fsolve, it only gives one set of solutions corresponding to $\{x = 1.049727,\ y = -.8068975\}$. In order to find the other three roots, we have to give appropriate range informations.

Example 5 *This example appeared in [20].*

$$\begin{aligned} p_1 &= ty^8 + y^3x + 3, \\ p_2 &= 4x^2 + 3xy + y^2 + 2. \end{aligned}$$

Suppose $t = 10^{-4}$ be a small number. The Gröbner basis with lexicographic term order $x \prec y$ is

$$\begin{aligned} g_1 &= 4096x^{16} + 16384x^{14} + 2308672x^{12} + 4648672x^{10} + 401969795x^8 \\ &\quad +600322168x^6 + 467731792x^4 + 385520256x^2 + 56310016, \\ g_2 &= 8349641086351584263053068672y + 4264054383431211693884392499 2x \\ &\quad +5290596276288978561901723107 9x^7 + 64447251228171657084673721132x^5 \\ &\quad +3381397706202098643128488715 2x^3 + 52887302028880293063468041 6x^9 \\ &\quad +30437897598314026164343737 6x^{11} + 20141150395669510415319 04x^{13} \\ &\quad +541204990029293392547840x^{15}. \end{aligned}$$

It is obvious that we have to compute the roots of g_1 to high accuracy to get reasonable solutions of y due to the large coefficients in g_2. For Digits $= 10$, the error of some solutions are about 1. Compute the zero decomposition by our subresultant method, we get

$$\mathrm{Zero}(\{p_1, p_2\}) = \mathrm{Zero}(\{f_1, f_2\}),$$

where f_1 is the same as g_1 and f_2 is

$$f_2 = (49496x^3 - 19904x + 93x^7 - 252x^5)y + 30016 + 364x^8 + 350x^6 + 119412x^4 + 59696x^2.$$

Substitute the solutions of g_1 to f_2, we get the solutions of $y-$component which are exact to five digits, i.e., the error is less than 10^{-5}. It has been pointed in [16], large coefficients originate through S-polynomial formation or reduction of a polynomial with a small leading coefficient and some other coefficients with a modulus of order 1, combined with another polynomial whose matching coefficient is of order 1. On the contrary, for subresultant chain, small leading coefficient does not cause large coefficients. It can be seen from the above example. In fact, we have the following proposition.

Proposition 4 *Let f and g be two multivariate polynomials in $\mathsf{K}[\mathsf{x_1}, \ldots, \mathsf{x_r}]$. Suppose* $\mathrm{lv}(f) = \mathrm{lv}(g) = x$ *and* $m = \mathrm{cdeg}(f), \mathrm{cdeg}(g) = n$,

$$f = f_m x^m + \cdots + f_0, \quad (7)$$
$$g = g_n x^n + \cdots + g_0. \quad (8)$$

If $f_m = 0$, $g_n \neq 0$, then consider

$$\overline{f} = f_{m-1}x^{m-1} + \cdots + f_0.$$

We have

$$S_j(\overline{f}, g) = \pm S_j(f, g)/b_n, \text{ for } j < \min(m-1, n).$$

Similarly, if $f_m \neq 0$ and $g_n = 0$, then consider g as $\overline{g}$ of degree $n-1$, we have

$$S_j(f, \overline{g}) = \pm S_j(f, g)/a_m, \text{ for } j < \min(m, n-1).$$

5 Conclusion

Polynomial equations used to describe practical problems usually have a limited meaningful accuracy. For a well-condition system, a small uncertainty in its data must not imply large uncertainties of its solutions. Gröbner basis is not suitable for this purpose [16]. Our algorithm is more stable due to the special properties of subresultant chain. Meanwhile, we also notice that the algorithms based on symbolic elimination and finding roots of a single polynomial have to be implement in high-precision arithmetic. It has been shown by Wilkinson[17] that the problem of finding roots of a univariate polynomial may be ill-conditioned for high degree polynomials. However, high-precision arithmetic will slow down the overall computation significantly. So we start with low accuracy and add the precision digits in the case the algorithms fail. More examples and analysis will appear in our forthcoming paper.

References

[1] Brown, W. S., *On Euclid's algorithm and the computation of polynomial greatest common divisors*, J. ACM **18,4** (1971), 478–504.

[2] Brown, W. S., *The subresultant PRS algorithm*, ACM Trans. Math. Software **4** (1978), 478–504.

[3] Canny, J., Emiris, I., *An efficient algorithm for the sparse mixed resultant,* in Proc. Intern. Symp. on Applied Algebra, Algebraic Algor. and Error-Corr., 1993, 89–104.

[4] Collins, G. E., Subresultants and reduced polynomial remainder sequences, J. ACM **14** (1967), 128–142.

[5] Collins, G. E., *The calculation of multivariate polynomial resultants,* J. ACM **19** (1971), 515–532.

[6] Corless, R. M., Gianni, P. M., Trager, B. M. and Watt, S. M.,*The singular value decomposition for polynomial Systems,* in Proc. Internat. Symp. on Symbolic and Algebraic Comput, ACM Press, New York, 1995, 195-207.

[7] Chin, P., Corless, R. M., Corliss, G. F., *Optimization strategies for the approximate GCD problem,*, in Proc. Internat. Symp. on Symbolic and Algebraic Comput, ACM Press, New York, 1998, 228–235.

[8] Garcia, C., Zangwill, W., *Finding all solutions to polynomial systems and other systems of equations,* Math. Prog. **16** (1979),159–176.

[9] Hribernig, V. , Stetter, H. J., *Detection and validation of clusters of polynomial zeros,* J. Sysmb. Comp. **24** (1997), 667–681.

[10] Karmarkar, N., Lakshman Y. N., *Approximate polynomial greatest common divisors and nearest singular polynomials,* in Proc. Internat. Symp. on Symbolic and Algebraic Comput, ACM Press, New York, 1996, 35–39.

[11] Li, T. Y., Sauer,T. and Yorke, J. A., *The cheater's homotopy: an efficient procedure for solving systems of polynomial equations,* SIAM J. Numer. Anal. /bf 26 (1989), 1241-1251.

[12] Morgan, A., *Polynomial continuation and its relationship to the symbolic reduction of polynomial systems,* Symbolic and Numerical Computation for Artificial Intelligence, Academic Press, Orlando, FL, 1992, 23–45.

[13] Noda, M.-T., Sasaki, T., *Approximate GCD and its application to ill-conditioned algebraic equations,* J. Comput. Appl. Math. **38**(1991), 335–351.

[14] Schonhage, A., *Quasi-GCD computations,* J. Complexity **1** (1985), 118–137.

[15] Stetter, H. J., *Analysis of Zero Clusters in Multivariate Polynomial Systems*, in Proc. Internat. Symp. on Symbolic and Algebraic Comput, ACM Press, New York, 1996, pp127–135.

[16] Stetter, H. J., *Stabilization of polynomial systems solving with Gröbner bases*, in Proc. Internat. Symp. on Symbolic and Algebraic Comput, ACM Press, New York, 1997, 117–124.

[17] Wilkinson, J., *The evaluation of the zeros of ill-conditioned polynomials,part 1 and 2,* Numerische Mathematik **1** (1959), 150–166 and 167–180.

[18] Wallack, A., Emiris, I., Manocha, D., *MARS: A Maple/Matlab/C Resultant-based solver*, in Proc. Internat. Symp. on Symbolic and Algebraic Comput, ACM Press, New York, 1998, 244–251.

[19] Wu, W. T., *On a linear equations method of non-linear polynomial equations-solving,* MM-Res, Preprints, **6** (1991), 23–36.

[20] Zhi, L. H., *polynomial factorization over algebraic fields and its applications,* Ph.D thesis, Academia Sinica, China (1996).