Automatic Computation of the Complete Root Classification for a Parametric Polynomial

Songxin Liang *, David J. Jeffrey

Department of Applied Mathematics, University of Western Ontario, London, Ontario, Canada

Abstract

An improved algorithm, together with its implementation, is presented for the automatic computation of the complete root classification of a real parametric polynomial. The algorithm offers improved efficiency and a new test for non-realizable conditions. The improvement lies in the direct use of 'sign lists', obtained from the discriminant sequence, rather than 'revised sign lists'. It is shown that the discriminant sequences, upon which the sign lists are based, are closely related both to Sturm-Habicht sequences and to subresultant sequences. Thus calculations based on any of these quantities are essentially equivalent.

One particular application of complete root classifications is the determination of the conditions for the positive definiteness of a polynomial, and here the new algorithm is applied to a class of sparse polynomials. It is seen that the number of conditions for positive definiteness remains surprisingly small in these cases.

Key words: complete root classification, parametric polynomial, real quantifier elimination, real root, subresultant polynomials.

1. Introduction

Counting and classifying the roots of a polynomial is a well-established problem area; see Basu, Pollack & Roy (2003) for references. Our present goal is to compute automatically the Complete Root Classification (CRC) of a parametric polynomial. The CRC has been applied in studies of ordinary differential equations, of integral equations, of mechanics problems, and to real quantifier elimination; for references see Liang & Jeffrey (2006). This paper describes two improvements that enable more efficient automatic computation. As well, an implementation in Maple is used to solve a series of problems in quantifier elimination, specifically in positive definiteness testing.

* Corresponding author.
Email addresses: `sliang22@uwo.ca` (Songxin Liang), `djeffrey@uwo.ca` (David J. Jeffrey).

Definition 1 (RC and CRC)**.** Let $p(x) \in \mathbb{R}[x]$. The root classification (RC) of $p(x)$ is denoted by $[\,[n_1, n_2, \ldots], [m_1, -m_1, m_2, -m_2, \ldots]\,]$ where n_k are the multiplicities of the distinct real roots of $p(x)$, and m_k are the multiplicities of the distinct complex conjugate pairs of $p(x)$.

For a polynomial $p(x)$ with real parametric coefficients, the complete root classification (CRC) of $p(x)$ is a collection of its all possible root classifications (RCs), together with the conditions on the parametric coefficients such that each RC is realized.

The CRC of a real parametric quartic polynomial was found by Arnon (1988), but the first method for establishing the CRC of a real parametric polynomial of any degree was given by Yang, Hou & Zeng (1996), using their discriminant sequence. They illustrated their method by computing the CRC of a reduced sextic polynomial. Liang & Zhang (1999) proposed and implemented an algorithm for the automatic generation of a CRC, and also extended the approach to complex parametric polynomials. Further improvements to the algorithm were proposed by Liang & Jeffrey (2006). In parallel, Gonzalez-Vega (1998) proposed the use of Sturm-Habicht sequences (Gonzalez-Vega, Lombardi, Recio & Roy, 1998) to solve similar problems. We show here that discriminant sequences, principal Sturm-Habicht coefficient sequences, principal subresultant coefficient sequences and signed subresultant coefficient sequences (Basu, Pollack & Roy, 2003) are equivalent for CRC computations.

This paper presents several advances on the above works. The main efficiency improvement is to work directly with 'sign lists', defined below, rather than 'revised sign lists'. A second improvement concerns the conditions generated. The basic approach, using any of the above sequences, produces a large set of conditions on the (symbolic) coefficients of the polynomial. A separate, but important, task is to reduce this set to a more manageable size, both by eliminating conditions that can never be realized, and by combining conditions into more compact forms. The automation of this step is also desirable. Here a new method is used to filter extraneous cases during the generation of the results.

The new algorithm has been implemented in Maple. As an application, some well-known benchmark problems are considered: the positive definiteness of polynomials. However, it should be emphasized that the CRC of a polynomial contains more information than is needed for these problems, and consequently it has more potential applications than the examples given here. As the problem of real quantifier elimination is well-known to be unsolvable in polynomial time for the general case (Davenport & Heintz, 1988), we have concentrated on a class of sparse polynomials, for which surprisingly compact results are possible.

The rest of the paper is organized as follows. In Section 2, the relationships between the discriminant sequence of a polynomial and other related concepts are discussed. In Section 3, existing algorithms related to CRC and their weaknesses are reviewed. In Section 4, the definitions and theorems on which the improved algorithm is based are presented. In Section 5, the improved algorithm is proposed, and its features are discussed. In Section 6, some CRCs and their applications to real quantifier elimination are shown. Finally, in Section 7, some issues for future consideration are mentioned.

2. Relationships among Different Concepts

As mentioned above, different approaches use equivalent constructions for CRC calculations. We show the equivalences here before reviewing the existing algorithms.

2.1. Discriminant Sequence and Related Sequences

Yang, Hou & Zeng (1996) defined the following quantities as the basis of their algorithm. Let $p \in \mathbb{R}[x]$ and write $p(x) = a_n x^n + a_{n-1}x^{n-1} + \cdots + a_0$, where $a_n \neq 0$.

Definition 2. The discrimination matrix of p is the $2n \times 2n$ matrix

$$M = \begin{pmatrix} a_n & a_{n-1} & a_{n-2} & \dots & a_0 & & & & \\ 0 & na_n & (n-1)a_{n-1} & \dots & a_1 & & & & \\ & a_n & a_{n-1} & \dots & a_1 & a_0 & & & \\ & 0 & na_n & \dots & 2a_2 & a_1 & & & \\ & & & \vdots & \vdots & & & & \\ & & & a_n & a_{n-1} & a_{n-2} & \dots & a_0 \\ & & & 0 & na_n & (n-1)a_{n-1} & \dots & a_1 \end{pmatrix} . \quad (1)$$

Definition 3 (Discriminant Sequence)**.** For $1 \leq k \leq 2n$, let M_k be the kth principal minor of M, and let $D_k = M_{2k}$. The n-tuple $D = [D_1, D_2, \dots, D_n]$ is called the discriminant sequence of p.

Definition 4 (Sign List)**.** If $[D_1, D_2, \dots, D_n]$ is the discriminant sequence of p and $\operatorname{sgn} x$ is the signum function with $\operatorname{sgn} 0 = 0$, then the sign list of p is $[\operatorname{sgn} D_1, \operatorname{sgn} D_2, \dots, \operatorname{sgn} D_n]$.

Definition 5 (Revised Sign List)**.** The revised sign list $[e_1, e_2, \dots, e_n]$ of p is constructed from the sign list $s = [s_1, s_2, \dots, s_n]$ of p as follows.

If $[s_i, s_{i+1}, \dots, s_{i+j}]$ is a section of s, where $s_i \neq 0$, $s_{i+1} = s_{i+2} = \ldots = s_{i+j-1} = 0$ and $s_{i+j} \neq 0$, then we replace the subsection $[s_{i+1}, \dots, s_{i+j-1}]$ by $[-s_i, -s_i, s_i, s_i, -s_i, -s_i, \dots]$ such that $e_{i+r} = (-1)^{\lfloor (r+1)/2 \rfloor} s_i$ $(r = 1, 2, \dots, j-1)$, and keep other elements unchanged, i.e., let $e_k = s_k$.

The revised sign list of p is denoted by $\operatorname{rsl}(p)$. Similarly, the revised sign list of s is denoted by $\operatorname{rsl}(s)$.

Definition 6 (Δ-Sequence)**.** Let $\Delta(p)$ denote $\gcd(p(x), p'(x))$, and let $\Delta^0(p) = p(x)$, $\Delta^j(p) = \Delta(\Delta^{j-1}(p))$, $j = 1, 2, \dots$. Then $\Delta^0(p), \Delta^1(p), \Delta^2(p), \dots$ is called the Δ-sequence of p.

Definition 7. The multiple factor sequence of p, denoted $\Theta_0(p), \Theta_1(p), \dots, \Theta_{n-1}(p)$ is defined using submatrices of (1). Let $M_{k,i}$ denote the submatrix formed by the first $2k$ rows of M, the first $2k-1$ columns of M and the $(2k+i)$th column of M. In the notation of the MAPLE `LinearAlgebra` package (Jeffrey & Corless, 2006), $M_{k,i}$ is the matrix

```
< M[ 1..2*k , 1..2*k-1 ] | M[ 1..2*k , 2*k+i ] > .
```

Then

$$\Theta_k(p) = \sum_{i=0}^{k} \det(M_{n-k,i}) x^{k-i} , \quad \text{for} \quad k = 0, \dots, n-1.$$

2.2. *Sturm-Habicht Sequence and Related Sequences*

Let $p(x) = a_n x^n + a_{n-1}x^{n-1} + \cdots + a_0$ and $q(x) = b_m x^m + b_{m-1}x^{m-1} + \cdots + b_0$ be two real polynomials with $n = \deg(p) > m = \deg(q)$.

In this section, we introduce the concept of subresultant sequence which comes from Sylvester (1853) and Collins (1967), the concept of Sturm-Habicht sequence which was first introduced by Habicht (1948) and extensively studied by Gonzalez-Vega, Lombardi, Recio & Roy (1998), and the concept of signed subresultant sequence which was extensively studied by Lombardi, Roy & Safely el Din (2000) and Lickteig & Roy (2001). We also discuss the relationship between each of them and the multiple factor sequence.

Definition 8. For $0 \le j \le m$, the jth subresultant associated to p and q is

$$\mathrm{Sres}_j(p, n, q, m) = \sum_{k=0}^{j} M_k^j(p, q)x^k$$

where each $M_k^j(p, q)$ is the determinant of the submatrix built by selecting columns $1, 2, \ldots, n + m - 2j - 1$ and column $n + m - k - j$ from the matrix

$$\mathrm{m}_j(p, n, q, m) = \begin{pmatrix} a_n & \cdots & \cdots & \cdots & a_0 & & & & \\ & \ddots & & & & \ddots & & & \\ & & & a_n & \cdots & \cdots & \cdots & a_0 & \\ b_m & \cdots & \cdots & \cdots & b_0 & & & & \\ & \ddots & & & & \ddots & & & \\ & & & b_m & \cdots & \cdots & \cdots & b_0 & \end{pmatrix}.$$

This matrix has dimensions $(n + m - 2j) \times (n + m - j)$ and its rows are the polynomials $x^{m-j-1}p, \ldots, xp, p, x^{n-j-1}q, \ldots, xq, q$ expressed in the basis $x^{n+m-j-1}, \ldots, x, 1$. The determinant $M_j^j(p, q)$ is called the jth principal subresultant coefficient and is denoted by $\mathrm{sres}_j(p, n, q, m)$.

Remark 9. The relationship between the discriminant sequence (Definition 3) and the principal subresultant coefficients of $p(x)$ and $p'(x)$ is as follows. For $1 \le j \le n$,

$$D_j = (-1)^{j(j-1)/2} a_n \,\mathrm{sres}_{n-j}(p, n, p', n - 1).$$

Consequently, the relationship between the multiple factor sequence (Definition 7) of $p(x)$ and the subresultant sequence of $p(x)$ and $p'(x)$ is as follows. For $0 \le j \le n - 1$,

$$\Theta_j(p) = (-1)^{(n-j)(n-j-1)/2} a_n \,\mathrm{Sres}_j(p, n, p', n - 1).$$

Definition 10. The Sturm-Habicht sequence associated to p and q is defined as the sequence of polynomials $\{\mathrm{StHa}_j(p, q)\}_{j=0,1,\ldots,v+1}$ where $v = n+m-1$, $\mathrm{StHa}_{v+1}(p, q) = p$, $\mathrm{StHa}_v(p, q) = p'q$ and for $0 \le j \le v - 1$,

$$\mathrm{StHa}_j(p, q) = (-1)^{(v-j)(v-j+1)/2}\,\mathrm{Sres}_j(p, v + 1, p'q, v).$$

For $0 \le j \le v + 1$, the jth principal Sturm-Habicht coefficient $\mathrm{stha}_j(p, q)$ is defined as the coefficient of x^j in $\mathrm{StHa}_j(p, q)$.

Remark 11. The relationship between the discriminant sequence $[D_1, \ldots, D_n]$ of p and the principal Sturm-Habicht coefficients of p and 1 is: $D_j = a_n \operatorname{stha}_{n-j}(p, 1)$ $(1 \le j \le n)$.

Definition 12. For $0 \le j \le m$, the jth Sylvester-Habicht matrix $\operatorname{SyHa}_j(p, q)$ of $p(x)$ and $q(x)$ is the $(n + m - 2j) \times (n + m - j)$ matrix

$$\operatorname{SyHa}_j(p,q) = \begin{pmatrix} a_n & \cdots & \cdots & \cdots & \cdots & a_0 & 0 & 0 \\ 0 & \ddots & & & & & \ddots & 0 \\ \vdots & \ddots & a_n & \cdots & \cdots & \cdots & \cdots & a_0 \\ \vdots & & 0 & b_m & \cdots & \cdots & \cdots & b_0 \\ \vdots & \iddots & \iddots & & & & \iddots & 0 \\ 0 & \iddots & & & & \iddots & \iddots & \vdots \\ b_m & \cdots & \cdots & \cdots & b_0 & 0 & \cdots & 0 \end{pmatrix}.$$

The rows are the polynomials $x^{m-j-1}p, \ldots, xp, p, q, xq, \ldots, x^{n-j-1}q$ expressed in the basis $x^{n+m-j-1}, \ldots, x, 1$. If we replace the matrix $\mathrm{m}_j(p, n, q, m)$ by $\operatorname{SyHa}_j(p, q)$ in Definition 8, then the jth subresultant associated to p and q is called the jth signed subresultant.

The jth signed subresultant coefficient $\operatorname{sRes}_j(p, q)$ is the determinant of $\operatorname{SyHa}_{j,j}(p, q)$ obtained by taking the first $n+m-2j$ columns of $\operatorname{SyHa}_j(p, q)$. By convention, we extend these definitions for $m < j < n$ by $\operatorname{sRes}_j(p, q) = 0$ and $\operatorname{sRes}_n(p, q) = \operatorname{sgn} a_n$.

Remark 13. The relationship between the discriminant sequence $[D_1, \ldots, D_n]$ of p and the signed subresultant coefficients of p and p' is: $D_j = a_n \operatorname{sRes}_{n-j}(p, p')$ $(1 \le j \le n)$.

From the definitions and remarks above we see that, up to a constant factor and a sign, the multiple factor sequence of p, the subresultant sequence of p and p', the signed subresultant sequence of p and p', and the Sturm-Habicht sequence of p and 1 are all the same. When p is a polynomial with constant coefficients, these sequences can be computed efficiently by a subresultant algorithm in Lombardi, Roy & Safely el Din (2000). However, the algorithm is not so efficient when p has a lot of parameters (Abdeljaoued, Diaz-Toca & Gonzalez-Vega, 2004).

3. Review of Existing Work

In view of the previous section, we shall mostly discuss existing work with respect to the algorithms in Yang, Hou & Zeng (1996) and Liang & Jeffrey (2006), but for some points Gonzalez-Vega (1998) is more explicit and detailed, and we shall refer to that work also.

3.1. Polynomial with Real, Non-symbolic, Coefficients

For a polynomial $p \in \mathbb{R}[x]$, Yang, Hou & Zeng (1996) gave an algorithm for obtaining the root classification (RC) of p based on the following propositions, where $\Delta^j(p), j = 0, 1, 2, \ldots$ is the Δ-sequence of p.

Proposition 14. *Let $p \in \mathbb{R}[x]$ have revised sign list* $\operatorname{rsl}(p)$*. If the number of non-vanishing elements in* $\operatorname{rsl}(p)$ *is s, and the number of sign changes in* $\operatorname{rsl}(p)$ *is v, then $p(x)$ has v pairs of distinct complex conjugate roots and $s - 2v$ distinct real roots.*

Proposition 15. *If $\Delta^j(p)$ has k distinct roots with respective multiplicities $n_1, n_2, \ldots, n_k$, then $\Delta^{j+1}(p)$ has at most k distinct roots with respective multiplicities $n_1 - 1, n_2 - 1, \ldots, n_k - 1$.*

Proposition 16. *If $\Delta^j(p)$ has k distinct roots with respective multiplicities $n_1, n_2, \ldots, n_k$, and $\Delta^{j-1}(p)$ has m distinct roots, then $m \geq k$, and the multiplicities of these m distinct roots are $n_1 + 1, n_2 + 1 \ldots, n_k + 1, 1, \ldots, 1$ respectively.*

The algorithm uses these propositions to count the number of roots of p, and then if necessary to count the roots of each relevant $\Delta^j(p)$ (until there are no multiplicities). When a polynomial has symbolic coefficients, however, the algorithm needs to be modified.

3.2. Polynomial with Symbolic Coefficients

For a parametric polynomial $p \in \mathbb{R}[a_0, a_1, \ldots, a_n][x]$, we first note that the Δ-sequence of p is difficult to compute directly, because if a standard GCD function is applied to a parametric polynomial the function will in general give $\gcd(p, p') = 1$. However, when the coefficients are specialized, the $\gcd(p, p')$ might not be equal to 1. Therefore the multiple factor sequence or its equivalents must be used.

At this point, the description in Gonzalez-Vega (1998) is clearer. Suppose the polynomial p has discriminant sequence

$$[D_1, D_2, \ldots, D_n] \ .$$

For each D_i in the sequence that contains symbolic terms, we assign combinatorially the possible values $+1$, 0, and -1. In principle, this could give 3^{n-1} cases (the first entry of a sign list is always 1). For each case, a revised sign list is constructed and the number of roots for this case is determined. To determine the multiplicities of the roots, the combinatorial procedure is repeated for each entry in the Δ-sequence.

3.3. An Example

The following example will be used at several places in the exposition below.

Example 17. We consider the real parametric polynomial

$$p(x) = x^6 + ax^2 + bx + c \ . \tag{2}$$

Its sign list is

$$[1, 0, 0, \operatorname{sgn} D_4, \operatorname{sgn} D_5, \operatorname{sgn} D_6] \tag{3}$$

where

$$\begin{aligned} D_4 &= a^3 \ , \qquad D_5 = 256a^5 + 1728a^2c^2 - 5400ab^2c + 1875b^4 \ , \\ D_6 &= -1024a^6c + 256a^5b^2 - 13824a^3c^3 + 43200a^2b^2c^2 - 22500ab^4c \\ &\qquad +3125b^6 - 46656c^5 \ . \end{aligned}$$

Since each of D_4, D_5, D_6 can be positive, zero or negative, there are 27 possible realizations of this sign list to examine. Let us consider the following example sign list: the case $D_4 < 0, D_5 = 0, D_6 = 0$. For this case, the sign list becomes $[1, 0, 0, -1, 0, 0]$, implying a revised sign list $[1, -1, -1, -1, 0, 0]$. Then by Proposition 14, $p(x)$ has one pair of distinct complex conjugate roots and two distinct real roots. For finding the multiplicities, the GCD is obtained from the multiple factor sequence as $\Delta^1(p) = \Theta_2(p) = 4ax^2 + 5bx + 6c$. The discriminant sequence for this polynomial is $[1, E_2]$, where $E_2 = 25b^2 - 96ac$. If $E_2 > 0$, then again using Proposition 14, $\Delta^1(p)$ has two distinct real roots. Therefore, we conclude that the case $D_4 < 0, D_5 = D_6 = 0, E_2 > 0$ implies that $p(x)$ has one pair of complex conjugate roots of multiplicity 1 and two real roots of multiplicities 2. If $E_2 < 0$, then $\Delta^1(p)$ has a pair of complex conjugate roots. So we conclude that for this case $p(x)$ has one pair of complex conjugate roots of multiplicity 2 and two real roots of multiplicities 1. Similarly, if $E_2 = 0$, then $p(x)$ has one pair of complex conjugate roots of multiplicity 1 and two real roots of respective multiplicities 1 and 3. This analysis could be repeated for each case.

There are 27 possible cases of the sign list in this example, but we show below that there are only 10 cases of RC in the CRC (further details of this problem are given below as Example 27). Therefore, there remains the work of condensing the 27 cases of sign lists into just 10 cases. This is done by *ad hoc* analysis, as noted by Gonzalez-Vega (1998).

3.4. Automatic Computation of CRC

An algorithm for the automatic computation of CRCs was described in Liang & Zhang (1999) and Liang & Jeffrey (2006). Although based on the propositions above, it followed a different direction. We first notice the following facts. For a general parametric polynomial of degree n (i.e. one in which all coefficients are present and symbolic), the initial number of cases in its sign list that must be examined is 3^{n-1} (see Section 3.2). Some of these cases will have subcases. In contrast, the number of entries in the CRC of a polynomial of degree n is

$$\sum_{k=0}^{\lfloor n/2 \rfloor} \mathfrak{P}(n-2k)\,\mathfrak{P}(k) < \exp\left(\pi\sqrt{2n/3}\right)/4\sqrt{3}\ ,$$

where $\mathfrak{P}(k)$ denotes the number of partitions of the integer k. The upper bound uses the well-known asymptotic result for $\mathfrak{P}$.

Not only is the number of members in a CRC less than the number of cases of a sign list, but in many applications, not all members of a CRC are needed. For example, in the applications to positive definiteness given below, only those RCs in a CRC with no real roots are needed. Therefore it is more efficient computationally to approach the calculations differently. The approach of Liang & Zhang (1999) and Liang & Jeffrey (2006) starts by generating all required RC members of the CRC. Then for each RC, the conditions that the coefficients of the polynomial should satisfy are found.

3.5. Need for Improvement

We identify the points in the existing algorithms where improvements are made here. The first point is the use of revised sign lists. This is a major source of inefficiency, because conditions expressed in terms of revised sign lists have until now been transferred to conditions in terms of sign lists manually. Definition 5 is equivalent to defining a mapping Φ from a sign list to a revised sign list. Therefore the existing algorithms require the inverse mapping Φ^{-1}. However, Φ is not injective, so Φ^{-1} is multivalued, and more importantly is difficult to compute.

As an example, consider the polynomial $p_6 := x^6 + ax^2 + bx + c$, whose discriminant sequence was given in (3). One condition (among many) for p_6 having no real roots is that its revised sign list be $[1, -1, -1, 1, -1, -1]$. According to the special structure of the discriminant sequence of p_6, we have

$$\Phi^{-1}[1, -1, -1, 1, -1, -1] = \{[1, 0, 0, 1, -1, -1], [1, 0, 0, 1, 0, -1], [1, 0, 0, 0, -1, -1]\} \ .$$

Therefore, the given condition is transferred to the following:

$$[D_4 > 0 \wedge D_5 < 0 \wedge D_6 < 0] \vee [D_4 > 0 \wedge D_5 = 0 \wedge D_6 < 0] \vee [D_4 = 0 \wedge D_5 < 0 \wedge D_6 < 0].$$

This case, already cumbersome, is none the less relatively simple because of the nature of the polynomial. However, if the polynomial were a general parametric polynomial, it would be very difficult to find $\Phi^{-1}[1, -1, -1, 1, -1, -1]$, and of course more so for higher degrees. Consequently, it would be a great improvement to avoid revised sign lists.

The second point concerns the realizability of the conditions obtained by the inverse mapping Φ^{-1}. We continue with the example of $x^6 + ax^2 + bx + c$.

Example 18. For the case of no real roots, one condition given above is

$$D_4 = 0 \wedge D_5 < 0 \wedge D_6 < 0 \ , \tag{4}$$

with the D_k given in (3).

We assert that this condition is not realizable. Since $D_4 = 0$, then $a = 0$, and then $D_5 = 1875b^4 \geq 0$. This is a contradiction. So non-realizable conditions are included in the output of the existing algorithm, and no mechanism was offered to detect them.

In summary, although the old CRC algorithms give correct results, the computations are difficult because we have to transfer conditions on revised sign lists to conditions on sign lists, and the results may contain non-realizable conditions. These weaknesses provide the motivation of the current paper.

4. Basis of Algorithm

We now continue to the basis of the new algorithm. The following definitions can be found in Basu, Pollack & Roy (2003).

Definition 19 (TaQ)**.** Let $p(x), q(x)$ be two polynomials in $\mathbb{R}[x]$. The Tarski Query of q for p in $\mathbb{R}$ is the number

$$\mathrm{TaQ}(q, p) = \#\{\alpha \in \mathbb{R} | p(\alpha) = 0 \wedge q(\alpha) > 0\} - \#\{\alpha \in \mathbb{R} | p(\alpha) = 0 \wedge q(\alpha) < 0\} \ .$$

Definition 20 (PmV)**.** Let $s = [s_n, \ldots, s_0]$ be a finite list of elements in $\mathbb{R}$ such that $s_n \neq 0$. Let $m < n$ be such that $s_{n-1} = \cdots = s_{m+1} = 0$, and $s_m \neq 0$, and $s' = [s_m, \ldots, s_0]$. If there is no such m, then s' is the empty list. We define inductively

$$\mathrm{PmV}(s) = \begin{cases} 0 & \text{if } \ s' = \emptyset, \\ \mathrm{PmV}(s') + \epsilon_{n-m} \operatorname{sgn}(s_n s_m) & \text{if } n - m \text{ odd}, \\ \mathrm{PmV}(s') & \text{if } n - m \text{ even}, \end{cases}$$

where $\epsilon_{n-m} = (-1)^{(n-m)(n-m-1)/2}$.

The main theorem for the improved CRC algorithm requires the following lemmas which can be found in Basu, Pollack & Roy (2003). Let $\mathrm{Ind}(q/p)$ be the Cauchy index of q/p on $\mathbb{R}$.

Lemma 21. *Given two polynomials $p(x), q(x)$ in $\mathbb{R}[x]$, we have* $\mathrm{TaQ}(q,p) = \mathrm{Ind}(p'q/p)$.

Lemma 22. *Let $p(x), q(x)$ be the two polynomials in Section 2.2. We have*

$$\mathrm{PmV}([\mathrm{sRes}_n(p,q), \mathrm{sRes}_{n-1}(p,q), \ldots, \mathrm{sRes}_0(p,q)]) = \mathrm{Ind}(q/p)\ .$$

The main theorem is the following

Theorem 23. *Let $D = [D_1, \ldots, D_n]$ be the discriminant sequence of a real polynomial $p(x)$ of degree n, and let ℓ be the maximal subscript such that $D_\ell \neq 0$. If* $\mathrm{PmV}(D) = r$, *then $p(x)$ has $r+1$ distinct real roots and $\frac{1}{2}(\ell - r - 1)$ pairs of distinct complex conjugate roots.*

Proof. We first prove that

$$\#\{\alpha \in \mathbb{R} | p(\alpha) = 0\} = \mathrm{PmV}([D_1, \ldots, D_n]) + 1\ .$$

Observe that $\#\{\alpha \in \mathbb{R} | p(\alpha) = 0\} = \mathrm{TaQ}(1,p)$. Then from Lemma 21, we have $\mathrm{TaQ}(1,p) = \mathrm{Ind}(p'/p)$. By Lemma 22,

$$\mathrm{Ind}(p'/p) = \mathrm{PmV}([\mathrm{sRes}_n(p,p'), \mathrm{sRes}_{n-1}(p,p'), \ldots, \mathrm{sRes}_0(p,p')])\ .$$

By Remark 13,

$$\begin{aligned} &\mathrm{PmV}([\mathrm{sRes}_n(p,p'), \mathrm{sRes}_{n-1}(p,p'), \ldots, \mathrm{sRes}_0(p,p')]) \\ &\quad = \mathrm{PmV}([\mathrm{sgn}(a_n), D_1/a_n, \ldots, D_n/a_n]) = 1 + \mathrm{PmV}([D_1/a_n, \ldots, D_n/a_n])\ , \end{aligned}$$

since $\mathrm{sgn}(a_n)$ and $D_1/a_n = na_n$ have the same sign. Finally,

$$1 + \mathrm{PmV}([D_1/a_n, \ldots, D_n/a_n]) = 1 + \mathrm{PmV}([D_1, \ldots, D_n])\ .$$

Now noticing that $p(x)$ has ℓ distinct roots, the last part of the theorem follows. □

From Theorem 23, we obtain the following important corollary, which is necessary for detecting non-realizable sign lists in the output conditions.

Corollary 24. *Let $S = [s_1, \ldots, s_n]$ and $R = [r_1, \ldots, r_n]$ be the sign list and the revised sign list of $p(x)$ respectively. Then* $\mathrm{PmV}(S) = \mathrm{PmV}(R)$.

Proof. Let ℓ be the maximal subscript of S such that $s_\ell \neq 0$. Let m and ν be the number of sign permanences and the number of sign changes of R. By Theorem 23, we have $\mathrm{PmV}(S) = \#\{\alpha \in \mathbb{R} | p(\alpha) = 0\} - 1$. By Proposition 14, $\#\{\alpha \in \mathbb{R} | p(\alpha) = 0\} = \ell - 2\nu$. So

$$\mathrm{PmV}(S) = \ell - 2\nu - 1\ . \tag{5}$$

On the other hand, it is easy to see that $\mathrm{PmV}(R) = m - \nu$ and $\ell - 1 = m + \nu$. So

$$2\nu = \ell - 1 - \mathrm{PmV}(R)\ . \tag{6}$$

Therefore, from (5) and (6), we get $\mathrm{PmV}(S) = \ell - (\ell - 1 - \mathrm{PmV}(R)) - 1 = \mathrm{PmV}(R)$. □

Example 25. We give an example of the use of the above corollary, by proving the non-realizability of condition (4) from a different point of view. The condition is equivalent to the sign list $[1, 0, 0, 0, -1, -1]$, which has revised sign list $[1, -1, -1, 1, -1, -1]$. Since

$$\mathrm{PmV}([1, 0, 0, 0, -1, -1]) = 1 \neq -1 = \mathrm{PmV}([1, -1, -1, 1, -1, -1]),$$

then Corollary 24 states that the sign list $[1, 0, 0, 0, -1, -1]$ is not realizable for p_6. Similarly we can prove that the sign list $[1, 0, 0, -1, 0, -1]$ is not realizable for p_6, because

$$\mathrm{PmV}([1, 0, 0, -1, 0, -1]) = 1 \neq -1 = \mathrm{PmV}([1, -1, -1, -1, 1, -1]) \ .$$

Observe that, by Proposition 14, $\mathrm{rsl}(p_6) = [1, -1, -1, -1, 1, -1]$ is one condition for p_6 having no real roots, and $[1, 0, 0, -1, 0, -1] \in \Phi^{-1}[1, -1, -1, -1, 1, -1]$, where Φ is the mapping from sign lists to revised sign lists (Section 3.5). This example shows again that the process for transferring the output conditions on revised sign lists to conditions on sign lists may include non-realizable conditions.

Example 26. Let $P_4 = x^4 + a_3x^3 + a_2x^2 + a_1x + a_0$ and s_j denote the jth principal Sturm-Habicht coefficient $\mathrm{stha}_j(P_4, 1)$ (Definition 10). Gonzalez-Vega (1998) used his combinatorial algorithm to derive the conditions for $(\forall x\ P_4 > 0)$. In order to reduce the output conditions, he removed the case when $s_2 = 0$ and $s_1 > 0$ by an ad hoc argument.

In the following, we use Corollary 24 to detect this non-realizable condition. By Remark 11, noticing $a_4 = 1$, we have $[s_3, s_2, s_1, s_0] = [D_1, D_2, D_3, D_4]$. So the case is equivalent to the sign list $[1, 0, 1, \operatorname{sgn} D_4]$. By Corollary 24, it is not realizable since its revised sign list is $[1, -1, 1, \operatorname{sgn} D_4]$ and

$$\mathrm{PmV}([1, 0, 1, \operatorname{sgn} D_4]) = \mathrm{PmV}([1, -1, 1, \operatorname{sgn} D_4]) + 2.$$

5. Algorithm

In the first part of this section, we propose an improved algorithm for computing the CRC of a real parametric polynomial. In the second part, we summarize some features of the algorithm. Examples for explaining the algorithm will be given in Example 27 below.

5.1. Algorithm Description

We need the following functions to present the algorithm.

- `AllListsReal`: Input $n \in \mathbb{N}$; output the list of all possible RCs for a real parametric polynomial of degree n. See Liang & Jeffrey (2006) for details.
- `RCInfo`: Input an RC L of a polynomial with real coefficients; output the list $[n, \ell, r]$, where n is the degree of the polynomial, ℓ the number of distinct (real and complex) roots, and r the number of distinct real roots specified by L.
- `MaxSubs`: Input a sign list S; output the maximal subscript of non-vanishing elements in S.
- `MinusOne`: Input an RC L; output the RC generated from L by decreasing the absolute values of all elements in L by 1, and then erasing all elements of value 0.
- `Op`: Input a set $\{a_1, \ldots, a_n\}$; output the sequence $a_1, \ldots, a_n$.

Let $p(x) = a_n x^n + \ldots + a_1 x + a_0$ be a real parametric polynomial with $a_n \neq 0$. The algorithm starts from generating all possible RCs for $p(x)$ using `AllListsReal`. Then for each RC L, we find the conditions on the parametric coefficients of $p(x)$ such that L is realized.

We first compute all possible sign lists of $p(x)$ for $p(x)$ having L as its RC.

Algorithm 1. GenAllSL
Input: A real parametric polynomial $p(x)$ and an RC L.
Output: The set of all the sign lists of $p(x)$ that lead to the RC given by L.
Procedure:

- $[n, \ell, r] \leftarrow \texttt{RCInfo}(L)$.
- Compute the discriminant sequence $D = [D_1, \ldots, D_n]$ of p.
- Compute the set S_0 of all possible sign lists from D: for $1 < k \leq n$, if $D_k \in \mathbb{R}$, then $D_k \to \mathrm{sgn}(D_k)$; otherwise, $D_k \to \{-1, 0, 1\}$. For example, if $D = [1, -2, a]$, then $S_0 = \{[1, -1, -1], [1, -1, 0], [1, -1, 1]\}$.
- Compute $S = \{s \in S_0 | \, \texttt{MaxSubs}(s) = \ell, \mathrm{PmV}(s) = \mathrm{PmV}(\mathrm{rsl}(s)) = r - 1\}$,
- Return S.

Then $S = \texttt{GenAllSL}(p, L)$ is the set of all possible sign lists of $p(x)$ for $p(x)$ having L as its RC. In order to make the multiplicities of the roots of $p(x)$ be those specified by L, we also have to determine the possible sign lists of the polynomials in the $\Delta-$sequence of $p(x)$ (Definition 6), except for the following five cases (*termination conditions*): if the RC of $p(x)$ is L and is such that $[n, \ell, r] = \texttt{RCInfo}(L)$, then these cases are

(1) $n = \ell$,
(2) $\ell = 1$.
(3) $\ell = 2$ and $r = 0$.
(4) $n - \ell = 1$.
(5) $r = 0$ and $n - \ell = 2$.

For other cases, $\Delta^1(p) = \Theta_{n-\ell}(p)$, the $(n - \ell)$th multiple factor of $p(x)$ (Definition 7). By Proposition 15, the RC of $\Delta^1(p)$ would be $L_1 = \texttt{MinusOne}(L)$. Then we can call `GenAllSL` recursively. This is the basis of the following algorithm which generates the conditions for $p(x)$ having L as its root classification. The output conditions are a sequence of *mixed lists*. Each mixed list consists of a polynomial in the Δ-sequence of $p(x)$, followed by all of its possible sign lists. We denote the empty sequence by NULL. Notice that if NULL is returned, then L is not realizable.

Algorithm 2. Cond
Input: a real parametric polynomial $p(x)$; an RC L.
Output: A sequence of mixed lists (the conditions for $p(x)$ having L as its RC).
Procedure:
$[n, \ell, r] \leftarrow \texttt{RCInfo}(L)$
$S \leftarrow \texttt{GenAllSL}(p, L)$
if $S = \emptyset$
    return NULL
else if $[n, \ell, r]$ meets one of the five cases
    return $[p, \texttt{Op}(S)]$
else

$C \leftarrow \texttt{Cond}(\Delta^1(p), \texttt{MinusOne}(L))$
if C = NULL
    return NULL
else
    return $[p, \texttt{Op}(S)], C$

Proof. It is easy to see that the number of recursions in the algorithm is bounded by $\deg(p) - 1$, so the algorithm will terminate in finite steps.

Now suppose that $L_0 := L$ is the RC of $\Delta^0(p) := p(x)$, and the number of recursions is m. Let $[n, \ell, r] = \texttt{RCInfo}(L_0)$, then $\Delta^0(p)$ would be a polynomial of degree d, with ℓ distinct roots and r distinct real roots. So by Theorem 23, the sign list of $\Delta^0(p)$ should belong to $S_0 := \texttt{GenAllSL}(\Delta^0(p), L_0)$. Let $L_1 = \texttt{MinusOne}(L_0)$. Then, as we pointed out above, the RC of $\Delta^1(p)$ is L_1. So similarly, we can conclude that the sign list of $\Delta^1(p)$ should belong to $S_1 := \texttt{GenAllSL}(\Delta^1(p), L_1)\ldots$ The sign list of $\Delta^m(p)$ should belong to $S_m := \texttt{GenAllSL}(\Delta^m(p), L_m)$. For any $k (0 \leq k \leq m)$, if $S_k = \emptyset$, then L_k is not realizable for $\Delta^k(p)$, so L is not realizable for p.

On the other hand, if the sign list of $\Delta^j(p)$ belongs to $S_j := \texttt{GenAllSL}(\Delta^j(p), L_j)$ $(0 \leq j \leq m)$, then by Propositions 15, 16, and Theorem 23, using similar discussion as above, we can conclude that the RC of $p(x)$ is L. □

In summary, the improved algorithm for generating the CRC of a real parametric polynomial is the following

Algorithm 3. CRC
Input: A real parametric polynomial $p(x)$.
Output: the CRC of $p(x)$.
Procedure:
$\mathcal{L} \leftarrow \texttt{AllListsReal}(\deg(p))$
for L in $\mathcal{L}$ do
    if $\texttt{Cond}(p, L) \neq$ NULL
        output L and $\texttt{Cond}(p, L)$

5.2. Algorithm Summary

In comparison with the old CRC algorithm (Liang & Jeffrey, 2006), the improved algorithm has the following advantages.

- It uses sign lists instead of revised sign lists in the output conditions, which makes the computation of CRC more efficient. Otherwise, one has to transfer the output conditions in terms of revised sign lists to conditions in terms of sign lists. The transferring process is usually difficult and full of opportunities for including non-realizable conditions.
- It uses Corollary 24 to detect and delete non-realizable conditions, and consequently reduces the size of the output conditions significantly.

  If $p(x)$ is a general parametric polynomial of degree n (see Section 3.4), then the number of all possible sign lists of $p(x)$ is 3^{n-1}. From Table 1 we can see that, as n increases, more and more sign lists becomes non-realizable and are detected by Corollary 24.

Table 1. Numbers of Non-realizable Sign Lists Detected by Corollary 24

Degree n	2	3	4	5	6	7	8	9	10	11
3^{n-1}	3	9	27	81	243	729	2187	6561	19683	59049
Detected	0	1	5	21	79	281	963	3217	10547	34089

- It takes advantage of any sparsity in $p(x)$. For example, the polynomial $x^{10}+ax^2+bx+c$ in Example 30 has the discriminant sequence $D = [1,0,0,0,0,0,0,a^7,D_9,D_{10}]$. By using its sparsity, one only need consider $3^3 = 27$ sign lists for $p(x)$, while in the old algorithm, one must consider $3^9 = 19683$ sign lists for $p(x)$.

On the other hand, although Corollary 24 is used in the algorithm to filter non-realizable sign lists, it is not guaranteed that all non-realizable sign lists are detected and deleted.

6. Examples

The algorithm has been implemented in MAPLE. In the following, we first demonstrate the output of the MAPLE program, then as a by-product we show applications to some problems in real quantifier elimination. A lot of work has been done in this field, for example, Hong (1993), Heintz, Roy & Solerno (1993) and Gonzalez-Vega (1998).

All computations were performed with Maple 10 running on a 1.6 GHz Pentium CPU. All times were less than 2 seconds and therefore are not reported.

Example 27. Find the CRC of the polynomial $p_6 = x^6 + ax^2 + bx + c$.

Here is the MAPLE output.

```
(*) p6:=x^6+a*x^2+b*x+c
The Complete Root Classification of p6 is:
(The condition format is: [poly, all possible sign lists])
(1) [[6],[]], if and only if
      [p6, [1,0,0,0,0,0]]
(2) [[1,1,1,1],[1,-1]], if and only if
      [p6, [1,0,0,-1,-1,-1]]
(3) [[1,1,2],[1,-1]], if and only if
      [p6, [1,0,0,-1,-1,0]]
(4) [[1,3],[1,-1]], if and only if
      [p6, [1,0,0,-1,0,0]], [p62, [1,0]]
(5) [[2,2],[1,-1]], if and only if
      [p6, [1,0,0,-1,0,0]], [p62, [1,1]]
(6) [[1,1],[2,-2]], if and only if
      [p6, [1,0,0,-1,0,0]], [p62, [1,-1]]
(7) [[1,1],[1,-1,1,-1]], if and only if
      [p6, [1,0,0,0,0,1],[1,0,0,-1,-1,1],[1,0,0,-1,0,1],
        [1,0,0,1,1,1],[1,0,0,0,1,1],[1,0,0,-1,1,1]]
(8) [[2],[1,-1,1,-1]], if and only if
      [p6, [1,0,0,1,1,0],[1,0,0,0,1,0],[1,0,0,-1,1,0]]
(9) [[],[1,-1,2,-2]], if and only if
```

```
      [p6, [1,0,0,1,0,0]]
(10) [[],[1,-1,1,-1,1,-1]], if and only if
      [p6, [1,0,0,0,1,-1],[1,0,0,-1,1,-1],[1,0,0,1,0,-1],
        [1,0,0,0,0,-1],[1,0,0,1,-1,-1],[1,0,0,1,1,-1]]
where
(#1) p6:=x^6+a*x^2+b*x+c,
and its discriminant sequence is:
[1, 0, 0, a^3, 256*a^5+1728*c^2*a^2-5400*a*c*b^2+1875*b^4,
-1024*a^6*c+256*a^5*b^2-13824*c^3*a^3+43200*c^2*a^2*b^2
-22500*b^4*c*a+3125*b^6-46656*c^5]
(#2) p62:=4*a*x^2+5*b*x+6*c,
and its discriminant sequence is:
 [1, 25*b^2-96*a*c]
```

Let us explain the CRC of p_6 with respect to the improved algorithm. First, the algorithm `CRC` calls the function `AllListsReal` to generate all possible root classifications (RCs) for a polynomial of degree 6. There are 23 RCs as follows. For the sake of simplicity, the order of them has been changed.

```
[ [[3,3],[]], [[2,4],[]], [[2,2,2],[]], [[1,5],[]], [[1, 2,3],[]],
[[1,1,4],[]], [[1,1,2,2],[]], [[1,1,1,3],[]], [[1,1,1,1,2],[]],
[[1,1,1,1,1,1],[]], [[4],[1,-1]], [[2],[2,-2]], [[],[3,-3]],
[[6],[]], [[1,1,1,1],[1,-1]], [[1,1,2],[1,-1]], [[1,3],[1,-1]],
[[2,2],[1,-1]], [[1,1],[2,-2]], [[1,1],[1,-1,1,-1]], [[2],
[1,-1,1,-1]], [[],[1,-1,2,-2]], [[],[1,-1,1,-1,1,-1]] ]
```

Second, in a "for-loop", for each RC L above and p_6, the algorithm `Cond` is called to generate the conditions for p_6 having L as its RC. It turns out that, for the first 13 RCs (those RCs in the first 3 lines), the outputs of `Cond` are all NULL (empty sequence). So the first 13 RCs are not realizable for p_6. On the other hand, the rest 10 RCs are realizable for p_6. There are 10 numbered lines in the output of the CRC algorithm, and each line represents an RC and the conditions such that the RC is realized for p_6.

For example, the first line describes that p_6 has a single real root of multiplicity 6, if and only if p_6 has the sign list $[1, 0, 0, 0, 0, 0]$. Since the discriminant sequence of p_6 is $[1, 0, 0, a^3, D_5, D_6]$ (see the Maple output above), where

$$D_5 = 256a^5 + 1728a^2c^2 - 5400ab^2c + 1875b^4 \ ,$$
$$D_6 = -1024a^6c + 256a^5b^2 - 13824a^3c^3 + 43200a^2b^2c^2 - 22500ab^4c + 3125b^6 - 46656c^5 \ ,$$

p_6 has a single real root of multiplicity 6, if and only if $a^3 = 0 \wedge D_5 = 0 \wedge D_6 = 0$. The latter is equivalent to $a = b = c = 0$. In summary, p_6 has a single real root of multiplicity 6, if and only if $a = b = c = 0$, which is what we expect.

Now we consider line (5). It describes that p_6 has 2 real roots, each of multiplicity 2, and one complex conjugate pair of multiplicity 1, if and only if p_6 has the sign list $[1, 0, 0, -1, 0, 0]$ and p_{62} has the sign list $[1, 1]$. The algorithm `Cond` works as follows. Let $L = [[2, 2], [1, -1]]$. First, it calls the function `RCInfo`(L) and gets the information about L: $[n, \ell, r] = [6, 4, 2]$; then it calls `GenAllSL`(p_6, L) to generate the set S of all possible sign lists of p_6 for p_6 having L as its RC, and it turns out that $S = \{[1, 0, 0, -1, 0, 0]\}$. Because $S \neq \emptyset$ and $[n, \ell, r]$ does not meet the termination conditions, `Cond` also has to compute all possible sign lists of $\Delta^1(p_6)$ which is p_{62} above, for $\Delta^1(p_6)$ having $\texttt{MinusOne}(L) = [[1, 1], []]$ as its RC. So `Cond` calls itself again: $\texttt{Cond}(p_{62}, [[1, 1], []])$. It turns out that p_{62} has $[[1, 1], []]$ as its RC, if and only if p_{62} has the sign list $[1, 1]$. At this point, the termination condition (1) is reached, and the algorithm `Cond` terminates.

Therefore, p_6 has 2 real roots, each of multiplicity 2, and one complex conjugate pair of multiplicity 1, if and only if p_6 has the sign list $[1, 0, 0, -1, 0, 0]$ and p_{62} has the sign list $[1, 1]$. The condition can be expressed as $a^3 < 0 \wedge D_5 = 0 \wedge D_6 = 0 \wedge E_2 > 0$, where E_2 is the second element of the discriminant sequence of p_{62} (see the Maple output above)

$$E = [1, E_2], \qquad E_2 = 25b^2 - 96ac.$$

In summary, p_6 has 2 real roots, each of multiplicity 2, and one complex conjugate pair of multiplicity 1, if and only if $a < 0 \wedge D_5 = 0 \wedge D_6 = 0 \wedge E_2 > 0$.

Next we consider line (8). It states that p_6 has one real root of multiplicity 2, and two pairs of complex conjugate roots, each of multiplicity 1, if and only if the sign list of p_6 be one of $[1, 0, 0, 1, 1, 0], [1, 0, 0, 0, 1, 0]$ or $[1, 0, 0, -1, 1, 0]$. The latter can be expressed as $[a > 0 \wedge D_5 > 0 \wedge D_6 = 0] \vee [a = 0 \wedge D_5 > 0 \wedge D_6 = 0] \vee [a < 0 \wedge D_5 > 0 \wedge D_6 = 0]$. It can be further simplified as $D_5 > 0 \wedge D_6 = 0$. Therefore p_6 has one real root of multiplicity 2, and two pairs of complex conjugate roots, each of multiplicity 1, if and only if $D_5 > 0 \wedge D_6 = 0$.

Other lines can be explained similarly. Finally, note that in line (10), the two non-realizable sign lists $[1, 0, 0, 0, -1, -1]$ and $[1, 0, 0, -1, 0, -1]$ have been detected and deleted by the improved algorithm automatically.

Problem 28. Find the CRC of the polynomial $p_6 = x^6 + ax^3 + bx^2 + cx + d$.

The following is the Maple output (where the discriminant sequences of p_6, p_{62} and p_{63} are omitted in order to give a more pleasing layout).

```
(*) p6:=x^6+a*x^3+b*x^2+c*x+d
The Complete Root Classification of p6 is:
(The condition format is: [poly,its all possible sign lists])
(1) [[6],[]], if and only if
      [p6, [1,0,0,0,0,0]]
(2) [[1,1,1,1],[1,-1]], if and only if
      [p6, [1,0,-1,-1,-1,-1],[1,0,0,-1,-1,-1]]
(3) [[1,1,2],[1,-1]], if and only if
      [p6, [1,0,-1,-1,-1,0],[1,0,0,-1,-1,0]]
(4) [[1,3],[1,-1]], if and only if
      [p6, [1,0,-1,-1,0,0],[1,0,0,-1,0,0]], [p62, [1,0]]
(5) [[2,2],[1,-1]], if and only if
      [p6, [1,0,-1,-1,0,0],[1,0,0,-1,0,0]], [p62, [1,1]]
(6) [[4],[1,-1]], if and only if
      [p6, [1,0,-1,0,0,0]], [p63, [1,0,0]]
(7) [[1,1],[2,-2]], if and only if
      [p6, [1,0,-1,-1,0,0],[1,0,0,-1,0,0]], [p62, [1,-1]]
(8) [[1,1],[1,-1,1,-1]], if and only if
```

```
      [p6, [1,0,-1,0,0,1],[1,0,0,0,0,1],[1,0,-1,-1,0,1],
        [1,0,0,-1,0,1],[1,0,-1,-1,-1,1],[1,0,0,-1,-1,1],
        [1,0,-1,1,1,1],[1,0,0,1,1,1],[1,0,-1,0,1,1],
        [1,0,0,0,1,1],[1,0,-1,-1,1,1],[1,0,0,-1,1,1]]
(9) [[2],[2,-2]], if and only if
      [p6, [1,0,-1,0,0,0]], [p63, [1,1,-1],[1,0,-1],[1,-1,-1]]
(10) [[2],[1,-1,1,-1]], if and only if
      [p6, [1,0,-1,1,1,0],[1,0,0,1,1,0],[1,0,-1,0,1,0],
        [1,0,0,0,1,0],[1,0,-1,-1,1,0],[1,0,0,-1,1,0]]
(11) [[],[1,-1,2,-2]], if and only if
      [p6, [1,0,0,1,0,0],[1,0,-1,1,0,0]]
(12) [[],[1,-1,1,-1,1,-1]], if and only if
      [p6, [1,0,0,1,0,-1],[1,0,-1,0,0,-1],[1,0,0,0,0,-1],
        [1,0,-1,1,-1,-1],[1,0,0,1,-1,-1],[1,0,-1,1,1,-1],
        [1,0,0,1,1,-1],[1,0,-1,0,1,-1],[1,0,0,0,1,-1],
        [1,0,-1,-1,1,-1],[1,0,0,-1,1,-1],[1,0,-1,1,0,-1]]
```

```
where
(#1) p62:=-9*c*a^3-180*d*c*a+192*d*b^2+Q1*x+Q2*x^2,
(#2) p6:=x^6+a*x^3+b*x^2+c*x+d,
(#3) p63:=-3*a*x^3-4*b*x^2-5*c*x-6*d,
 and
Q1:=160*c*b^2-18*b*a^3-150*a*c^2-144*a*d*b,
Q2:=-27*a^4+108*d*a^2-240*a*b*c+128*b^3,
```

The discriminant sequence of p_6 is $D := [1, 0, -a^2, D_4, D_5, D_6]$, where

$$D_4 = -27a^4 + 108da^2 - 240abc + 128b^3$$

$$D_5 = 81a^5c - 27a^4b^2 - 1134dca^3 + 648a^2db^2 + 1620c^2a^2b$$

$$-1344acb^3 + 3240acd^2 + 256b^5 + 1728d^2b^2 - 5400bdc^2 + 1875c^4$$

$$D_6 = 108c^3a^5 + 729d^2a^6 - 8748d^3a^4 + 34992a^2d^4 - 46656d^5 - 486ca^5db$$

$$+21384ca^3d^2b - 9720c^2b^2da^2 - 77760d^3cab - 22500dc^4b + 43200c^2b^2d^2$$

$$+6912ab^4cd + 3125c^6 - 27b^2a^4c^2 + 108b^3a^4d - 8640b^3a^2d^2 - 1350dc^3a^3$$

$$+2250c^4ba^2 - 13824d^3b^3 + 27000d^2ac^3 - 1600ab^3c^3 + 256b^5c^2 - 1024b^6d.$$

From the CRC of p_6, we can obtain the conditions on a, b, c, d such that $(\forall x)[p_6 > 0]$. $(\forall x)[p_6 > 0] \Leftrightarrow$ case (11) or case (12) holds. We can write the sign conditions directly. No mapping of conditions is necessary.

Case (11) holds $\Leftrightarrow$ the sign list of p_6 be $[1, 0, 0, 1, 0, 0]$ or $[1, 0, -1, 1, 0, 0] \Leftrightarrow [-a^2 = 0 \wedge D_4 > 0 \wedge D_5 = 0 \wedge D_6 = 0] \vee [-a^2 < 0 \wedge D_4 > 0 \wedge D_5 = 0 \wedge D_6 = 0] \Leftrightarrow D_4 > 0 \wedge D_5 = 0 \wedge D_6 = 0$.

Case (12) holds $\Leftrightarrow$ the sign list of p_6 be one of the following 12 lists:
$[1, 0, 0, 1, 0, -1] \Leftrightarrow [-a^2 = 0 \wedge D_4 > 0 \wedge D_5 = 0 \wedge D_6 < 0]$
$[1, 0, -1, 0, 0, -1] \Leftrightarrow [-a^2 < 0 \wedge D_4 = 0 \wedge D_5 = 0 \wedge D_6 < 0]$
$[1, 0, 0, 0, 0, -1] \Leftrightarrow [-a^2 = 0 \wedge D_4 = 0 \wedge D_5 = 0 \wedge D_6 < 0]$
$[1, 0, -1, 1, -1, -1] \Leftrightarrow [-a^2 < 0 \wedge D_4 > 0 \wedge D_5 < 0 \wedge D_6 < 0]$

$[1, 0, 0, 1, -1, -1] \Leftrightarrow [-a^2 = 0 \wedge D_4 > 0 \wedge D_5 < 0 \wedge D_6 < 0]$
$[1, 0, -1, 1, 1, -1] \Leftrightarrow [-a^2 < 0 \wedge D_4 > 0 \wedge D_5 > 0 \wedge D_6 < 0]$
$[1, 0, 0, 1, 1, -1] \Leftrightarrow [-a^2 = 0 \wedge D_4 > 0 \wedge D_5 > 0 \wedge D_6 < 0]$
$[1, 0, -1, 0, 1, -1] \Leftrightarrow [-a^2 < 0 \wedge D_4 = 0 \wedge D_5 > 0 \wedge D_6 < 0]$
$[1, 0, 0, 0, 1, -1] \Leftrightarrow [-a^2 = 0 \wedge D_4 = 0 \wedge D_5 > 0 \wedge D_6 < 0]$
$[1, 0, -1, -1, 1, -1] \Leftrightarrow [-a^2 < 0 \wedge D_4 < 0 \wedge D_5 > 0 \wedge D_6 < 0]$
$[1, 0, 0, -1, 1, -1] \Leftrightarrow [-a^2 = 0 \wedge D_4 < 0 \wedge D_5 > 0 \wedge D_6 < 0]$
$[1, 0, -1, 1, 0, -1] \Leftrightarrow [-a^2 < 0 \wedge D_4 > 0 \wedge D_5 = 0 \wedge D_6 < 0]$

Simplifying by hand or by QEPCAD (Brown, 2004), we conclude that case (12) holds $\Leftrightarrow D_6 < 0 \wedge [D_4 > 0 \vee D_5 > 0 \vee [D_4 = 0 \wedge D_5 = 0]]$.

Finally, by combining the conditions for case (11) and case (12), we obtain the desired result: $(\forall x)[p_6 > 0] \Leftrightarrow [D_4 > 0 \wedge D_5 = 0 \wedge D_6 = 0] \vee [D_4 = 0 \wedge D_5 = 0 \wedge D_6 < 0] \vee [D_4 > 0 \wedge D_6 < 0] \vee [D_5 > 0 \wedge D_6 < 0]$.

Example 29. Find the conditions on a, b, c, d such that $(\forall x)[x^8 + ax^3 + bx^2 + cx + d > 0]$.

For this problem, it is not necessary to generate the whole CRC. Only those RCs with no real roots are needed. The solution is
$(\forall x)[x^8 + ax^3 + bx^2 + cx + d > 0] \Leftrightarrow [D_6 < 0 \wedge D_7 = 0 \wedge D_8 = 0] \vee [D_6 < 0 \wedge D_8 > 0] \vee [D_7 < 0 \wedge D_8 > 0] \vee [D_6 = 0 \wedge D_7 = 0 \wedge D_8 > 0], where$

$$D_6 = 9375a^6 + 112000a^3cd - 172800a^2b^2d - 176400a^2bc^2 + 241920ab^3c - 62208b^5,$$

$$D_7 = -9375a^7c + 3125a^6b^2 - 332500a^4c^2d - 152000a^4bd^2$$
$$+416500a^3bc^3 + 744000a^3b^2cd - 409600a^2d^4 - 718200a^2b^3c^2$$
$$-216000a^2b^4d + 2580480abcd^3 - 1756160ac^3d^2 + 334368ab^5c$$
$$+2304960bc^4d - 46656b^7 - 2709504b^2c^2d^2 - 470596c^6 + 442368b^3d^3$$

$$D_8 = -823543c^8 + 42147840abc^3d^3 + 16777216d^7 + 3931200a^2b^4c^2d$$
$$-41287680ab^2cd^4 + 7529536bc^6d + 56250a^7bcd - 1960000a^3b^2c^3d$$
$$-8524800a^3b^3cd^2 + 186624b^8d + 381024ab^5c^3 - 46656b^7c^2 + 3125a^6b^2c^2$$
$$-2880000a^5cd^3 - 20070400a^2c^2d^4 + 19660800a^2bd^5 - 84375a^8d^2$$
$$+21676032b^3c^2d^3 - 22127616b^2c^4d^2 - 1617408ab^6cd + 600250a^3bc^5$$
$$-926100a^2b^3c^4 + 1907712a^2b^5d^2 + 4224000a^4b^2d^3 - 428750a^4c^4d$$
$$-12500a^6b^3d - 8605184ac^5d^2 - 3538944d^4b^4 - 12500a^7c^3 + 5992000a^4bc^2d^2.$$

Example 30. Find the conditions on a, b, c such that $(\forall x)[x^{10} + ax^2 + bx + c > 0]$.

A solution to this problem was given in Liang & Jeffrey (2006). A new solution is obtained by using the new algorithm.
$(\forall x)[x^{10} + ax^2 + bx + c > 0] \Leftrightarrow [a > 0 \wedge D_9 = 0 \wedge D_{10} = 0] \vee [a > 0 \wedge D_{10} < 0] \vee [D_9 > 0 \wedge D_{10} < 0] \vee [a = 0 \wedge D_9 = 0 \wedge D_{10} < 0]$, where D_9 and D_{10} were given in Liang & Jeffrey (2006).

Please notice the difference between the two solutions. The solution given here has been refined.

7. Conclusion

In this paper, we have proposed an improved algorithm for the automatic computation of the complete root classification of a real parametric polynomial, and a new test for non-realizable conditions. However, some issues deserve further consideration. For example, the output conditions are basically equalities and inequalities in terms of the parametric coefficients. A further step would be to determine what are the possible values of the parametric coefficients such that the conditions described are satisfied. This is essentially the problem of solving semi-algebraic systems, a problem well-known to be difficult. This problem may be addressed using interval analysis (Colagrossi & Miola, 1983) or method based on Gröbner basis (Rouillier, 2005). We will leave these issues in further work.

Acknowledgements

We are indebted to Professor Hoon Hong for his helpful discussion at MACIS 2006, Beijing. The discussion initiated this paper. We would like to thank the anonymous referees for valuable suggestions which help us greatly improve the first version of the paper.

References

Abdeljaoued, J., Diaz-Toca, G.M., Gonzalez-Vega, L., 2004. Minor of Bezout matrices, Subresultants and parameterization of the degree of the polynomial greatest common divisor. International Journal of Computer Mathematics 81, 1223-1238.

Arnon, D.S., 1988. Geometric Reasoning with Logic and Algebra. Artificial Intelligence 37, 37-60.

Basu, S., Pollack, R., Roy, M.-F., 2003. Algorithms in Real Algebraic Geometry. Springer-Verlag.

Brown, C.W., 2004. QEPCAD B: a program for computing with semi-algebraic sets using CADs. ACM SIGSAM Bulletin 37, 97-108.

Colagrossi, A., Miola, A.M., 1983. Computing real zeros of polynomials with parametric coefficients. ACM SIGSAM Bulletin 17, 12-15.

Collins, G.E., 1967. Subresultants and reduced polynomial remainder sequences. Journal of Association for Computing Machinery 14, 128-142.

Davenport, J.H., Heintz, J., 1988. Real quantifier elimination is doubly exponential. Journal of Symbolic Computation 5, 29-36.

Gonzalez-Vega, L., 1998. A combinatorial algorithm solving some quantifier elimination problems. Quantifier Elimination and Cylindrical Algebraic Decomposition (B.F. Caviness and J.R. Johnson, eds.), Springer-Verlag, 365-375.

Gonzalez-Vega, L., Lombardi, H., Recio, T., Roy, M.-F., 1998. Sturm-Habicht sequences, determinants and real roots of univariate polynomials. Quantifier Elimination and Cylindrical Algebraic Decomposition (B.F. Caviness and J.R. Johnson, eds.), Springer-Verlag, 300-316.

Habicht, V.W., 1948. Eine Verallgemeinerung des Sturmschen Wurzelzahlverfahrens. Comm. Math. Helvetici 21, 99-116.

Heintz, J., Roy, M.-F., Solerno, P., 1993. On the theoretical and practical complexity of the existential theory of the reals. The Computer Journal 36(5), 427-431.

Hong, H., 1993. Quantifier elimination for formulas constrained by quadratic equations. ISSAC'93 Proceedings, ACM Press, 264-274.

Jeffrey, D.J., Corless, R.M., 2006. Linear Algebra in Maple. In *CRC Handbook of Linear Algebra*, Editor L. Hogben. Chapter 72.

Liang, S., Jeffrey, D.J., 2006. An algorithm for computing the complete root classification of a parametric polynomial. Lecture Notes in Computer Science 4120, 116-130.

Liang, S., Zhang, J., 1999. A complete discrimination system for polynomials with complex coefficients and its automatic generation. Science in China (Series E) 42, 113-128.

Lickteig, T., Roy, M.F., 2001. Sylvester-Habicht sequences and fast Cauchy index computation. Journal of Symbolic Computation 31, 315-341.

Lombardi, H., Roy, M.F., Safely el Din, M., 2000. New structure theorem for subresultants. Journal of Symbolic Computation 29, 663-689.

Rouillier, F., 2005. On solving parametric systems. In Workshop on *Challenges in Linear and Polynomial Algebra in Symbolic Computation Software*. Banff International Research Center.

Sylvester, J.J., 1853. On a theory of syzygetic relations of two rational integral functions, comprising an application to the theory of Sturm's function and that of the greatest algebraical common measure. Trans. Roy. Soc. London 143, 407-548.

Yang, L., 1999. Recent advances on determining the number of real roots of parametric polynomials. Journal of Symbolic Computation 28, 225-242.

Yang, L., Hou, X., Zeng, Z., 1996. Complete discrimination system for polynomials. Science in China (Series E) 39, 628-646.