A NEW SUBRESULTANT PRS METHOD

Alkiviadis G. Akritas
University of Kansas
Department of Computer Science
Lawrence, Kansas, 66045

dedicated to the memory of my father

Abstract

A new subresultant polynomial-remainder-sequence (prs) method is presented where the polynomial remainders are obtained with matrix triangularization (whereas with the existing subresultant prs method they are obtained with a sequence of polynomial divisions). This method is based on our generalization [1] of a theorem by Van Vleck (1899) and uniformly treats both complete and incomplete prs's, making use of Bareiss's (1968) integer-preserving transformation algorithm for Gaussian elimination. Moreover, for the polynomials of the prs's, this method provides the smallest coefficients that can be expected without coefficient gcd computations (in certain cases the coefficients are smaller than those obtained with the existing prs method) and it clearly demonstrates the divisibility properties.

Introduction

In this note we restrict our discussion to univariate polynomials with integer coefficients and to computations in $Z[x]$, a unique factorization domain. Given the polynomial $p(x) = c_nx^n + c_{n-1}x^{n-1} + \cdots + c_0$, its degree is denoted by $\deg(p(x))$ and c_n, its leading coefficient, by $lc(p(x))$; moreover, $p(x)$ is called *primitive* if its coefficients are relatively prime.

Consider now $p_1(x)$ and $p_2(x)$, two primitive, nonzero polynomials in $Z[x]$, $\deg(p_1(x)) = n$ and $\deg(p_2(x)) = m$, $n \geq m$. Clearly, the polynomial division (with remainder) algorithm, call it PDF, that works over a field, cannot be used in $Z[x]$ since it requires exact divisibility by $lc(p_2(x))$. So we use *pseudo-division*, which always yields a pseudo-quotient and pseudo-remainder; in this process we have to premultiply $p_1(x)$ by $lc(p_2(x))^{n-m+1}$ and then apply algorithm PDF. Therefore we have:

$$lc(p_2(x))^{n-m+1}p_1(x) = q(x)p_2(x) + p_3(x),$$
$$\deg(p_3(x)) < \deg(p_2(x)). \qquad (1)$$

Applying the same process to $p_2(x)$ and $p_3(x)$, and then to $p_3(x)$ and $p_4(x)$, etc. (Euclid's algorithm), we obtain a *polynomial remainder sequence* (prs)

$$p_1(x), p_2(x), p_3(x), \ldots, p_h(x), p_{h+1}(x) = 0,$$

where $p_h(x) \neq 0$ is a greatest common divisor (gcd) of $p_1(x)$ and $p_2(x)$. If $n_i = \deg(p_i(x))$ and we have $n_i - n_{i+1} = 1$, for all i, the prs is called *complete*, otherwise, it is called *incomplete*. The problem with the above approach is that the coefficients of the polynomials in the prs grow exponentially and hence slow down the computations. We wish to control this coefficient growth. We observe that equation (1) can also be written more generally as

$$lc(p_{i+1}(x))^{n_i-n_{i+1}+1}p_i(x) = q_i(x)p_{i+1}(x) + \beta_ip_{i+2}(x),$$
$$\deg(p_{i+2}(x)) < \deg(p_{i+1}(x)), \qquad (2)$$

$i = 1, 2, \ldots, h-1$. That is, if a method for choosing β_i is given, the above equation provides an algorithm for constructing a prs. The obvious choice $\beta_i = 1$, for all i, is called the *Euclidean prs*; it was described above and leads to exponential growth of coefficients. Choosing β_i to be the greatest common divisor of the coefficients of $p_{i+2}(x)$ results in the *primitive prs*, and it is the best that can be done to control the coefficient growth. (Notice that here we are dividing $p_{i+2}(x)$ by the greatest common divisor of its coefficients before we use it again.) However, computing the greatest common divisor of the coefficients for each member of the prs (after the first two, of course) is an expensive operation and should be avoided.

So far, in order both to control the coefficient growth and to avoid the coefficient gcd computations, the Sylvester-Habicht subresultant prs method has been used (resultants are briefly defined below). According to this method, in case of a complete prs we choose

$$\beta_1 = 1 \text{ and } \beta_i = lc(p_i(x))^2, \quad i = 2, 3, \ldots, h-1, \qquad (3)$$

(Sylvester) [2], [11], whereas, in case of an incomplete prs we set

$$\beta_1 = (-1)^{n_1-n_2+1} \text{and } \beta_i = (-1)^{n_i-n_{i+1}+1}lc(p_i(x))H_i^{n_i - n_{i+1}},$$
$$i = 2, 3, \ldots, h-1, \qquad (4)$$

(Habicht) [8], where

$$H_2 = lc(p_2(x))^{n_1-n_2} \text{ and } H_i = lc(p_i(x))^{n_{i-1}-n_i} H_{i-1}^{1-(n_{i-1}-n_i)},$$
$$i = 3, 4, \ldots, h-1;$$

the latter, (4), is also known as the *subresultant* prs method. See also [4], [5], [6], [7], ([9] p. 410) where credit for the discovery of the Sylvester-Habicht subresultant prs method is given to the wrong person. (In general, (3) can be also expressed as

$$\beta_1 = 1 \text{ and } \beta_i = lc(p_i(x))^{n_i - n_{i+1}+1}, \quad i = 2, 3, \ldots, h-1, \qquad (3')$$

which is also known as the *reduced* prs method.)

Observe that in both cases above we perform polynomial pseudo-divisions. Moreover, in both cases we divide $p_{i+2}(x)$ by the corresponding β_i before we use it again. The proofs that the β_i's shown in (3) and (4) exactly divide $p_{i+2}(x)$ are rather complicated [6] and have up to now "hidden simple divisibility properties" [10]. Moreover, we cannot determine apriori whether a sequence is complete or incomplete.

In what follows we present a new subresultant prs method which uniformly treats both complete and incomplete prs's and provides the smallest coefficients in absolute value that can be expected without coefficient greatest common divisor computations; moreover, this method does not explicitly perform polynomial divisions and clearly demonstrates the existing divisibility properties. We also present a theorem which is a generalization of a theorem by Van Vleck. (We have failed to detect prior use of Van Vleck's theorem in the literature.)

The new method

Consider in $Z[x]$ the two (primitive) polynomials $p_1(x) = c_nx^n + c_{n-1}x^{n-1} + \cdots + c_0$ and $p_2(x) = d_mx^m + d_{m-1}x^{m-1} + \cdots + d_0$, $c_n \neq 0$, $d_m \neq 0$, $n \geq m$.

Step 1: Form the matrix corresponding to the resultant, $res(p_1(x),p_2(x))$, of the two polynomials $p_1(x)$ and $p_2(x)$ shown below:

$$R(p_1(x),p_2(x)) = \begin{bmatrix} c_n & c_{n-1} & \ldots c_0 & 0 & 0 \ldots & 0 \\ d_n & d_{n-1} & \ldots d_0 & 0 & 0 \ldots & 0 \\ 0 & c_n & \ldots\ldots c_0 & 0 & \ldots & 0 \\ 0 & d_n & \ldots\ldots d_0 & 0 & \ldots & 0 \\ & & \ldots\ldots & & & \\ 0 & 0 & \ldots c_n & c_{n-1} & \ldots\ldots & c_0 \\ 0 & 0 & \ldots d_n & d_{n-1} & \ldots\ldots & d_0 \end{bmatrix} \qquad (5)$$

Step 2: Using Bareiss's algorithm (described below) transform the matrix $R(p_1(x),p_2(x))$ (5) into its upper triangular form $T(R)$; then the coefficients of all the members of the polynomial remainder sequence of $p_1(x)$ and $p_2(x)$ are obtained from the rows of $T(R)$ according to our main theorem, which is presented below.

Theorem [1]: Let $p_1(x)$ and $p_2(x)$ be two polynomials, of degrees n and m, respectively, $n \geq m$. Transform the matrix $R(p_1(x),p_2(x))$, (5), of $p_1(x)$ and $p_2(x)$ into its upper triangular form $T(R)$; let n_i be the degree of the polynomial corresponding to the i-th row of $T(R)$, $i = 1, 2, ..., 2n$, and let $p_k(x)$, $k \geq 2$, be the k-th member of the (complete *or* incomplete) polynomial remainder sequence of $p_1(x)$ and $p_2(x)$. Then, if $p_k(x)$ is in row i of $T(R)$, the coefficients of $p_{k+1}(x)$ (within sign) are obtained from row i+j of $T(R)$, where j is the smallest integer such that $n_{i+j} < n_i$. (If $n = m$ associate both $p_1(x)$ and $p_2(x)$ with the first row of $T(R)$.)

Proof. Similar to the proof of Van Vleck's theorem [12].//

As a special case of the above theorem we obtain Van Vleck's theorem for complete prs's which states that the coefficients of the (i+1)-th member of the polynomial remainder sequence of $p_1(x)$ and $p_2(x)$ are obtained from the even rows 2i, $i = 1, 2, ..., h$, of the upper triangular form. Van Vleck demonstrated this theorem with an example [12]. However, the resultant is transformed into its upper triangular form by performing elementary row operations and removing at each step the greatest common divisor of the coefficients, a computation which we want to avoid.

On the other hand, in our method, we transform (5) into its upper triangular form using Bareiss's integer-preserving transformation algorithm [3]. That is: let $r_{00}^{(-1)} = 1$, and $r_{ij}^{(0)} = r_{ij}$, $i,j = 1, ..., n$; then for $k < i,j, \leq n$,

$$r_{ij}^{(k)} := (1 / r_{k-1,k-1}^{(k-2)}) \cdot \begin{vmatrix} r_{kk}^{(k-1)} & r_{kj}^{(k-1)} \\ r_{ik}^{(k-1)} & r_{ij}^{(k-1)} \end{vmatrix} \tag{6}$$

Of particular importance in Bareiss's algorithm is the fact that the determinant of order 2 is divided *exactly* by $r_{k-1,k-1}^{(k-2)}$ (the proof is very short and clear and is described in Bareiss's paper) and that the resulting coefficients are the smallest that can be expected without coefficient gcd computations and without introducing rationals. Notice how all the complicated expressions for β_i in the reduced and subresultant prs algorithms are mapped to the simple factor $r_{k-1,k-1}^{(k-2)}$ of this method.

Empirical results and conclusions

Below we present two examples to demonstrate our new method.

Example. If we consider the polynomials $p_1(x) = x^5 + 5x^4 + 10x^3 + 5x^2 + 5x + 2$ and $p_2(x) = x^4 + 4x^3 + 6x^2 + 2x + 1$ the upper triangular form of the matrix corresponding to $res(p_1(x),p_2(x))$ is

$$\left[\begin{array}{cccccccccc} 1 & 5 & 10 & 5 & 5 & 2 & 0 & 0 & 0 & 0 \\ 0 & 1 & 4 & 6 & 2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 4 & 3 & 4 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & -2 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 9 & -6 & -3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 58 & 50 & 18 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -266 & -112 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -756 & -532 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -980 \end{array}\right] \begin{array}{c} \\ \\ \\ * \\ \\ \\ \\ \\ \\ \end{array}.$$

The *-ed row indicates that a pivot took place. Therefore, the members of the prs generated by $p_1(x)$ and $p_2(x)$ are $3x^2 - 2x - 1$, $-266x - 112$ and -980.

As an example where the coefficients of the members of the prs, obtained with the new method, are smaller than those obtained with the Sylvester-Habicht method consider the polynomials $p_1(x) = x^3 - 7x + 7$ and $p_2(x) = 3x^2 - 7$. The Sylvester-Habicht prs method generates the polynomial sequence $p_3(x) = -42x + 63$, $p_4(x) = 49$, whereas our new method generates $p_3(x) = -14x + 21$, $p_4(x) = 49$. This occurs because the quotient on pseudo-dividing $p_1(x)$ by $p_2(x)$ is 3x with *no constant term* involved; this means that we did not have to multiply $p_1(x)$ by 3^2 but only by 3. (However, there is no way for knowing this before the actual division.) This explains why the coefficients obtained by our method for $p_3(x)$ are smaller by a factor of 3 than those obtained by the Sylvester-Habicht method.

The computing time of our method is $O(n^5L^2(|p(x)|_\infty))$, where $|p(x)|_\infty = \max(|p_1(x)|_\infty, |p_2(x)|_\infty)$, the maximum coefficient in absolute value of both $p_1(x)$ and $p_2(x)$, and $L(|p(x)|_\infty)$ is the number of bits used in the representation of this maximum coefficient; this is also the computing time of the Sylvester-Habicht prs method.

References

[1] Akritas, A.G.: A new method for computing polynomial greatest common divisors. TR-86-9, University of Kansas, Department of Computer Science, Lawrence, Ks. 66045.

[2] Akritas, A.G.: A simple proof of the validity of the reduced prs algorithm. Computing, Vol. 38, pp. 369- 372, 1987.

[3] Bareiss, E.H.: Sylvester's identity and multistep integer-preserving Gaussian elimination. Mathematic of Computation, Vol. 22, pp. 565-578, 1968.

[4] Brown, W.S.: On Euclid's algorithm and the computation of polynomial greatest common divisors. JACM, Vol. 18, pp. 476-504, 1971.

[5] Brown, W.S.: The subresultant prs algorithm. ACM Transactions On Mathematical Software, Vol. 4, pp. 237-249, 1978.

[6] Collins, G.E.: Subresultants and reduced polynomial remainder sequences. JACM, Vol.14, pp. 128-142, 1967.

[7] Fryer, W.D.: Applications of Routh's algorithm to network theory problems. IEEE Transactions on Circuit Theo T-6, pp. 144-149, 1959.

[8] Habicht, W.: Eine Verallgemeinung des Sturmschen Wurzelzaehlverfahrens. Commentarii Mathematici Helvetici, Vol. 21, pp. 99-116, 1948.

[9] Knuth, D.E.: The art of computer programming. Vol. 2/ Seminumerical algorithms. Second Edition. Addison-Wesley, Reading, Massachusetts, 1981.

[10] Loos, R.: Generalized polynomial remainder sequences. In: Computer Algebra Symbolic and Algebraic Computations. Ed. by B. Buchberger, G.E. Collins and R. Loos, Springer Verlag, Wien, New York, 1982, Computing Supplement 4, pp. 115-137.

[11] Sylvester, J.J.: On a theory of the syzygetic relations of two rational integral functions, comprising an application to the theory of Sturm's functions, and that of the greatest algebraical common measure. Philoshophical Transactions, Vol. 143, pp. 407-548, 1853.

[12] Van Vleck, E. B.: On the determination of a series of Sturm's functions by the calculation of a single determinant. Annals of Mathematics, Second Series, Vol. 1, pp. 1-13, 1899-1900.