An effective algorithm for isolating the real solutions of semi-algebraic systems and its applications *

Bican Xia[1]
School of Mathematical Sciences
Peking University, Beijing 100871, China
Lu Yang[2]
Chengdu Institute of Computer Applications
Chinese Academy of Sciences, Chengdu 610041, China

Abstract. We present an effective algorithm for isolating the real solutions of semi-algebraic systems, which has been implemented in MAPLE-program `realzero`. For a large number of examples with various backgrounds, `realzero` gets the solutions very efficiently.

Key Words: semi-algebraic system, real solution isolation, resultant, Uspensky algorithm

1 Introduction

By semi-algebraic systems [2, 5], we mean systems of polynomial equations and inequalities. Strictly speaking, we call

$$\begin{cases} p_1(x_1,x_2,...,x_s)=0,\\ p_2(x_1,x_2,...,x_s)=0,\\ \cdots\cdots,\\ p_n(x_1,x_2,...,x_s)=0,\\ q_1(x_1,x_2,...,x_s)\ge 0,...,q_r(x_1,x_2,...,x_s)\ge 0,\\ g_1(x_1,x_2,...,x_s)>0,...,g_t(x_1,x_2,...,x_s)>0,\\ h_1(x_1,x_2,...,x_s)\ne 0,...,h_m(x_1,x_2,...,x_s)\ne 0, \end{cases}$$

a *semi-algebraic system*, where $p_i (1\le i\le n)$, $q_l (1\le l\le r)$, $g_j (1\le j\le t)$, $h_k (1\le k\le m)$ are all polynomials in $x_1,...,x_s$ with integer coefficients and we always assume that $\{p_1,...,p_n\}$ has a finite number of common zeros. For convenience, we call it SAS.

*This work is supported in part by Chinese National 973 Project (G1998030602) and the 95 Key Project on Fundamental Research of Academia Sinica.

[1]Corresponding author, e-mail: xbc@math.pku.edu.cn
[2]E-mail: cdluyang@public.sc.cninfo.net

Many problems in both practice and theory can be reduced to the problems of solving system SAS. For example, some special cases of "p-3-p" problem [8] which originates from computer vision, the problem of constructing limit cycles of plane polynomial systems [13, 16] and the problem of automated discovering and proving for geometric inequalities [22, 21]. Moreover, many problems in geometry, topology and differential dynamical systems are expected to be solved by translating them into solving certain semi-algebraic systems [2].

There are two classical algorithms for solving semi-algebraic systems, which are Tarski's method [15] and cylindrical algebraic decomposition method proposed by Collins [5]. The complexity of these two algorithms are super-exponential time and double-exponential time, respectively. Therefore, they are not effective in practice. In the past two decades, there have been some exciting results [9, 3, 2, 14] in theory. For example, Smale proved there exists a singly exponential time algorithm for semi-algebraic systems [2]. However, the problem is far from being solved. Efficient algorithms are highly needed to help solving related problems in those fields mentioned above.

In this paper, combining the algorithms for solving systems of polynomial equations such as Ritt-Wu method [20, 19] and WR algorithm [24] with Uspensky algorithm [6] for isolating real zeros of a univariate polynomial, we present an effective algorithm for isolating the real solutions of semi-algebraic systems which, in some sense, can be viewed as the generalization of Uspensky algorithm. Our algorithm has been found to be very efficient in practice on a large number of problems with various backgrounds though it is not a complete one in theory.

The paper is divided into four sections. Section 2 presents the kernel part of our algorithm while Section 3 shows the lifting part and recursive part of our algorithm. Section 4 includes some examples solved by our program `realzero` which implements the algorithm in MAPLE.

2 Basic Algorithm

In this paper, all the polynomials, if not specified, are in $Z[x_1, ..., x_s]$. For any polynomial P with positive degree, the *leading variable* x_l of P is the one with greatest index l that effectively appears in P. By a *triangular set*, we mean a set of polynomials $\{f_i(x_1, ...x_i), f_{i+1}(x_1, ...x_{i+1}), ..., f_l(x_1, ...x_l)\}$ in which the leading variable of f_j is x_j.

It is well known that for a system of polynomial equations with zero dimensional solutions, there exist many algorithms based on Ritt-Wu method, Gröbner basis method or subresultant method, which can decompose the given system into systems of triangular equations (see, for example, [19, 20, 4, 17, 18, 11, 1, 24]). Therefore, in Sections 2 and 3, we only consider triangular equations and the problem we discuss is to isolate the real solutions of following system TSA,

$$\begin{cases} f_1(x_1) = 0, \\ f_2(x_1, x_2) = 0, \\ \cdots\cdots, \\ f_s(x_1, x_2, ..., x_s) = 0, \\ g_1(x_1, x_2, ..., x_s) \geq 0, ..., g_r(x_1, x_2, ..., x_s) \geq 0, \\ g_{r+1}(x_1, x_2, ..., x_s) > 0, ..., g_t(x_1, x_2, ..., x_s) > 0, \\ h_1(x_1, x_2, ..., x_s) \neq 0, ..., h_m(x_1, x_2, ..., x_s) \neq 0, \end{cases}$$

where $\{f_1, f_2, \cdots, f_s\}$ is a normal ascending chain [24] (also see Definition 2.3 and Remark 2 in this section).

Definition 2.1 (Discriminant)

Given a polynomial $g(x)$, let resultant(g, g'_x, x) be the Sylvester resultant of g and g'_x with respect to (w.r.t.) x. We call it the *discriminant* of g w.r.t. x and denote it by Discrim(g, x) or simply by Discrim(g) if its meaning is clear.

It should be pointed out that the definition of discriminant here is little different from others which are the quotient of resultant(g, g'_x, x) by the leading coefficient of $g(x)$.

Definition 2.2 (Resultant and Pseudo-remainder w.r.t. a Triangular Set)

Given a polynomial g and a triangular set $\{f_1, f_2, ..., f_s\}$, let

$$r_s := g, \quad r_{s-i} := \text{resultant}(r_{s-i+1}, f_{s-i+1}, x_{s-i+1}), \quad i = 1, 2, ..., s;$$

$$q_s := g, \quad q_{s-i} := \text{prem}(q_{s-i+1}, f_{s-i+1}, x_{s-i+1}), \quad i = 1, 2, ..., s,$$

where resultant(p, q, x) means the Sylvester resultant of p, q w.r.t. x and prem(p, q, x) means the pseudo remainder of p divided by q w.r.t. x.

Let res$(g, f_s, ..., f_i)$ and prem$(g, f_s, ..., f_i)$ denote r_{i-1} and q_{i-1} $(1 \leq i \leq s)$, respectively, and call them the *resultant* and *pseudo-remainder* of g w.r.t. the triangular set $\{f_i, f_{i+1}, ..., f_s\}$, respectively.

Definition 2.3 [24] (Normal Ascending Chain)

Given a triangular set $\{f_1, f_2, ..., f_s\}$, by $I_i\,(i = 1, 2, ..., s)$ denote the leading coefficient of f_i in x_i. A triangular set $\{f_1, f_2, ..., f_s\}$ is called a *normal ascending chain* if

$$I_1 \neq 0, \quad \text{res}(I_i, f_{i-1}, ..., f_1) \neq 0, \quad i = 2, ..., s.$$

Definition 2.4 (Critical Polynomial of System TSA)

Given a TSA, called T. For every $f_i (i \geq 2)$, let

$$\begin{aligned} &Bf_2 = \text{Discrim}(f_2, x_2), \\ &Bf_i = \text{res}\,(\text{Discrim}(f_i, x_i), f_{i-1}, f_{i-2}, \cdots, f_2), i > 2. \end{aligned}$$

For $\forall q \in \{g_j (1 \leq j \leq t)\} \bigcup \{h_k (1 \leq k \leq m)\}$, let

$$Bq = \text{res}\,(q, f_s, f_{s-1}, \cdots, f_2).$$

We define

$$BP_T(x_1) = \prod_{2\le i\le s} Bf_i \cdot \prod_{1\le j\le t} Bg_j \cdot \prod_{1\le j\le m} Bh_k.$$

and call it the *critical polynomial* of the system T w.r.t. x_1. We also use BP or $BP(x_1)$ to denote $BP_T(x_1)$ if its meaning is clear.

Definition 2.5 (Regular TSA)
A TSA is *regular* if resultant$(BP(x_1), f_1(x_1), x_1) \neq 0$.

Remark 1 According to Definition 2.5, for a regular TSA, every $Bh_k(1 \le k \le m)$ has no common zeros with $f_1(x_1)$, which means every solution of $\{f_1 = 0, f_2 = 0, \cdots, f_s = 0\}$ always satisfies $h_k \neq 0$ $(1 \le k \le m)$. Thus, if a TSA is regular, we can always assume it has no h_k, without loss of generality.

Given two polynomials $p(x) \in Z[x]$ and $q(x) \in Z[x]$, suppose $p(x)$ and $q(x)$ has no common zeros, i.e., resultant$(p, q, x) \neq 0$, and $\alpha_1 < \alpha_2 < ... < \alpha_n$ are all distinct real zeros of $p(x)$. By modified Uspensky algorithm [6], we can get a sequence of intervals, $[a_1, b_1], \cdots, [a_n, b_n]$, which satisfies,

1) $\alpha_i \in [a_i, b_i]$ for $i = 1, \cdots, n$,

2) $[a_i, b_i] \bigcap [a_j, b_j] = \emptyset$ for $i \neq j$,

3) $a_i, b_i (1 \le i \le n)$ are all rational numbers,

4) the maximal size of each isolating interval can be less than any positive number given in advance.

Because $p(x)$ and $q(x)$ has no common zeros, the intervals can also satisfy,

5) any zeros of $q(x)$ are not in any $[a_i, b_i]$.

In the following, we denote the above algorithm by `nearzero`(p, q, x) which plays a very important role in our method.

Theorem 2.1 *Given a regular* TSA*. Suppose $f_1(x_1)$ has n distinct real zeros, then, by calling* `nearzero`$(f_1, BP(x_1), x_1)$ *we can get a sequence of intervals,* $[a_1, b_1], \cdots, [a_n, b_n]$, *which satisfies that, for* $\forall [a_i, b_i] (1 \le i \le n)$ *and* $\forall \beta, \gamma \in [a_i, b_i]$,

1) *if $s > 1$, the system*

$$\begin{cases} f_2(\beta, x_2) = 0, \\ \cdots\cdots, \\ f_s(\beta, x_2, ..., x_s) = 0, \\ g_1(\beta, x_2, ..., x_s) \ge 0, ..., g_r(\beta, x_2, ..., x_s) \ge 0, \\ g_{r+1}(\beta, x_2, ..., x_s) > 0, ..., g_t(\beta, x_2, ..., x_s) > 0, \\ h_1(\beta, x_2, ..., x_s) \neq 0, ..., h_m(\beta, x_2, ..., x_s) \neq 0, \end{cases}$$

and the system

$$\begin{cases} f_2(\gamma, x_2) = 0, \\ \cdots\cdots, \\ f_s(\gamma, x_2, ..., x_s) = 0, \\ g_1(\gamma, x_2, ..., x_s) \geq 0, ..., g_r(\gamma, x_2, ..., x_s) \geq 0, \\ g_{r+1}(\gamma, x_2, ..., x_s) > 0, ..., g_t(\gamma, x_2, ..., x_s) > 0, \\ h_1(\gamma, x_2, ..., x_s) \neq 0, ..., h_m(\gamma, x_2, ..., x_s) \neq 0, \end{cases}$$

have the same number of distinct real solutions and,

2) *if* $s = 1$, *for* $\forall q \in \{g_j(1 \leq j \leq t)\} \bigcup \{h_k(1 \leq k \leq m)\}$, $\mathrm{sign}(q(\beta)) = \mathrm{sign}(q(\gamma))$, *where*

$$\mathrm{sign}(x) = \begin{cases} 1, & x > 0, \\ 0, & x = 0, \\ -1, & x < 0. \end{cases}$$

Proof. Because the TSA is regular, f_1 has no common zeros with the critical polynomial $BP(x_1)$. So, by calling $\texttt{nearzero}(f_1, BP(x_1), x_1)$ we can get a sequence of intervals which satisfies the five conditions of `nearzero`. If $s = 1$, the conclusion is obvious. So, suppose $s > 1$. Because $BP(x_1) = \prod_{2 \leq i \leq s} Bf_i \cdot \prod_{1 \leq j \leq t} Bg_j$ has no zero on $[a_i, b_i]$, clearly, the sign of each Bf_i and Bg_j is invariant on the interval $[a_i, b_i]$.

First of all, $Bf_2 = \mathrm{Discrim}(f_2, x_2) \neq 0$ on the interval $[a_i, b_i]$ implies the number of distinct real zeros of f_2 is invariant on $[a_i, b_i]$. Furthermore, $Bf_3 \neq 0$ on $[a_i, b_i]$ implies that if $f_2 = 0$, then $\mathrm{Discrim}(f_3, x_3) \neq 0$ on $[a_i, b_i]$, which means the number of distinct real solutions of equations $\{f_2 = 0, f_3 = 0\}$ is invariant on $[a_i, b_i]$. Continuing similar discussions, we get that the number of distinct real solutions of equations $\{f_2 = 0, \cdots, f_s = 0\}$ is invariant on $[a_i, b_i]$. Secondly, $Bg_j \neq 0$ on $[a_i, b_i]$ implies that if $\{f_2 = 0, \cdots, f_s = 0\}$, then $g_j \neq 0$ on $[a_i, b_i]$, which means the number of distinct real solutions of the given TSA without f_1 is invariant on $[a_i, b_i]$. The proof is complete.

In the rest of this section, we discuss irregular TSAs, i.e. $\mathrm{resultant}(BP, f_1, x_1) = 0$, and give a theorem which guarantees that we can always assume a given system to be regular, without loss of generality. Our main tool is WR algorithm [24, 25]. Here are some related definitions and results.

Definition 2.6 [24, 25] (Simplicial)

A normal ascending chain $\{f_1, f_2, ..., f_s\}$ is *simplicial* with respect to a polynomial g if either $\mathrm{prem}(g, f_s, ..., f_1) = 0$ or $\mathrm{res}(g, f_s, ..., f_1) \neq 0$.

Theorem 2.2 [24, 25] *For a triangular set* $AS: \ \{f_1, f_2, ..., f_s\}$ *and a polynomial* g, *there is a constructive algorithm which can decompose* AS *into some normal ascending chains* $AS_i: \ \{f_{i1}, f_{i2}, ..., f_{is}\}\, (1 \leq i \leq n)$, *in which every chain is simplicial w.r.t.* g *and this decomposition satisfies that* $Zero(AS) = \bigcup_{1 \leq i \leq n} Zero(AS_i)$.

Remark 2 This decomposition is called the WR decomposition of AS w.r.t. g and the algorithm is called the WR algorithm. By Theorem 2.2, we always consider the triangular set $\{f_1, f_2, ..., f_s\}$ which appears in a TSA as a normal ascending chain, without loss of generality.

Definition 2.7 [23] (Discrimination Matrix)

Given a polynomial with general symbolic coefficients,

$$f(x) = a_0x^n + a_1x^{n-1} + \cdots + a_n,$$

the following $2n \times 2n$ matrix in terms of the coefficients,

$$\begin{bmatrix} a_0 & a_1 & a_2 & \cdots & a_n & & & & \\ 0 & na_0 & (n-1)a_1 & \cdots & a_{n-1} & & & & \\ & a_0 & a_1 & \cdots & a_{n-1} & a_n & & & \\ & 0 & na_0 & \cdots & 2a_{n-2} & a_{n-1} & & & \\ & & & \cdots & \cdots & & & & \\ & & & \cdots & \cdots & & & & \\ & & & a_0 & a_1 & a_2 & \cdots & a_n \\ & & & 0 & na_0 & (n-1)a_1 & \cdots & a_{n-1} \end{bmatrix}$$

is called the *discrimination matrix* of $f(x)$, and denoted by Discr (f). By d_k or $d_k(f)$ denote the determinant of the submatrix of Discr (f), formed by the first k rows and the first k columns for $k = 1, 2, \cdots, 2n$.

Definition 2.8 [25] (Discriminant Sequence)

Let $D_k = d_{2k}, k = 1, \cdots, n$. We call the n-tuple

$$[D_1, \ D_2, \cdots, D_n]$$

the *discriminant sequence* of $f(x)$. Obviously, the last term D_n is Discrim(f, x).

Definition 2.9 [25, 12] (Principal Subresultants)

Let D_k^t be the submatrix of Discr (f), formed by the first $2n-2k$ rows, the first $2n-2k-1$ columns and the $(2n-2k+t)$th column, where $0 \le k \le n-1, \ 0 \le t \le 2k$. Let $|D_k^t| = \det(D_k^t)$. We call $|D_k^0|$ $(0 \le k \le n-1)$ the kth *principal subresultant* of $f(x)$. Obviously, $|D_k^0| = D_{n-k}$ $(0 \le k \le n-1)$.

Definition 2.10 [25, 12] (Subresultant Polynomial Chain)

For $k = 0, 1, ..., n-1$, let

$$Q_{n+1}(f, x) = f(x), \ Q_n(f, x) = f'(x),$$

$$Q_k(f, x) = \sum_{t=0}^{k} |D_k^t| x^{k-t} = |D_k^0|x^k + |D_k^1|x^{k-1} + \cdots + |D_k^k|.$$

We call $\{Q_0(f, x), \ Q_1(f, x), ..., Q_{n+1}(f, x)\}$ the *subresultant polynomials chain* of $f(x)$.

Theorem 2.3 [25] *Suppose $\{f_1, f_2, ..., f_j\}$ is a normal ascending chain, where K is a field and $f_i \in K[x_1, ..., x_i]$, $i = 1, 2, ..., j$ and $f(y) = a_0y^n + a_1y^{n-1} + \cdots + a_{n-1}y + a_n$ is a polynomial in $K[x_1, ..., x_i][y]$, let*

$$PD_k = \text{prem}(|D_k^0|, f_j, ..., f_1) = \text{prem}(D_{n-k}, f_j, ..., f_1), \ 0 \le k \le n-1.$$

If for some $k_0 \geq 0$,

$$\text{res}(a_0, f_j, ..., f_1) \neq 0$$

$$PD_0 = \cdots = PD_{k_0-1} = 0, \quad \text{res}(|D^0_{k_0}|, f_j, ..., f_1) \neq 0,$$

then, in $K[x_1, ..., x_j]/(f_1, ..., f_j)$*, we have* $\gcd(f, f'_x) = Q_{k_0}(f, x)$.

Now, we deal with irregular TSA with resultant$(BP, f_1, x_1) = 0$. The idea is to decompose this system by WR algorithm into some new regular systems.

- If there is some Bh_k so that resultant$(BP, Bh_k, x_1) = 0$, do WR decomposition of $\{f_1, f_2, ..., f_s\}$ w.r.t. h_k and, without loss of generality, suppose we get two new chains $\{A_1, A_2, ..., A_s\}$ and $\{C_1, C_2, ..., C_s\}$, in which prem$(g_j, A_s, ..., A_1) = 0$ but res$(g_j, C_s..., C_1) \neq 0$. If we replace $\{f_1, f_2, ..., f_s\}$ by $\{C_1, C_2, ..., C_s\}$ in the TSA, the new system is regular. Obviously, another system obtained by replacing $\{f_1, f_2, ..., f_s\}$ with $\{A_1, A_2, ..., A_s\}$ in the TSA, has no real solution.

- If there is some Bg_j so that resultant$(BP, Bg_j, x_1) = 0$, do WR decomposition of $\{f_1, f_2, ..., f_s\}$ w.r.t. g_j and, without loss of generality, suppose we get two new chains $\{A_1, A_2, ..., A_s\}$ and $\{C_1, C_2, ..., C_s\}$, in which prem$(g_j, A_s, ..., A_1) = 0$ but res$(g_j, C_s..., C_1) \neq 0$. Now, if in the TSA we have $g_j > 0$, we simply replace $\{f_1, f_2, ..., f_s\}$ by $\{C_1, C_2, ..., C_s\}$ and the new system is regular. If in the TSA we have $g_j \geq 0$, we first get a system TSA1 by replacing $\{f_1, f_2, ..., f_s\}$ with $\{C_1, C_2, ..., C_s\}$ and then, get another system TSA2 by replacing $\{f_1, f_2, ..., f_s\}$ with $\{A_1, A_2, ..., A_s\}$ and deleting g_j from it. These two systems are both regular.

- If there is some Bf_i so that resultant$(BP, Bf_i, x_1) = 0$, let $[D_1, \cdots, D_{n_i}]$ be the discriminant sequence of f_i w.r.t. x_i. First of all, we do WR decomposition of $\{f_1, ..., f_{i-1}\}$ w.r.t. D_{n_i} and, without loss of generality, suppose we get two new chains $\{A_1, ..., A_{i-1}\}$ and $\{C_1, ..., C_{i-1}\}$, in which prem$(D_{n_i}, A_{i-1}, ..., A_1) = 0$ but res$(D_{n_i}, C_{i-1}..., C_1) \neq 0$. Step 1, replacing $\{f_1, ..., f_{i-1}\}$ with $\{C_1, ..., C_{i-1}\}$, we will get a regular system. Step 2, let us consider the system obtained by replacing $\{f_1, ..., f_{i-1}\}$ with $\{A_1, ..., A_{i-1}\}$ which is still irregular. Consider D_{n_i-1}, the next term in $[D_1, \cdots, D_{n_i}]$. If res$(D_{n_i-1}, A_{i-1}, ..., A_1) = 0$, do WR decomposition of $\{A_1, ..., A_{i-1}\}$ w.r.t. D_{n_i-1}. Keep repeating the same procedure [1] until at a certain step we have, for certain D_{i_0} and $\{\bar{A}_1, ..., \bar{A}_{i-1}\}$, res$(D_{i_0}, \bar{A}_{i-1}, ..., \bar{A}_1) \neq 0$ and $\forall j\ (i_0 < j \leq n_i)$, prem$(D_j, \bar{A}_{i-1}, ..., \bar{A}_1) = 0$. By Theorem 2.3, we have $\gcd(f_i, f'_i) = Q_{n_i-i_0}(f_i, x_i)$ in $K[x_1, ..., x_{i-1}]/(\bar{A}_1, ..., \bar{A}_{i-1})$. Now, let $\bar{f}_i$ be the pseudo quotient of f_i divided by $\gcd(f_i, f'_i)$ and replace $\{f_1, ..., f_{i-1}, f_i\}$ with $\{\bar{A}_1, ..., \bar{A}_{i-1}, \bar{f}_i\}$, the new system will be regular.

- If by repeating above three kinds of processes, we decompose an irregular TSA into some regular systems $TSC_i\,(1 \leq i \leq n)$ and we have $Zero(TSC_{i1}) \subseteq Zero(TSC_{i2})$ for some $i1 \neq i2$, $1 \leq i1, i2 \leq n$, then delete system TSC_{i1}.

[1] This procedure must terminate because $\{f_1, ..., f_s\}$ being a normal ascending chain implies res$(I_i, f_{i-1}, ..., f_1) \neq 0$ and $D_1 = n_i I_i^2$ implies res$(D_1, f_{i-1}, ..., f_1) \neq 0$.

Thus, we have got

Theorem 2.4 *For an irregular* TSA*, there is a constructive algorithm which can decompose* TSA *into some regular systems TSC_i. Let $Rzero(\cdot)$ denote all the distinct real solutions of a given system, then this decomposition satisfies $Rzero(TSC) = \sum Rzero(TSC_i)$.*

3 Lifting and Recursion

By Theorem 2.4, we need only to consider regular TSA. For a regular TSA, by calling `nearzero`$(f_1(x_1), BP(x_1), x_1)$, we can get a sequence of intervals satisfying the 5 conditions of algorithm `nearzero`. How do we take use of these isolating intervals of $f_1(x_1)$ to get those of $f_2(x_2), ..., f_s(x_s)$?

Given a regular TSA T, for every $f_i (i \geq 2)$, let

$$U_{ij} = \begin{cases} \mathrm{res}\,(\frac{\partial f_i}{\partial x_j}, f_i, f_{i-1}, \cdots, f_2), & \frac{\partial f_i}{\partial x_j} \not\equiv 0, \\ 1, & \frac{\partial f_i}{\partial x_j} \equiv 0, \end{cases} \quad (2 \leq i \leq s,\ 1 \leq j \leq i-1),$$

$$MP_T(x_1) = \prod_{2 \leq i \leq s} \prod_{1 \leq j \leq i-1} U_{ij}.$$

Algorithm: `REALZERO`

input: a regular TSA T_1

output: isolating intervals of real solutions of T_1 or reports fail

Step 0 i:=1;

Step 1 resultant$(f_i(x_i), MP_{T_i}(x_i), x_i) = 0$? If yes, the algorithm does not work, stop; if no, by `nearzero`$(f_i(x_i), BP_{T_i} \cdot MP_{T_i}, x_i)$, get a sequence of isolating intervals on x_i, say $S^{(i)}$.

Step 2 For each element $I = [a^{(1)}, b^{(1)}] \times \cdots \times [a^{(i)}, b^{(i)}]$ in $S^{(i)}$, let V_I be the vertexes of the i-dimensional cube I.

Step 3 Because resultant$(f_i(x_i), MP_{T_i}(x_i), x_i) \neq 0$, regarding x_{i+1} as an implicitly defined function by f_{i+1} in $x_1, x_2, ..., x_i$, respectively, x_{i+1} is monotonic when x_1 is on $[a^{(1)}, b^{(1)}]$,..., x_i is on $[a^{(i)}, b^{(i)}]$. For every vertex $(v_j^{(1)}, ..., v_j^{(i)})$ in V_I, substitute $x_1 = v_j^{(1)}, ..., x_i = v_j^{(i)}$ into T_1 and delete the first i equations of it. Let $i := i+1$, we denote the new system by $T_j^{(i)}$. Regarding $T_j^{(i)}$ as a new regular TSA, do Step 1. If not exit, we get a sequence of isolating intervals on x_i, say $S_j^{(i)} : [\alpha_{j,1}^{(i)}, \beta_{j,1}^{(i)}], ..., [\alpha_{j,n_i}^{(i)}, \beta_{j,n_i}^{(i)}]$.

Step 4 By Theorem 2.1, the numbers of intervals in any two sequences, $S_{j_1}^{(i)}$ and $S_{j_2}^{(i)}$ $(1 \le j_1 \le j_2 \le |V_I|)$, are the same. So, we merge these sequences into one: $S^{(i)}: [\alpha_1^{(i)}, \beta_1^{(i)}], ..., [\alpha_{n_i}^{(i)}, \beta_{n_i}^{(i)}]$, where

$$\alpha_k^{(i)} = \min(\alpha_{1,k}^{(i)}, ..., \alpha_{|V_I|,k}^{(i)}),\ \beta_k^{(i)} = \max(\beta_{1,k}^{(i)}, ..., \beta_{|V_I|,k}^{(i)}),\ (1 \le k \le n_i).$$

If two intervals in $S^{(i)}$ intersect or the maximal size of these intervals exceeds the given positive number, by a sub-algorithm below, we shrink the cube I and still denote it by I. Let $i := i - 1$, back to Step 2; else we get a sequence of i-dimensional cube: $I \times S^{(i)}$. We denote the sequence still by $S^{(i)}$. If $i < s$, back to Step 2, else

Step 5 For each s-dimensional cube $I = [a^{(1)}, b^{(1)}] \times \cdots \times [a^{(s)}, b^{(s)}]$ in $S^{(s)}$, substitute $x_1 = a^{(1)}, ..., x_s = a^{(s)}$ into each g_j $(1 \le j \le t)$ and check whether $g_j > 0$ or $g_j \ge 0$ according to T_1. If all the inequalities satisfied, output I.

Sub-algorithm: `SHR`

input: a k-dimensional cube I_0 in $S^{(k)}$

output: a k-dimensional cube $I \subset I_0$

Step 0 Suppose $I_0 = [a_1, b_1] \times \cdots \times [a_k, b_k]$. We know $f_1(x_1)$ has one and only one zero, say x_1^0, in $[a_1, b_1]$ (Without loss of generality, we assume $f_1(x_1)$ has no repeated zeros). By intermediate value theorem, we can get an interval $[a_1^{'}, b_1^{'}] \subset [a_1, b_1]$ with $x_1^0 \in [a_1^{'}, b_1^{'}]$ and $b_1^{'} - a_1^{'} = (b_1 - a_1)/10$.

Step 1 Let $i := 1$, $I = [a_1^{'}, b_1^{'}]$ and V_I be the vertexes of the i-dimensional cube I.

Step 2 For every vertex $(v_j^{(1)}, ..., v_j^{(i)})$ in V_I, substitute $x_1 = v_j^{(1)}, ..., x_i = v_j^{(i)}$ into T_1 and delete the first i equations of it. Let $i := i + 1$, we denote the new system by $T_j^{(i)}$. Regarding $T_j^{(i)}$ as a new regular TSA, call `nearzero` $(f_i(x_i), BP_{T_i} \cdot MP_{T_i}, x_i)$. When calling `nearzero`, let the maximal size of intervals be 1/10 of that when we computed $[a_i, b_i]$ in `REALZERO`. We get a sequence of isolating intervals on x_i, say $S_j^{(i)}$.

Step 3 Merge $S_j^{(i)}$ $(1 \le j \le |V_I|)$ into one sequence $S^{(i)}$. Of course we know $[a_i, b_i]$ should correspond to which interval in $S^{(i)}$. Denote the interval by $[a_i^{'}, b_i^{'}]$.

Step 4 Denote $I \times [a_i^{'}, b_i^{'}]$ still by I. If i=k, output I; else let V_I be the vertexes of the i-dimensional cube I and back to Step 2.

For a regular TSA with resultant$(f_i(x_i), MP_{T_i}(x_i), x_i) \neq 0$, the algorithm `REALZERO` can isolate the real solutions of the system. It has been found to be very efficient in practice on a large number of problems with various backgrounds though it is not a complete one in theory.

4 `Realzero` and Examples

Combining the algorithm `REALZERO` with Ritt-Wu method and WR algorithm, we generalize `REALZERO` to deal with general semi-algebraic systems SAS which is defined in Section 1. Our method has been implemented in MAPLE program `realzero`.

There are three basic kinds of calling sequences for SAS in MAPLE:

$$\begin{array}{l}
\texttt{realzero}([p_1,\cdots,p_s],[q_1,\cdots,q_r],[g_1,\cdots,g_t],[h_1,\cdots,h_m],[x_1,\cdots,x_s]);\\
\texttt{realzero}([p_1,\cdots,p_s],[q_1,\cdots,q_r],[g_1,\cdots,g_t],[h_1,\cdots,h_m],[x_1,\cdots,x_s],width);\\
\texttt{realzero}([p_1,\cdots,p_s],[q_1,\cdots,q_r],[g_1,\cdots,g_t],[h_1,\cdots,h_m],[x_1,\cdots,x_s],[w_1,...,w_s]);\ .
\end{array}$$

The command `realzero` returns a list of isolating intervals for all real solutions of the input system or reports the method does not work on some branches. If the 6-th parameter "width", a positive number, is given, the maximal size of the output intervals is less than or equal to the number. If the 6-th parameter is a list of positive numbers, $[w_1, ..., w_s]$, the maximal sizes of the output intervals on $x_1, ...$ and x_s are less than or equal to $w_1, ...$ and w_s, respectively. If the 6-th parameter is omitted, the most convenient width is used for each interval returned. In what follows, all the examples were computed on a Pentium/800 PC with 256 Mb RAM under MAPLE V.4.

Example 1 Given a system of polynomial equations [2],

$$\begin{cases}
p_1 = x_1(2-x_1-y_1)+x_2/2-x_1/2=0,\\
p_2 = x_2(2-x_2-y_2)+x_1/2-x_2/2=0,\\
p_3 = y_1(5-x_1-2y_1)+y_2/2-y_1/2=0,\\
p_4 = y_2(3/2-x_2-2y_2)+y_1/2-y_2/2=0,
\end{cases}$$

find the isolating intervals of positive solutions and non-negative solutions of it.

Call

$$\texttt{realzero}\,([p_1,p_2,p_3,p_4],[x_1,x_2,y_1,y_2],[\,],[\,],[x_1,x_2,y_1,y_2],1/1000);$$

the output is (the maximal size of all the intervals is less than 1/1000),

$$\begin{array}{l}
[[[\frac{123699}{262144}, \frac{151}{320}],\\
\quad [\frac{156047501938406335153557625253476418829899981}{154296032586880081850687976687470345226955597}, \frac{25646736065207290639}{25350470632055620751}],\\
\quad [\frac{31940045261606640254 9}{15210282379233372450 6}, \frac{6480771405445170790944400919067165781120176 5}{3085920651737601637013759533749406904539119 4}],\\
\quad [\frac{11766526981955972576 8}{16304965803039035040 1}, \frac{2386788743612120084475521809759314652066228 0}{3307054016778071802309848103602576881598825 7}]],\\
\quad [[0, 0], [0, 0], [0, 0], [0, 0]], [[0, 0], [0, 0], [\frac{77397}{32768}, \frac{38699}{16384}], [\frac{283969593}{268435456}, \frac{71012665}{67108864}]],\\
\quad [[2, 2], [2, 2], [0, 0], [0, 0]]],
\end{array}$$

which means the system has 4 non-negative real solutions and obviously, only one of them is positive. The time spent is 4.855 seconds.

[2]The example was provided by Prof. Lu Zhengyi in a talk in Chengdu, China this spring, who proposed a method, based on an entirely different principle, for isolating the real solutions of a polynomial system of equations.

Example 2 This example is one of the special cases of "p-3-p" problem [8]. Given

$$\begin{cases} p_1 = x^2 + y^2 - xy - 1 = 0, \\ p_2 = y^2 + z^2 - yz - 4 = 0, \\ p_3 = 100z^2 + 100x^2 - 100zx - 441 = 0, \\ x > 0, y > 0, z > 0, \end{cases}$$

call

$$\texttt{realzero}\,([p_1, p_2, p_3], [\,], [x, y, z], [\,], [x, y, z], 1/100);$$

the output is (the maximal size of all the intervals is less than 1/100),

$$[[[\frac{329}{640}, \frac{1317}{2560}], [\frac{7484377440343}{6494479183360}, \frac{52609887507}{45636563840}], [\frac{43054377865877}{18655898501120}, \frac{24498926286479469}{10606777162792960}]]].$$

The time spent is 0.510 seconds.

Example 3 This problem originates from automated proving for inequality-type theorems. Given

$$\begin{cases} p_1 = (x-y)^2 + (y-z)^2 + (z-x)^2 - 13/10 = 0, \\ p_2 = (x+y+z)xyz + 1/25 = 0, \\ p_3 = (x+y)^2 + (y+z)^2 + (z+x)^2 - 1 = 0, \\ x + y > 0, y + z > 0, z + x > 0, \end{cases}$$

call

$$\texttt{realzero}\,([p_1, p_2, p_3], [\,], [x+y, y+z, z+x], [\,], [x, y, z], [1/100, 1/10, 1/10, 1/10]);$$

the output is (1.211 seconds),

$$[[[\frac{-6081}{20480}, \frac{-19}{64}], [\frac{173}{512}, \frac{87}{256}], [\frac{2121}{3520}, \frac{265581}{429568}]], [[\frac{-609}{2048}, \frac{-19}{64}], [\frac{35}{64}, \frac{5}{8}], [\frac{283}{840}, \frac{57423}{163520}]],$$

$$[[\frac{6939}{20480}, \frac{347}{1024}], [\frac{-609}{2048}, \frac{-607}{2048}], [\frac{9197}{15360}, \frac{360041}{579584}]], [[\frac{693}{2048}, \frac{347}{1024}], [\frac{155}{256}, \frac{157}{256}], [\frac{-370699}{1248000}, \frac{-733683}{2474240}]],$$

$$[[\frac{1249}{2048}, \frac{625}{1024}], [\frac{-39}{128}, \frac{-37}{128}], [\frac{66473}{210560}, \frac{145251}{400000}]], [[\frac{1249}{2048}, \frac{625}{1024}], [\frac{43}{128}, \frac{11}{32}], [\frac{-46663}{156320}, \frac{-366839}{1239680}]]].$$

Example 4 Given following regular TSA,

$$\begin{cases} f_1(b) = 0, \\ f_2(b, c) = 0, \\ f_3(b, c, d) = 0, \\ f_4(b, c, d, e) = 0, \\ b > 0, c > 0, d > 0, e > 0, c - d \neq 0, \end{cases}$$

where f_1 is a polynomial in b with degree 32, which is given in the appendix, and

$$\begin{aligned} f_2 = {} & 2075\, b^{16}\, c^{12} + 284580\, b^{14}\, c^{10} + 357840\, b^{12}\, c^9 + 10185588\, b^{12}\, c^8 + 20167488\, b^{10}\, c^7 \\ & + (21285312\, b^8 - 62355744\, b^{10})\, c^6 - 99610560\, b^8\, c^5 \\ & + (-4855244976\, b^8 - 361573632\, b^6)\, c^4 + (-37158912\, b^4 - 3758980608\, b^6)\, c^3 \\ & + (54181472832\, b^6 + 429235200\, b^4)\, c^2 + (4897760256\, b^4 + 488374272\, b^2)\, c \\ & - 123974556480\, b^4 + 18874368 + 9432723456\, b^2, \end{aligned}$$

$$
\begin{aligned}
f_3 \; &= 9\, b^4\, d^3 + 45\, b^4\, c\, d^2 + (35\, b^4\, c^2 - 486\, b^2)\, d - 108\, b^2\, c - 264 + 10\, b^4\, c^3, \\
f_4 \; &= (36\, d^2\, b^2 - 8\, c^2\, b^2 - 28\, d\, b^2\, c)\, e + 15\, b^4\, c\, d^2 + 6\, b^4\, c^3 + 21\, b^4\, c^2\, d - 144\, b^2\, c \\
&\quad + 9\, b^4\, d^3 - 120 - 648\, b^2\, d.
\end{aligned}
$$

We call

$$\texttt{realzero}([f_1, f_2, f_3, f_4], [\,], [b, c, d, e], [c - d], [b, c, d, e]);$$

and get (201.977 seconds),

$$
\begin{aligned}
&[[[\frac{741}{2048}, \frac{1483}{4096}], [\frac{76905}{32768}, \frac{76995}{32768}], [17, \frac{35}{2}], \\
&\quad [\frac{10861925319343565779854723937}{22127792367701489429879193600}, \frac{16551194692093223298992446177 9}{33326217932928391846374355763 2}]], [\\
&\quad [\frac{741}{2048}, \frac{1483}{4096}], [\frac{199727}{32768}, \frac{49971}{8192}], [\frac{21}{2}, 11], \\
&\quad [\frac{10501218509973981520215735655}{4910135500314640502581362688}, \frac{2424563760027166415456804899}{1058153874210595032992317440}]], \\
&\quad [[\frac{1803}{2048}, \frac{3607}{4096}], [\frac{9}{4}, \frac{289}{128}], [\frac{17}{4}, \frac{9}{2}], [\frac{23829095983254931}{4908557229096960}, \frac{68808656977494510860283}{13000075334176032686080}]], \\
&\quad [[\frac{1803}{2048}, \frac{3607}{4096}], [\frac{311}{128}, \frac{39}{16}], [\frac{31}{8}, 4], [\frac{13934002895579729859 19}{20385274532122853376 0}, \frac{624427174855568222 43}{851463164497441587 2}]] \\
&\quad , [[\frac{8177}{4096}, \frac{4089}{2048}], [\frac{1935}{2048}, \frac{121}{128}], [\frac{29}{16}, \frac{15}{8}], \\
&\quad [\frac{2156344139389111695038220 07}{1868790848887422525112320 0}, \frac{31325177508416778452 29}{25795294217386328064 0}]], [[\frac{39343}{4096}, \frac{2459}{256}], \\
&\quad [\frac{97}{512}, \frac{195}{1024}], [\frac{769}{2048}, \frac{1547}{4096}], \\
&\quad [\frac{599554507678818070801636466 1}{105759302845003541459763200}, \frac{45550235812704818962737}{7895725093641508618 24}]]].
\end{aligned}
$$

References

[1] P. Aubry, D. Lazard and M. Moreno Maza, On the theories of triangular sets, *J. Symb. Comput.* **28** 105-124 (1999).

[2] L. Blum, F. Cucker, M. Shub and S. Smale, *Complexity and Real Computation*, Springer-verlag, New York, 1997.

[3] L. Blum, M. Shub and S. Smale, On the theory of computation and complexity over the real numbers: NP-completeness, recursive functions and universal machines, *Bulletin of the Amer. Math. Soc.*, **21** 1-46 (1989).

[4] B. Buchberger, Gröbner bases: An algorithmic method in polynomial ideal theory, In *Multidimensional Systems Theory*, (Edited by N.K. Bose), pp. 184-232, D. Reidel, Dordrecht, (1985).

[5] G. E. Collins, Quantifier elimination for real closed fields by cylindrical algebraic decomposition, in *Lecture Notes in Computer Science* **33** 134-183 (1975).

[6] G. E. Collins and R. Loos, Real Zeros of Polynomials, In *Computer Algebra: Symbolic and Algebraic Computation*, (Edited by B. Buchberger *et al.*), pp. 83-94, Springer-Verlag, New York, (1983).

[7] D. Cox, J. Little, and D. O'Shea, *Using Algebraic Geometry*, Springer-Verlag, New York (1998).

[8] X.-S. Gao and H. F. Cheng, On the Solution Classification of the "P3P" Problem, In *Proc. of ASCM'98*, (Edited by Z. Li), pp. 185-200, Lanzhou University Press, Lanzhou, (1998).

[9] D. Grigoriev and N. Vorobjov, Solving systems of polynomial inequalities in subexponential time, *J. of Symbolic Computation* **5** 37-64 (1988).

[10] L. Gonzalez-Vega, F. Rouillier, M.-F. Roy and G. Trujillo, Symbolic Recipes for Real Solutions, In *Some Tapas of Computer Algebra*, (Edited by A.M. Cohen *et al.*), pp. 121-167, Springer-Verlag, New York, (1999).

[11] D. Lazard, A new method for solving algebraic systems of positive dimension, *Discrete Appl. Math.* **33** 147-160 (1991).

[12] R. Loos, Generalized Polynomial Remainder Sequences, In *Computer Algebra: Symbolic and Algebraic Computation*, (Edited by B. Buchberger *et al.*), pp. 115-137, Springer-Verlag, New York, (1983).

[13] S. L. Ma and Z. M. Zheng, On a cubic system with eight small-amplitude limit cycles, *Appl. Math. Lett.*, vol. 7, (4)1994, 23-27.

[14] S. Smale, The fundamental theorem of algebra and complexity theory, *Bulletin of the Amer. Math. Soc.*, **4** 1-36 (1981).

[15] A. Tarski, *A Decision Method for Elementary Algebra and Geometry*, University of California Press, 1951.

[16] D. M. Wang, A class of cubic differential systems with 6-tuple focus, *J. Diff. Eqns.*, **87** 305-315 (1990).

[17] D. M. Wang, An elimination method for polynomial systems, *J. Symb. Comput.* **16** 83-114 (1993).

[18] D. M. Wang, Decomposing polynomial systems into simple systems, *J. Symb. Comput.* **25** 295-314 (1998).

[19] W. T. Wu, On zeros of algebraic equations–An application of Ritt principle, *Kexue Tongbao* **31** 1-5 (1986).

[20] W. T. Wu, *Mechanical theorem proving in geometries: Basic principles* (translated from Chinese by X. Jin and D. Wang), Springer, New York, (1994).

[21] L. Yang, X. R. Hou and B. C. Xia, Automated Discovering and Proving for Geometric Inequalities, In *Automated Deduction in Geometry*, (Edited by X.-S. Gao *et al.*), *LNAI 1669*, pp. 30-46, Springer-Verlag, New York, (1999).

[22] L. Yang, X. R. Hou and B. C. Xia, A Complete Algorithm for Automated Discovering of a Class of Inequality-Type Theorems, *Science in China*, Series **F**, Vol. 44, 33-49, 2001.

[23] L. Yang and B. C. Xia, An explicit criterion to determine the number of roots in an interval of a polynomial, *Progress in Natural Science*, Vol. 10, No. 12, 897-910 (2000).

[24] L. Yang, J. Z. Zhang and X. R. Hou, An Efficient Decomposition Algorithm for Geometry Theorem Proving Without Factorization, In *Proceedings of Asian Symposium on Computer Mathematics 1995*, (Edited by H. Shi *et al.*), pp.33-41, Scientists Incorporated, Japan, (1995).

[25] L. Yang, J. Z. Zhang and X. R. Hou, *Nonlinear Algebraic Equation System and Automated Theorem Proving* (in Chinese), Shanghai Scientific and Technological Education Publishing House, Shanghai (1996).

Appendix

$$\begin{aligned}
f_1 = {} & 241538508382138075462768483549507937558926051383237186598921\backslash \\
& 35143477508299833761559265231377708635407176637146131171128509762\backslash \\
& 9761b^{32} + 6350667137788405987107494985775044967930708508840979479 74\backslash \\
& 38029219177777722424790935669882905230018966867662706346221816526\backslash \\
& 252732166 40b^{30} - 6175167296855942313472468772823089190877893406 0236\backslash \\
& 33346079511963379997673499794946894262027603963333723121547282957\backslash \\
& 2882495611572684 8000b^{28} + 2739003464675363976662421206959900129096 7\backslash \\
& 59448312686194199639473757366983350460922339943178170551929762470\backslash \\
& 25147731418749702808210505728 0b^{26} - 1437145166237554579477639351890\backslash \\
& 59794915618143392460779148627234144024674310258895855296843282026\backslash \\
& 2735689676445367034239551743254142648320b^{24} - 298609258728339835915\backslash \\
& 11873209280400659942863793385889444751464527738926059859502184401\backslash \\
& 2436391877650836905308408943702288447254625779712b^{22} + 435447287511\backslash \\
& 29852462155896216013929270344442811835212525492551812771844033485\backslash \\
& 66248913207745838840779180167383076700642516430126841856 0b^{20} - 3414\backslash \\
& 88074367456093473004956003122708578333573667293973935929910141878\backslash \\
& 354056591935239593924781478529672997249005700302610906831283814 4b^{18} + \\
& 82237565552698611657570566658152443771941213949595401920250155054\backslash \\
& 50490851490444645492294205808268848998735781764749955762521605406\backslash \\
& 72b^{16} - 184658639115346142227714072547272185405530779039814940607 26\backslash \\
& 33493473284280274236822702569461113776661096178555364273916711096\backslash \\
& 2872320b^{14} + 2104583981543015152643938721287577501834262434998301 92\backslash \\
& 25945815801961543179698127213788008273551581371156105365020925245\backslash \\
& 8008412160b^{12} - 1397866755744633176764219378285539600474935695399 85\backslash
\end{aligned}$$

$182506772060979341228101552328831048785648035273563871414131172 28\backslash$
$188455403 52b^{10} + 5851882153024252534337022484145153131833664445 3000\backslash$
$074974420330373344768912105949137931244320397583710622673511160 39\backslash$
$560626176b^{8} - 155390183378463952221186520858978074062380250509 9793\backslash$
$477782141492270038759399553741113732276673307699808273731883493 01\backslash$
$88288b^{6} + 3139960540165071204464736713291845422900077966277745 6747\backslash$
$422632241786296296046865042734023650341502533877789531725365248b^{4} -$
$221479815284662082377510954691437636974994885572262012135141667 02\backslash$
$1889911271014168057494169088077631899897509875548 16b^{2} + 6072087665\backslash$
$340276114250766413149532025614824736717699042961055022961306776 39\backslash$
$9525491814383795284511167695839821824$