A Subresultant Theory for Ore Polynomials with Applications

Ziming Li*

GMD-SCAI
D-53754 Sankt Augustin, Germany
ziming.li@gmd.de
http://www.gmd.de/SCAI/people

Abstract

The subresultant theory for univariate commutative polynomials is generalized to Ore polynomials. The generalization includes: the subresultant theorem, gap structure, and subresultant algorithm. Using this generalization, we define Sylvester's resultant of two Ore polynomials, derive the respective determinantal formulas for the greatest common right divisor and least common left multiple of two Ore polynomials, and present a fraction-free version of the non-commutative extended Euclidean algorithm.

1 Introduction

Greatest common right divisors (abbreviated as: gcrd) and least common left multiples (abbreviated as: lclm) are basic objects in the theory and computation of Ore polynomials [11, 2, 3]. For example, the gcrd of linear ordinary differential (shift) operators represents the intersection of their solution spaces, and the lclm of these operators represents the sum of their solution spaces. The non-commutative Euclidean and extended Euclidean algorithms are used to compute gcrd's and lclm's, respectively. Naive applications of these two algorithms lead to inefficient implementations because of the coefficient growth of intermediate polynomials, as seen in the commutative case. Motivated by the improvements made on the commutative Euclidean algorithm, we generalize the subresultant theory for univariate commutative polynomials to univariate Ore polynomials. This generalization provides a way to control the coefficient growth in the non-commutative Euclidean and extended Euclidean algorithms.

The commutative subresultant theory has undergone a quite intensive study since the work of Collins, Brown and Traub [6, 4, 9, 10]. Chardin presented a subresultant theory for linear ordinary differential operators [5]. In this paper we present a subresultant theory for Ore polynomials over a commutative domain, which includes not only linear ordinary differential operators but also linear shift operators, q-difference operators, etc. The subresultant theory can be further extended to linear inhomogeneous differential and difference equations [7], but we will not present this extension because most of the applications of subresultants are found in an Ore polynomial ring.

To extend the commutative subresultant theory, we will overcome two difficulties. First, we need to find new techniques to prove essentially the same statements as those in the commutative case without assuming the commutativity of multiplication. This problem is solved by Lemma 3.1. Second, we need to simplify σ-factorial expressions in order to remove extraneous factors. We also remark that it looks rather complicated to adapt the approach in [9, 10] to extend the commutative subresultant theory, because it is not trivial to build up a generic coefficient domain for Ore polynomials.

Applications of the subresultant theory include: computing gcrd's [8]; extending Sylvester's resultant to Ore polynomials (Definition 6.1), expressing the gcrd and lclm of two Ore polynomials by determinants (Proposition 6.1), and computing lclm's (Proposition 6.2). The correspondence between subresultants and intermediate polynomials occurring in the Euclidean algorithm is also useful to estimate coefficient and degree bounds, and to analyze complexities.

The organization of this paper is as follows. Section 2 introduces Ore polynomial rings, specifies the notation that will be used later, and defines subresultants. Section 3 presents the row-reduction formula for subresultants, which makes it possible to extend the techniques for establishing commutative subresultant theory in [6, 4]. The subresultant theorem for Ore polynomials is proved in Section 4. The subresultant algorithm is described in Section 5. Applications are presented in Section 6.

2 Preliminaries

Let $\mathcal{R}$ be a commutative domain and σ an injective endomorphism from $\mathcal{R}$ to itself, which is called a *conjugate operator.* An endomorphism δ of the additive group $(\mathcal{R}, +, 0)$ is called a *pseudo-derivation* with respect to σ if

$$\delta(ab) = \sigma(a)\delta(b) + \delta(a)b, \quad \text{for all } a, b \in \mathcal{R}.$$

The (non-commutative) multiplication in $\mathcal{R}[X]$ is defined by the commutation rule

$$Xa = \sigma(a)X + \delta(a), \quad \text{for all } a \in \mathcal{R}. \tag{1}$$

The triple $(\mathcal{R}[X], \sigma, \delta)$ is called an *Ore polynomial ring.* For $A, B \in \mathcal{R}[X]$, the product of A and B is denoted by AB and

*on leave from the Mathematics-Mechanization Research Center, Institute of Systems Sciences, Academia Sinica, Beijing.

ISSAC'98, Rostock, Germany. © 1998 ACM 1-58113-002-3/ 98/ 0008 $5.00

the degree of AB is equal to the sum of the degrees of A and B. The conjugate operator σ and pseudo-derivation δ can be uniquely extended to the quotient field of $\mathcal{R}$ by letting $\sigma(a/b) = \sigma(a)/\sigma(b)$ and $\delta(a/b) = (b\delta(a) - a\delta(b))/(\sigma(b)b)$, for $a, b \in \mathcal{R}$ with $b \neq 0$ (see [7, Proposition 2.2]).

Example 1 Denote the identity and null mappings on $\mathcal{R}$ by $\mathbf{1}$ and $\mathbf{0}$, respectively. The ring $(\mathcal{R}[X], \mathbf{1}, \mathbf{0})$ is the ring of usual commutative polynomials over $\mathcal{R}$. If D is a derivation operator on $\mathcal{R}$, then the ring $(\mathcal{R}[X], \mathbf{1}, D)$ is isomorphic to the ring of linear homogeneous differential polynomials in one differential indeterminate over $\mathcal{R}$. If E is an injective endomorphism of the domain $\mathcal{R}$, then the ring $(\mathcal{R}[X], E, \mathbf{0})$ is isomorphic to the ring of linear homogeneous difference polynomials in one difference indeterminate (with respect to E) over $\mathcal{R}$.

We denote $(\mathcal{R}[X], \sigma, \delta)$, $\sigma(r)$ and $\delta(r)$ by $\mathcal{R}[X]$, σr and δr, respectively. For $A \in \mathcal{R}[X]$, the leading coefficient of A is denoted by $\mathrm{lc}(A)$. An easy induction shows that

$$\mathrm{lc}(X^n A) = \sigma^n \mathrm{lc}(A), \quad \text{for all } n \in \mathbf{N}. \qquad (2)$$

Definition 2.1 Let $r \in \mathcal{R}$. The *nth σ-factorial of* r is the product

$$\prod_{i=0}^{n-1} \sigma^i r, \quad \text{for all } n \in \mathbf{Z}^+,$$

which is denoted by $r^{[n]}$. In addition, $r^{[0]}$ is set to be 1.

The next lemma holds because σ is a ring homomorphism.

Lemma 2.1 If $r, s \in \mathcal{R}$, and $m, n \in \mathbf{N}$, then

1. $(rs)^{[m]} = r^{[m]} s^{[m]}$,
2. $r^{[m+n]} = r^{[m]} (\sigma^m r)^{[n]}$,
3. $(r^{[m]})^{[n]} = (r^{[n]})^{[m]}$, (define $r^{[m][n]}$ as $(r^{[m]})^{[n]}$),
4. $r^{[m+1][n+1]} = r^{[m+n+1]} (\sigma r)^{[m][n]}$.

Definition 2.2 For $A, B \in \mathcal{R}[X]$ with $\deg A = m$ and $\deg B = n \geq 0$, a *pseudo-remainder* of A and B is either A, if $m < n$; or $C \in \mathcal{R}[X]$ such that $\deg C < \deg B$ and

$$\left(\prod_{i=0}^{m-n} \mathrm{lc}(X^i B) \right) A = QB + C, \quad Q \in \mathcal{R}[X].$$

We call Q the (left) *pseudo-quotient* of A and B.

The pseudo-remainder of A and B is unique and denoted by $\mathrm{prem}(A, B)$. By (2) we get the *pseudo-remainder formula*

$$\mathrm{lc}(B)^{[m-n+1]} A = QB + \mathrm{prem}(A, B). \qquad (3)$$

Let $A_1, A_2, \ldots, A_{k-1}, A_k$ be a sequence of non-zero elements of $\mathcal{R}[X]$ such that A_i is $\mathcal{R}$-linearly dependent on $\mathrm{prem}(A_{i-2}, A_{i-1})$, for $i = 3, \ldots\ k$, and $\mathrm{prem}(A_{k-1}, A_k) = 0$. Such a sequence is called a *polynomial remainder sequence* (abbreviated as: p.r.s.) of A_1 and A_2. We will show that any member of a p.r.s. of A_1 and A_2 is $\mathcal{R}$-linearly dependent on a subresultant of A_1 and A_2 (Corollary 5.3).

Now, we recall the definition of determinant polynomials [9, 10] and define subresultants.

Definition 2.3 Let M be an $r \times c$ matrix over $\mathcal{R}$. Assume that $r \leq c$. The determinant polynomial of M is

$$\mid M \mid = \sum_{i=0}^{c-r} \det(M_i) X^i,$$

where M_i is the $r \times r$ matrix whose first $(r-1)$ columns are the first $(r-1)$ columns of M and whose last column is the $(c-i)$th column of M, for $i = 0, \ldots, c-r$.

We will encounter determinants whose last columns contain polynomials in $\mathcal{R}[X] \setminus \mathcal{R}$. When expanding such a determinant, we place the product of entries in $\mathcal{R}$ on the left-hand side of the entry in $\mathcal{R}[X] \setminus \mathcal{R}$. With this convention we can express determinant polynomials by determinants. Let the matrix M in Definition 2.3 be

$$\begin{pmatrix} m_{11} & \cdots & m_{1,r-1} & m_{1r} & \cdots & m_{1c} \\ m_{21} & \cdots & m_{2,r-1} & m_{2r} & \cdots & m_{2c} \\ \cdot & \cdots & \cdot & \cdot & \cdots & \cdot \\ m_{r1} & \cdots & m_{r,r-1} & m_{rr} & \cdots & m_{rc} \end{pmatrix}$$

and, for $i = 1, \ldots, r$, let

$$H_i = m_{i1} X^{c-1} + \cdots + m_{ir} X^{c-r} + \cdots + m_{ic}.$$

Then

$$\mid M \mid = \det \begin{pmatrix} m_{11} & \cdots & m_{1,r-1} & H_1 \\ m_{21} & \cdots & m_{2,r-1} & H_2 \\ \cdot & \cdots & \cdot & \cdot \\ m_{r1} & \cdots & m_{r,r-1} & H_r \end{pmatrix}. \qquad (4)$$

Thus a determinant polynomial is alternative and multilinear for its rows.

Let $\mathcal{A} : A_1, A_2, \ldots, A_r$ be a sequence of polynomials in $\mathcal{R}[X]$ and let d be the maximum of the degrees of the A_i's. The matrix associated with $\mathcal{A}$, denoted by $\mathrm{mat}(\mathcal{A})$, is the $r \times (d+1)$ matrix whose entry in the ith row and jth column is the coefficient of X^{d+1-j} in A_i, where $1 \leq i \leq r$ and $1 \leq j \leq d+1$. If $r \leq d+1$, then the determinant polynomial of $\mathcal{A}$ is defined to be $|\mathrm{mat}(\mathcal{A})|$, which is denoted by $\mid A_1, A_2, \ldots, A_r \mid$ or $\mid \mathcal{A} \mid$.

Lemma 2.2 Let $r \in \mathcal{R}$ with $r \neq 0$, and $A \in \mathcal{R}[X]$. If

$$H_m = \mid \ldots, X^m(rA), X^{m-1}(rA), \ldots, X(rA), rA, \ldots \mid$$

is a determinant polynomial of a polynomial sequence in $\mathcal{R}[X]$, where $m \in \mathbf{N}$, then

$$H_m = r^{[m+1]} \mid \ldots, X^m A, X^{m-1} A, \ldots, XA, A, \ldots \mid .$$

Proof We proceed by induction on m. The assertion is trivial when $m = 0$. Assume that $m > 0$ and the lemma is true for $m - 1$. Thus,

$$H_m = r^{[m]} \mid \ldots, X^m(rA), X^{m-1}A, \ldots, XA, A, \ldots \mid . \qquad (5)$$

It follows from [8, Lemma 4.1] that $X^m(rA)$ in (5) can be replaced by $(\sigma^m r)(X^m A)$.

Definition 2.4 Let $A, B \in \mathcal{R}[X]$ with $\deg A = m$ and $\deg B = n$, where $m \geq n$. The nth subresultant of A and B is B. For $j = n-1, n-2, \ldots, 0$, the *jth subresultant* of A and B, $\mathrm{sres}_j(A, B)$, is

$$\mid \underbrace{X^{n-j-1}A, \ldots, XA,\ A,}_{n-j}\ \underbrace{X^{m-j-1}B, \ldots, XB,\ B}_{m-j} \mid,$$

The sequence $\mathcal{S}(A, B)$: $A, B, \mathrm{sres}_{n-1}(A, B), \ldots, \mathrm{sres}_0(A, B)$, is called the *subresultant sequence* of A and B.

Example 2 Let $A = a_2X^2 + a_1X + a_0$ and $B = b_2X^2 + b_1X + b_0$. Then

$$\mathrm{sres}_1(A,B) \;=\; \mid A, B \mid \;=\; \begin{vmatrix} a_2 & A \\ b_2 & B \end{vmatrix},$$

and

$$\begin{aligned} \mathrm{sres}_0(A,B) &= \mid XA, A, XB, B \mid \\ &= \begin{vmatrix} \sigma a_2 & \delta a_2 + \sigma a_1 & \delta a_1 + \sigma a_0 & XA \\ 0 & a_2 & a_1 & A \\ \sigma b_2 & \delta b_2 + \sigma b_1 & \delta b_1 + \sigma b_0 & XB \\ 0 & b_2 & b_1 & B \end{vmatrix}. \end{aligned}$$

Note that the jth subresultant has degree no greater than j, and that all the subresultants of A and B are contained in the left ideal generated by A and B because of (4). The next lemma links $\mathrm{sres}_{n-1}(A,B)$ and $\mathrm{prem}(A,B)$.

Lemma 2.3 If A and B are the same as in Definition 2.4, then $\mathrm{sres}_{n-1}(A,B) = (-1)^{m-n+1}\mathrm{prem}(A,B)$.

Proof The lemma is proved by the following calculation:

$$\begin{aligned} &\mathrm{lc}(B)^{[m-n+1]}\mathrm{sres}_{n-1}(A,B) \\ &= \mid \mathrm{lc}(B)^{[m-n+1]}A,\ X^{m-n}B, \ldots, B \mid \\ &= \mid \mathrm{prem}(A,B),\ X^{m-n}B, \ldots, B \mid \text{ (by (3))} \\ &= (-1)^{m-n+1} \mid X^{m-n}B, \ldots, B,\ \mathrm{prem}(A,B) \mid \\ &= (-1)^{m-n+1}\mathrm{lc}(B)^{[m-n+1]}\mathrm{prem}(A,B). \end{aligned}$$

3 Row-Reduction Formula for Subresultants

Some proofs in the commutative subresultant theory are tacitly based on the following property: if i is a positive integer, and A and B are two univariate *commutative* polynomials in the indeterminate X, then

$$X^i\mathrm{prem}(A,B) = \mathrm{prem}(X^iA, X^iB).$$

However, this equality no longer holds when A and B are Ore polynomials. To establish a subresultant theory for Ore polynomials, we replace this property by the property that *if A and B are in $\mathcal{R}[X]$, then the difference between $X^i\mathrm{prem}(A,B)$ and $\mathrm{prem}(X^iA, X^iB)$ is an $\mathcal{R}$-linear combination of $X^{i-1}A, \ldots, XA, A, X^{\deg A - \deg B + i}B, \ldots, XB, B$* (see [8, Lemma 4.2]). This replacement allows us to prove the row-reduction formula (6), by which the techniques for proving the commutative subresultant theorem is extended.

Lemma 3.1 Let $A, B \in \mathcal{R}[X]$ with respective degrees m and n, where $m \geq n \geq 0$. If there exist non-zero elements $q, r, s \in \mathcal{R}$ and $F, G \in \mathcal{R}[X]$ such that $qB = rF$ and $\mathrm{sres}_{n-1}(A,B) = sG$, then, for $0 \leq j \leq n$, we have

$$\begin{aligned} &q^{[m-j]}\mathrm{lc}(B)^{[m-n+1][n-j]}\mathrm{sres}_j(A,B) = \\ &\quad r^{[m-j]}s^{[n-j]} \mid X^{m-j-1}F, \ldots, F,\ X^{n-j-1}G, \ldots, G \mid. \end{aligned} \tag{6}$$

Proof Let $C = \mathrm{prem}(A,B)$, $C_k = \mathrm{prem}(X^kA, X^kB)$, for $k \in \mathbf{N}$, and $S_j = \mathrm{sres}_j(A,B)$. By Definition 2.4 we have

$$S_j = \mid \underbrace{X^{n-j-1}A, \ldots, XA,\ A,}_{n-j}\ \underbrace{X^{m-j-1}B, \ldots, XB,\ B}_{m-j} \mid.$$

The pseudo-remainder formula for $X^{n-j-1}A$ and $X^{n-j-1}B$ implies that the polynomial

$$\sigma^{n-j-1}(\mathrm{lc}(B))^{[m-n+1]}X^{n-j-1}A - C_{n-j-1}$$

is an $\mathcal{R}$-linear combination of the polynomials

$$X^{m-j-1}B, \ldots, X^{n-j}B, X^{n-j-1}B.$$

Thus, Lemma 4.2 in [8] implies that the polynomial

$$\sigma^{n-j-1}(\mathrm{lc}(B))^{[m-n+1]}X^{n-j-1}A - X^{n-j-1}C$$

is an $\mathcal{R}$-linear combination of the polynomials

$$X^{n-j-2}A, \ldots, XA,\ A, X^{m-j-1}B, \ldots, XB,\ B.$$

Hence, we have

$$\begin{aligned} &\sigma^{n-j-1}(\mathrm{lc}(B))^{[m-n+1]}S_j \\ &\quad = \mid X^{n-j-1}C, X^{n-j-2}A, \ldots,\ A,\ X^{m-j-1}B, \ldots,\ B \mid. \end{aligned} \tag{7}$$

The same reasoning allows us to replace X^iA by X^iC on the right-hand of (7), while at the same time multiplying by the power $\sigma^i(\mathrm{lc}(B))^{[m-n+1]}$ on the left-hand of (7), for $i = n-j-2,\ n-j-3,\ \ldots,\ 0$. We eventually arrive at

$$\begin{aligned} &\mathrm{lc}(B)^{[n-j][m-n+1]}S_j \\ &\quad = \mid X^{n-j-1}C, X^{n-j-2}C, \ldots,\ C,\ X^{m-j-1}B, \ldots,\ B \mid. \end{aligned}$$

Lemma 2.3 then asserts that

$$\begin{aligned} &\mathrm{lc}(B)^{[n-j][m-n+1]}S_j \\ &\quad = \mid X^{m-j-1}B, \ldots, B,\ X^{n-j-1}S_{n-1}, \ldots, S_{n-1} \mid. \end{aligned}$$

Thus, the lemma follows from Lemma 2.2.

4 Subresultant Theorem

Notation To avoid endlessly repeating the assumptions, in the rest of this article, we let A and B be in $\mathcal{R}[X]$ with respective degrees m and n, where $m \geq n \geq 0$. Let $S_n = B$ and $S_j = \mathrm{sres}_j(A,B)$, for $j = n-1,\ n-2,\ \ldots,\ 0$. The subresultant sequence $\mathcal{S}(A,B)$ consists of the polynomials $A, S_n, \cdots, S_1, S_0$.

Definition 4.1 The jth subresultant S_j is said to be *regular* if $\deg S_j = j$ and otherwise S_j is *defective.* In particular, the nth subresultant S_n is always regular.

First, we demonstrate the relation between the members of $\mathcal{S}(A,B)$ and subresultants of the two consecutive non-zero members of $\mathcal{S}(A,B)$ in the next lemma.

Lemma 4.1 Let $\alpha_i = \mathrm{lc}(S_i)$, for all i with $n \geq i \geq 0$, $\beta_n = \sigma\mathrm{lc}(S_n)^{[m-n]}$, and $\beta_i = \sigma\mathrm{lc}(S_i)$, for all i with $n-1 \geq i \geq 0$. If S_{j+1} is regular, for some j with $n-1 \geq j \geq 0$, then the following hold:

1. If $S_j = 0$, then $S_i = 0$, for $j-1 \geq i \geq 0$.
2. If $S_j \neq 0$ and $\deg S_j = r$, then
$$S_i = 0, \quad (j-1 \geq i \geq r+1), \tag{8}$$
$$\beta_{j+1}^{[j-r]}S_r = \beta_j^{[j-r]}S_j, \tag{9}$$
and
$$\alpha_{j+1}^{[r-i]}\beta_{j+1}^{[j-i]}S_i = \mathrm{sres}_i(S_{j+1}, S_j), \quad (r-1 \geq i \geq 0). \tag{10}$$

Proof The proof will be done by induction on the sequence of the regular subresultants in $\mathcal{S}(A,B)$. As S_n is the first regular subresultant in $\mathcal{S}(A,B)$, we start with the case in which $j = n-1$. Let i be an integer such that $n-2 \geq i \geq 0$. By the definition of subresultants, we have

$$S_i = \mid X^{n-1-i}A, \ldots, A,\ X^{m-1-i}S_n, \ldots, S_n \mid .$$

It then follows from the row-reduction formula (6) that

$$\alpha_n^{[m-n+1][n-i]} S_i = R_i, \tag{11}$$

where $R_i = \mid X^{m-1-i}S_n, \ldots, S_n, X^{n-1-i}S_{n-1}, \ldots, S_{n-1} \mid$.

If $S_{n-1} = 0$, then $R_i = 0$ $(n-2 \geq i \geq 0)$, so is S_i by (11).

Let $\deg S_{n-1} = r \geq 0$ and $\deg \left(X^{n-1-i}S_{n-1}\right) = d$.

If $n-2 \geq i \geq r+1$, then $R_i = 0$ since $\deg S_n > d+1$. Therefore, $S_i = 0$ by (11). If $i = r$, then $R_r = \alpha_n^{[m-r]}\beta_{n-1}^{[n-1-r]}S_{n-1}$ because $\deg S_n = d+1$. Hence, (11) can be rewritten as $\alpha_n^{[m-n+1][n-r]}S_r = \alpha_n^{[m-r]}\beta_{n-1}^{[n-1-r]}S_{n-1}$. Since $\alpha_n^{[m-n+1][n-r]} = \alpha_n^{[m-r]}\beta_n^{[n-1-r]}$ by Lemma 2.1, (9) holds for $j = n-1$. If $r-1 \geq i \geq 0$, then

$$R_i = (\sigma^{r-i}\alpha_n)^{[m-r]}\mathrm{sres}_i(S_n, S_{n-1}), \tag{12}$$

because $\deg S_n = d - (r-1-i)$. By (12) and (11), we get $\alpha_n^{[m-n+1][n-i]}S_i = (\sigma^{r-i}\alpha_n)^{[m-r]}\mathrm{sres}_i(S_n, S_{n-1})$. Since $\alpha_n^{[m-n+1][n-i]} = \alpha_n^{[r-i]}(\sigma^{r-i}\alpha_n)^{[m-r]}\beta_n^{[n-1-i]}$ by Lemma 2.1, (10) holds for $j = n-1$. The proof of the base case is done.

We assume that the lemma holds for the regular subresultant S_{j+1} and that $\deg S_j = r$. In other words, the first assertion of the lemma, (8), (9) and (10) hold. If $S_j = 0$ then there is no regular subresultant that follows S_j. So, there is nothing to prove. Suppose $S_j \neq 0$. Then the regular subresultant next to S_{j+1} must be S_r by our induction hypothesis. Let $\deg(S_{r-1}) = t$. We have to prove that if $S_{r-1} = 0$, $S_i = 0$ for $r-2 \geq i \geq 0$, and that if $S_{r-1} \neq 0$,

$$S_i = 0, \quad (r-2 \geq i \geq t+1), \tag{13}$$

$$\beta_r^{[r-1-t]}S_t = \beta_{r-1}^{[r-1-t]}S_{r-1}, \tag{14}$$

and

$$\alpha_r^{[t-i]}\beta_r^{[r-1-i]}S_i = \mathrm{sres}_i(S_r, S_{r-1}), \quad (t-1 \geq i \geq 0). \tag{15}$$

Before going to induction, we point out two important relations hiding in (9). Equating the leading coefficients of both sides of (9) yields

$$\beta_{j+1}^{[j-r]}\alpha_r = \beta_j^{[j-r]}\alpha_j. \tag{16}$$

Applying σ to both sides of this equality, we get

$$(\sigma\beta_{j+1})^{[j-r]}\beta_r = \beta_j^{[j-r+1]}. \tag{17}$$

Based on the induction hypothesis we claim that

$$\alpha_r^{[j-i+1]}\beta_r^{[r-1-i]}S_i = T_i, \quad (r-2 \geq i \geq 0)\ , \tag{18}$$

where

$$T_i = \mid X^{j-i}S_r, \ldots, S_r,\ X^{r-1-i}S_{r-1}, \ldots, S_{r-1} \mid . \tag{19}$$

Proof of the Claim. Observe that (9) and (10) (setting $i = r-1$) imply that

$$\beta_{j+1}^{[j-r]}S_r = \beta_j^{[j-r]}S_j \tag{20}$$

and

$$\alpha_{j+1}\beta_{j+1}^{[j-r+1]}S_{r-1} = \mathrm{sres}_{r-1}(S_{j+1}, S_j). \tag{21}$$

The induction hypothesis (10) asserts that, for $t \geq i \geq 0$,

$$\alpha_{j+1}^{[r-i]}\beta_{j+1}^{[j-i]}S_i = \mid X^{r-1-i}S_{j+1}, \ldots, S_{j+1},\ X^{j-i}S_j, \ldots S_j \mid .$$

It follows from the above equality, (20), (21), and Lemma 3.1 that $r_iS_i = T_i$, where

$$r_i = \frac{\left(\beta_j^{[j-r][j-i+1]}\alpha_j^{[j-r+2][r-i]}\right)\left(\alpha_{j+1}^{[r-i]}\beta_{j+1}^{[j-i]}\right)}{\beta_{j+1}^{[j-r][j-i+1]}\left(\alpha_{j+1}^{[r-i]}\beta_{j+1}^{[j-r+1][r-i]}\right)}.$$

Note that, in the above deduction, S_{j+1}, S_j, r, S_r and S_{r-1} play the roles of A, B, n, F, and G in the statement of Lemma 3.1, respectively. It remains to show that $r_i = \alpha_r^{[j-i+1]}\beta_r^{[r-1-i]}$. Canceling $\alpha_{j+1}^{[r-i]}$ yields

$$r_i = \left(\frac{\beta_j^{[j-r][j-i+1]}}{\beta_{j+1}^{[j-r][j-i+1]}}\right)\frac{\alpha_j^{[j-r+2][r-i]}\beta_{j+1}^{[j-i]}}{\beta_{j+1}^{[j-r+1][r-i]}}.$$

The above equality can be simplified by (16) to

$$r_i = \left(\frac{\alpha_r^{[j-i+1]}}{\alpha_j^{[j-i+1]}}\right)\frac{\alpha_j^{[j-r+2][r-i]}\beta_{j+1}^{[j-i]}}{\beta_{j+1}^{[j-r+1][r-i]}}. \tag{22}$$

Lemma 2.1 implies

$$\alpha_j^{[j-r+2][r-i]} = \alpha_j^{[j-i+1]}\beta_j^{[j-r+1][r-1-i]}$$

and

$$\beta_{j+1}^{[j-r+1][r-i]} = \beta_{j+1}^{[j-i]}(\sigma\beta_{j+1})^{[j-r][r-1-i]}.$$

So equation (22) can be further simplified to

$$r_i = \alpha_r^{[j-i+1]}\left(\frac{\beta_j^{[j-r+1][r-1-i]}}{(\sigma\beta_{j+1})^{[j-r][r-1-i]}}\right).$$

It then follows from (17) that $r_i = \alpha_r^{[j-i+1]}\beta_r^{[r-1-i]}$. The claim is proved.

If $S_{r-1} = 0$, then $T_i = 0$, $(r-2 \geq i \geq 0)$, so is S_i by (18).

Assume $t \geq 0$ and denote $\deg\left(X^{r-1-i}S_{r-1}\right)$ by d.

If $r-2 \geq i \geq t+1$, $T_i = 0$ since $\deg(S_r) > d+1$, so is S_i by (18). If $i = t$, then $T_i = \alpha_r^{[j-t+1]}\beta_{r-1}^{[r-t-1]}S_{r-1}$ because $\deg(S_r) = d+1$. Hence (18) implies $\alpha_r^{[j-t+1]}\beta_r^{[r-t-1]}S_t = \alpha_r^{[j-t+1]}\beta_{r-1}^{[r-t-1]}S_{r-1}$. Equation (14) follows. If $t-1 \geq i \geq 0$, then $T_i = (\sigma^{t-i}\alpha_r)^{[j-t+1]}\mathrm{sres}_i(S_r, S_{r-1})$ because $\deg S_r = d - (t-i-1)$. Hence (18) implies

$$\alpha_r^{[j-i+1]}\beta_r^{[r-1-i]}S_i = (\sigma^{t-i}\alpha_r)^{[j-t+1]}\mathrm{sres}_i(S_r, S_{r-1}). \tag{23}$$

Observe that Lemma 2.1 implies

$$\alpha_r^{[j-i+1]} = \alpha_r^{[(t-i)+(j-t+1)]} = \alpha_r^{[t-i]}(\sigma^{t-i}\alpha_r)^{[j-t+1]}.$$

Using this relation to remove the like σ-factorial expressions from both sides of (23), we find (15). The lemma is proved.

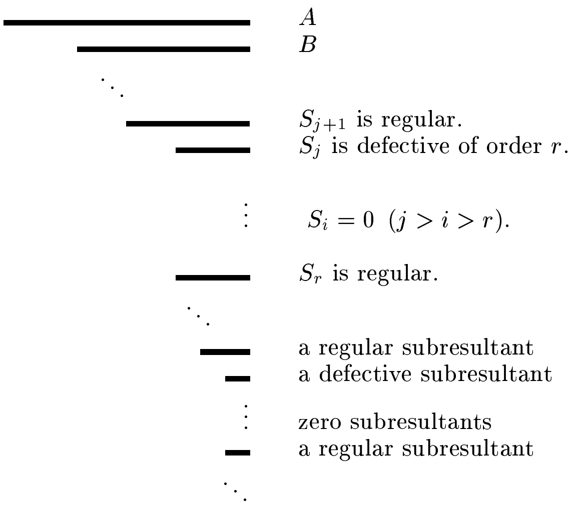

Figure 1: The gap structure of $\mathcal{S}(A, B)$

Theorem 4.2 (Subresultant Theorem) Let $\alpha_i = \mathrm{lc}(S_i)$, for $i = n, n-1, \ldots, 0$; $\beta_n = \sigma\mathrm{lc}(S_n)^{[m-n]}$, and $\beta_i = \sigma\mathrm{lc}(S_i)$, for $i = n-1, \ldots, 0$. If S_{j+1} is regular, for some j with $n-1 \geq j \geq 0$, and $\deg S_j = r$, then

1. If $S_j = 0$, then
$$S_i = 0, \quad (j-1 \geq i \geq 0). \tag{24}$$

2. If $S_j \neq 0$, then
$$S_i = 0, \quad (j-1 \geq i \geq r+1), \tag{25}$$
$$\beta_{j+1}^{[j-r]} S_r = \beta_j^{[j-r]} S_j, \tag{26}$$
and
$$\alpha_{j+1}\beta_{j+1}^{[j-r+1]} S_{r-1} = (-1)^{j-r}\mathrm{prem}(S_{j+1}, S_j). \tag{27}$$

Proof Equations (24), (25), and (26) were proved in Lemma 4.1. If $i = r-1$, (10) in Lemma 4.1 becomes
$$\alpha_{j+1}(\beta_{j+1})^{[j-r+1]} S_{r-1} = \mathrm{sres}_{r-1}(S_{j+1}, S_j)$$
Equation (27) then follows from Lemma 2.3.

A "formula-free" version of Theorem 4.2 reads:

Corollary 4.3 Let $n-1 \geq j \geq 0$. If S_{j+1} is regular and $\deg S_j = r$, then, for all i with $j-1 \geq i \geq r+1$, the subresultant S_i is zero. If, moreover, $S_j \neq 0$, then S_r is regular, S_j and S_r are $\mathcal{R}$-linearly dependent, and S_{r-1} and $\mathrm{prem}(S_{j+1}, S_j)$ are $\mathcal{R}$-linearly dependent.

This corollary can be illustrated by the gap structure of $\mathcal{S}(A, B)$ in Figure 1.

5 Subresultant Algorithm

We shall now extend the subresultant sequences of the first and second kinds in [13]), and describe the subresultant algorithm.

Definition 5.1 *The subresultant sequence of A and B of the first kind* is the subsequence of $\mathcal{S}(A, B)$ that consists of the following polynomials: A, B, and S_j if S_{j+1} is regular and S_j is nonzero. *The subresultant sequence of A and B of the second kind* is the subsequence of $\mathcal{S}(A, B)$ that consists of A, B and the other regular subresultants of $\mathcal{S}(A, B)$. We denote the subresultant sequences of A and B of the first and second kinds by $\mathcal{S}_1(A, B)$ and $\mathcal{S}_2(A, B)$, respectively.

The next lemma describes the relation between $\mathcal{S}_1(A, B)$ and $\mathcal{S}_2(A, B)$.

Lemma 5.1 Let $\mathcal{S}_2(A, B)$ consist of
$$A,\ S_n,\ S_{j_k},\ S_{j_{k-1}}, \cdots, S_{j_1},\ S_{j_0},$$
with $n > j_k > j_{k-1} > \cdots > j_0$. Then $\mathcal{S}_1(A, B)$ consists of
$$A,\ S_n,\ S_{n-1},\ S_{j_k-1},\ S_{j_{k-1}-1}, \cdots, S_{j_1-1}.$$

Proof The sequence $\mathcal{H}$:
$$A,\ S_n,\ S_{n-1},\ S_{j_k-1},\ S_{j_{k-1}-1}, \cdots, S_{j_1-1},$$
is a subsequence of $\mathcal{S}_1(A, B)$ by Definition 5.1. Notice that S_{j_0-1} is zero, for otherwise there would be a regular subresultant S_r with $r = \deg S_{j_0-1}$, but S_{j_0} is the last regular subresultant in $\mathcal{S}(A, B)$, a contradiction. Hence all the subresultants following S_{j_0} are zero because of (24) in Theorem 4.2. The sequence $\mathcal{H}$ is $\mathcal{S}_1(A, B)$. The lemma is proved.

Let the subresultants $S_{j_{i+1}}$ and S_{j_i} be two consecutive members of $\mathcal{S}_2(A, B)$. Then Corollary 4.3 asserts that the subresultants between $S_{j_{i+1}-1}$ and S_{j_i} are all zero. Hence, all the non-zero subresultants are contained in either $\mathcal{S}_2(A, B)$ or $\mathcal{S}_1(A, B)$. Accordingly, all the defective subresultants are contained in $\mathcal{S}_1(A, B)$. The members of $\mathcal{S}_1(A, B)$ and $\mathcal{S}_2(A, B)$ are $\mathcal{R}$-linearly dependent in order. If there is no defective subresultant in $\mathcal{S}(A, B)$, then both $\mathcal{S}_1(A, B)$ and $\mathcal{S}_2(A, B)$ coincide with $\mathcal{S}(A, B)$.

We present the subresultant algorithm in the next theorem, which is an extension of the commutative subresultant algorithm in [4]. The algorithm computes $\mathcal{S}_1(A, B)$ without expanding determinants directly. It proceeds as the Euclidean algorithm but removes an extraneous factor from the coefficients in the pseudo-remainder after each pseudo-division. A byproduct of this algorithm is the leading coefficients of the members of $\mathcal{S}_2(A, B)$, that is, the leading coefficients of all regular subresultants of A and B.

Theorem 5.2 (Subresultant Algorithm)

1. If S_{n-1} is nonzero, the third member of $\mathcal{S}_1(A, B)$ is
$$S_{n-1} = (-1)^{m-n+1}\mathrm{prem}(A, B).$$

2. If S_k is the fourth member of $\mathcal{S}_1(A, B)$ and l is equal to $(n - \deg S_{n-1})$, then
$$S_k = \mathrm{prem}(B, S_{n-1})/e, \tag{28}$$
where $e = (-1)^{l+1}\sigma(\mathrm{lc}(B))^{[m-n][l]}\mathrm{lc}(B)$.

3. If S_i, S_j and S_k are three consecutive members in $\mathcal{S}_1(A, B)$ $(n > i > j > k)$, and l is $(\deg S_i - \deg S_j)$, then
$$S_k = \mathrm{prem}(S_i, S_j)/e, \tag{29}$$
where $e = (-1)^{l+1}\sigma(\mathrm{lc}(S_{j+1}))^{[l]}\mathrm{lc}(S_i)$.

Proof The first assertion is due to Lemma 2.3. The second assertion follows from (27) in Theorem 4.2 (set $j = n - 1$).

To prove the last assertion, we let $\deg S_i = t$ and $\deg S_j = r$. Since both S_{i+1} and S_{j+1} are regular by the definition of $\mathcal{S}_1(A, B)$, $t = j + 1$ by Corollary 4.3. It then follows from (26) in Theorem 4.2 that

$$\sigma(\mathrm{lc}(S_{i+1}))^{[i-t]} S_{j+1} = \sigma(\mathrm{lc}(S_i))^{[i-t]} S_i. \qquad (30)$$

From this equation we see that

$$\begin{aligned} & (\mathrm{lc}(S_i))^{[i-t+1]} \mathrm{prem}(S_i, S_j) \\ = \;& \mathrm{lc}(S_i) \mathrm{prem}(\sigma(\mathrm{lc}(S_i))^{[i-t]} S_i, S_j) \\ = \;& \mathrm{lc}(S_i) \mathrm{prem}(\sigma(\mathrm{lc}(S_{i+1}))^{[i-t]} S_{j+1}, S_j) \\ = \;& \mathrm{lc}(S_i) \sigma(\mathrm{lc}(S_{i+1}))^{[i-t]} \mathrm{prem}(S_{j+1}, S_j) \\ = \;& (-1)^{j-r+2} \sigma(\mathrm{lc}(S_{i+1}))^{[i-t]} \mathrm{lc}(S_{j+1}) \mathrm{lc}(S_i) \\ & \sigma(\mathrm{lc}(S_{j+1}))^{[j+1-r]} S_{r-1} \text{ (by (27))}. \end{aligned}$$

Note that $S_{r-1} = S_k$ since S_r is regular. Thus, the theorem will be proved if

$$\mathrm{lc}(S_i)^{[i-t+1]} = \sigma(\mathrm{lc}(S_{i+1}))^{[i-t]} \mathrm{lc}(S_{j+1}) \qquad (31)$$

holds. Equating the leading coefficients on both sides of (30) yields (31). The theorem is proved.

By Theorem 5.2 we can design the subresultant algorithm for computing $\mathcal{S}_1(A, B)$. In order to compute S_k from S_i and S_j according to the third assertion of Theorem 5.2, we need to compute $\mathrm{lc}(S_{j+1})$. This leading coefficient can be obtained recursively from (31).

Corollary 5.3 Both $\mathcal{S}_1(A, B)$ and $\mathcal{S}_2(A, B)$ are p.r.s.'s.

Proof $\mathcal{S}_1(A, B)$ is a p.r.s. by Theorem 5.2. $\mathcal{S}_2(A, B)$ is a p.r.s. because each member of $\mathcal{S}_2(A, B)$ is $\mathcal{R}$-linearly dependent on one (and only one) member of $\mathcal{S}_1(A, B)$ by Lemma 5.1.

6 Applications

First, we extend the resultant of two differential operators given in [1, 5].

Definition 6.1 The subresultant S_0 is called *Sylvester's resultant* of A and B and denoted by $\mathrm{res}\,(A, B)$.

Although the first two statements of the following proposition were proved in [8], the proofs given below are much shorter due to Theorems 4.2 and 5.2.

Proposition 6.1 Let d be the degree of the gcrd of A and B. Then the following hold.

1. S_d is a gcrd of A and B.
2. $d = 0$ if and only if $\mathrm{res}\,(A, B) \neq 0$.
3. UA is an lclm of A and B, where U is the determinant of order $(m + n - 2d + 2)$ whose first $(m + n - 2d + 1)$ columns are the first $(m + n - 2d + 1)$ columns of

   $$\mathrm{mat}(X^{n-d}A, \ldots, XA, A, X^{m-d}B, \ldots, XB, B)$$

   and whose last column is the transpose of the vector

   $$(X^{n-d}, \ldots X, 1, \underbrace{0, 0, \ldots, 0}_{m-d+1}).$$

Proof Since $\mathcal{S}_2(A, B)$ is a p.r.s., the last member of $\mathcal{S}_2(A, B)$ is a gcrd of A and B. This member is a regular subresultant, so, it must be S_d. In particular, $d = 0$ iff $S_0 \neq 0$. The first and second assertions are proved.

To prove the last assertion, we first show that UA is right divisible by B. Let M be the matrix whose first $(m + n - 2d + 1)$ columns are the same as those of U, and its last column is the transpose of the vector

$$(X^{n-d}A, \ldots, XA, A,\ X^{m-d}B, \ldots, XB, B).$$

Then $|M|$ is zero because $|M| = S_{d-1}$ when $d > 0$, and the last column of M can be reduced to zero by the previous columns when $d = 0$. Expanding $|M|$ according to its last column yields $UA + VB = 0$, for some $V \in \mathcal{R}[X]$, that is, UA is right divisible by B.

By [11, p. 486] the degree of lclm's of A and B is equal to $(m + n - d)$. Hence, it suffices to prove that the degree of U is equal to $(n - d)$, that is, the cofactor of X^{n-d} in the determinant U is nonzero. From the definition of U one sees that this cofactor is the product of $\pm\sigma^{m-d}(\mathrm{lc}(B))$ and the coefficient of X^d in S_d. As S_d is a gcrd of A and B, the degree of S_d is d. Hence, the cofactor is nonzero.

Example 3 Let $D = \frac{d}{dt}$, and let

$$A = (t - 1)D^3 + (-t^2 + t)D^2 + (-2t + 3)D - t$$

and

$$B = 3D^2 + (t^3 - 3t)D - t^4 - 3$$

be in $\mathbf{Z}[t][D]$. By Proposition 6.1, a gcrd of A and B is

$$\begin{vmatrix} t-1 & -t^2+t & A \\ 3 & -3t+t^3 & DB \\ 0 & 3 & B \end{vmatrix} = (t^7 - t^6 + 9 - 9t^3 + 9t^2)(D - t)$$

and an lclm of A and B is

$$\begin{vmatrix} t-1 & -t^2+t+1 & -4t+4 & -t-2 & D \\ 0 & t-1 & -t^2+t & -2t+3 & 1 \\ 3 & -3t+t^3 & -9-t^4+6t^2 & -8t^3+6t & 0 \\ 0 & 3 & -3t+t^3 & -6-t^4+3t^2 & 0 \\ 0 & 0 & 3 & -3t+t^3 & 0 \end{vmatrix} A.$$

Remark 1 The determinant formula for gcrd's is a generalization of the corresponding formula for gcd's in [12]. The determinant formula for lclm's is possibly new. With these two formulas, one may estimate the coefficient and degree bounds for gcrd's and lclm's.

Remark 2 If $\mathcal{R}$ is a unique factorization domain, then

$$\sigma^{m-d}(\mathrm{lc}(B))\sigma^{n-d}(\mathrm{lc}(A))\mathrm{lc}(S_d)$$

is a multiple of the leading coefficient of the primitive lclm of A and B, because UA is an lclm of A and B and the leading coefficient of U is $\sigma^{m-d}(\mathrm{lc}(B))\mathrm{lc}(S_d)$.

In the rest of this section we study algorithms for computing lclm's. Computing lclm's is not as simple as computing lcm's in the commutative case because of the non-commutativity of multiplication. One method for computing lclm's is the extended (right) Euclidean algorithm [3]. Of course, it is sufficient to use the half-extended Euclidean algorithm, which computes only one co-sequence. If one uses the polynomial division over the fraction field of $\mathcal{R}$ to carry out the half-extended Euclidean algorithm, the algorithm is inefficient because there are too many gcd-computations among coefficients. We present a fraction-free version of the half-extend Euclidean algorithm for computing lclm's, which reduces coefficient growth by exact division over $\mathcal{R}$.

For a later convenience we modify our notation. Let $A_1 = A$, $A_2 = B$ and $\mathcal{S}_1(A, B)$ be the sequence:

$$A_1, A_2, A_3, \ldots, A_k.$$

Let $U_1 = 1$ and $U_2 = 0$. If A_i is the jth subresultant of A and B, then we let U_i be the $(m+n-2j) \times (m+n-2j)$ determinant whose first $(m+n-2j-1)$ column is the same as those in

$$\mathrm{mat}(X^{n-j-1}A, \ldots, A, X^{m-j-1}B, \ldots, B).$$

and whose last column is the transpose of

$$(X^{n-j-1}, \ldots X, 1, \underbrace{0, 0, \ldots, 0}_{m-j}).$$

Note that the index j decreases as the index i increases. The sequence

$$U_1, U_2, U_3, \ldots, U_k$$

is called the first co-sequence associated with $\mathcal{S}_1(A, B)$, because $U_iA_1 \equiv A_i \bmod A_2$ for $i = 1, 2, \ldots, k$, i.e., $U_iA_1 - A_i$ is right-divisible by A_2. Another property of the first co-sequence is that $\deg U_i < (n - \deg A_i)$, for $i = 3, \ldots, k$, because $\deg A_i \leq j$ if A_i is the jth subresultant of A and B. By Theorem 5.2 we have, for $i = 3, 4, \ldots, k$,

$$l_iA_{i-2} = Q_iA_{i-1} + e_iA_i,$$

where

$$l_i = \mathrm{lc}(A_{i-1})^{[\deg A_{i-2} - \deg A_{i-1} + 1]},$$

e_3 is $(-1)^{m-n+1}$ and e_i is the extraneous factor e removed in (28) or (29) when we use the subresultant algorithm to compute A_i.

The next proposition provides a fraction-free version of the half-extended Euclidean algorithm.

Proposition 6.2 With the notation just introduced, we have

$$U_i = (l_{i-1}U_{i-2} - Q_iU_{i-1})/e_i \tag{32}$$

for $i = 3, \ldots, k$. Furthermore

$$L = (l_kU_{k-1} - Q_kU_k)A_1 \tag{33}$$

is an lclm of A_1 and A_2.

Proof Let $V_1 = U_1$ and $V_2 = U_2$, and let

$$V_i = (l_{i-1}V_{i-2} - Q_iV_{i-1})/e_i, \tag{34}$$

for $i = 3, 4, \ldots, k$.

Since $l_iA_{i-2} = Q_iA_{i-1} + e_iA_i$, $V_iA_1 \equiv A_i \bmod A_2$, for $i = 3, \ldots, k$. Now, we prove $U_i = V_i$, for $i = 3, \ldots, k$. Notice that $(U_i - V_i)A_1$ is right divisible by A_2, and that (34) implies $\deg V_i = (\deg A_2 - \deg A_{i-1})$. Recall $\deg U_i < (\deg A_2 - \deg A_i)$. We conclude

$$\deg(U_i - V_i) < (\deg A_2 - \deg A_k).$$

It then follows that $U_i - V_i = 0$, for otherwise $(U_i - V_i)A_1$ would be a non-zero common left multiple of A_1 and A_2 with degree less than $(\deg A_1 + \deg A_2 - \deg A_k)$, the degree of lclm's of A_1 and A_2, a contradiction. The formula (32) holds. Since U_i is in $\mathcal{R}[X]$, the division in (32) is exact.

In particular, we have

$$U_{k-1}A_1 \equiv A_{k-1} \bmod A_2, \quad U_kA_1 \equiv A_k \bmod A_2,$$

and $l_kA_{k-1} = Q_kA_k$. Hence L defined in (33) is right-divisible by A_2. It remains to show

$$\deg L = \deg A_1 + \deg A_2 - \deg A_k.$$

This follows from the observation that $\deg U_{k-1} = (\deg A_2 - \deg A_{k-2})$, $\deg U_k = (\deg A_2 - \deg A_{k-1})$, and $\deg Q_k = (\deg A_{k-1} - \deg A_k)$. The proposition is proved.

We present an experimental comparison among three algorithms (labeled L, S, and M, respectively) for computing lclm's of two Ore polynomials A and B in $\mathcal{R}[X]$, where $\deg A = m$ and $\deg B = n$. For a matrix H over $\mathcal{R}$, we denote by (H) the linear homogeneous system whose matrix is H. Let M be the matrix

$$\mathrm{mat}(X^nA, \ldots, A, X^mB, \ldots, B)$$

and M^τ be the transpose of M.

The idea of Algorithm L goes as follows. First, compute a triangular form T of M^τ by row-reduction. Second, extract the submatrix T_r from T whose columns correspond to the polynomials $X^{n-r}A, \ldots, A, X^{m-r}B, \ldots, B$, for $r = 1, 2, \ldots, m-n$, and decide if (T_r) has a non-trivial solution. If $(r-1)$ is the largest index such that (T_{r-1}) has only trivial solution, then any nontrivial solution of (T_r) gives us the lclm. Algorithm L has two time-consuming costs:

- compute a triangular form of the matrix M^τ by row reduction
- Find a linear relation (over $\mathcal{R}$) among the polynomials $X^{n-r}A, \ldots, A, X^{m-r}B, \ldots, B$ with r as large as possible.

Algorithm S is given in Proposition 6.2, which has two time-consuming costs:

- compute $\mathcal{S}_1(A, B): A_1, A_2, \ldots, A_k$
- compute the 1st co-sequence associated with $\mathcal{S}_1(A, B)$.

Algorithm M is based on Proposition 6.1, which has two costs:

- compute the degree of gcrd's of A and B
- expand the determinant U given in Proposition 6.1.

If the gcrd of A and B is trivial, then both Algorithms L and S compute some triangular forms of the matrices M^τ and M, respectively. However, Algorithm S makes use of the special structure of M and avoids solving any linear system, although it has the additional cost for computing the first co-sequence. Furthermore, Algorithm S avoids any gcd-computation in $\mathcal{R}$ and controls the growth of coefficients by dividing out extraneous factors.

If the gcrd of A and B is of degree $d > 0$, then Algorithm S only needs to compute a triangular form of the matrix

$$\mathrm{mat}(X^{n-d}A, \ldots, A, X^{m-d}B, \ldots, B).$$

But Algorithm L still needs to compute a triangular form of M^τ.

If the degree of the gcrd of A and B can be computed efficiently, then Algorithm M needs only to expand one determinant of order $(m+n-2d+2)$. For example, if $\mathcal{R} = \mathbf{Z}[t]$, then the gcrd of A and B can be computed efficiently by the modular method in [8].

We compared the function `LCLM` in the Maple package `diffop` by Mark van Hoeij, which appeared to use the idea of Algorithm L, with our Maple implementations of Algorithm S and Algorithm M. We generated three random polynomials in $\mathbf{Z}[t][D]$, A, B and C, with respective degrees d_A, d_B, and d_C in X, regarded A, B and C as linear differential operators, and computed the primitive lclm of AC and BC. In Algorithm M the degree of the gcrd of AC and BC was computed by the modular algorithm in [8].

Our experiment was carried out in Maple V (Release 3) on an Alpha-workstation. The random polynomials in the experiment were generated by the Maple function `randpoly`. When $d_C = 0$, we set $C = 1$. Some of the timings are summarized in Figure 2, in which the column labeled l gives the average maximal length of the integral coefficients of AC and BC, and the column labeled d_t gives the average degree of AC and BC in the variable t. All the timings are Maple CPU time and given in seconds.

$d_A + d_C$	$d_B + d_C$	d_C	d_t	l	L	S	M
5	3	0	4	2	5.9	1.0	1.0
5	3	1	9	6	31.4	2.7	2.6
5	3	2	8	6	52.9	1.0	0.2
5	4	0	4	2	11.2	3.2	3.1
5	4	1	8	6	61.2	8.0	7.8
5	4	2	8	6	82.8	2.3	2.2
5	4	3	9	5	125.2	0.5	0.3
5	5	0	4	2	15.3	5.3	5.1
5	5	1	8	6	100.9	18.1	18.1
5	5	2	8	6	138.2	6.0	6.1
5	5	3	8	5	228.2	2.6	2.2
5	5	4	9	5	314.1	0.7	0.1

Figure 2: Times for computing lclm's

The timings show that Algorithms S and M tend to be faster than Algorithm L when d_C increases. But I think that the difference between Algorithm L and Algorithms M and S should be smaller when d_C is zero, because, in this case, the most time-consuming computation in the three algorithms is to triangularize the matrix M. For the time being, my explanation on this difference is that there is about one-third of computing time of `LCLM` spent on other costs, since `LCLM` works for various coefficient domains. I expect that better understanding of the function `LCLM` in the package `diffop`, and functions `solve` and `linsolve` in Maple would result in a clear explanation of the timings, and lead to a better way of designing experimental data.

Acknowledgment

I thank Dr. Chardin for the discussion on differential subresultants, and thank Dr. van Hoeij for putting his package `diffop` on the internet.

References

[1] Berkovich, M., Tsirulik, G., *Differential Resultants and Some of Their Applications,* Differential'nye Uravneniya, **22**, No. 5, (1986), 750-757.

[2] Bronstein, M., Petkovšek, M., *On Ore Rings, Linear Operators and Factorisation,* Programming and Comput. Software, **20**, (1994), 14–26.

[3] Bronstein, M., Petkovšek, M., *An introduction to pseudo-linear algebra,* Theoretical Computer Science, **157**, (1996), 3-33.

[4] Brown, S., Traub, F., *On the Euclid's Algorithm and the Theory of Subresultants,* J. ACM, **18,** (1971), 505-514.

[5] Chardin, M., *Differential Resultants and Subresultants*, Proceedings of Fundamentals of Computation Theory, Lecture Notes in Computer Science, **529**, (1991), 180-189.

[6] Collins, G., *Subresultant and Reduced Polynomial Remainder Sequences*, J. ACM, **16**, (1967), 708-712.

[7] Li, Z., *A Subresultant Theory for Linear Differential, Linear Difference, and Ore Polynomials with Applications*, PhD Thesis, Technical Report 96-14, Research Institute for Symbolic Computation, Johannes Kepler University, Linz, A-4040, Austria, 1996.

[8] Li, Z., Nemes, I., *A Modular Algorithm for Computing Greatest Common Right Divisors of Ore Polynomials,* Proceedings of the 1997 International Symposium on Symbolic and Algebraic Computation, ACM Press, (1997), 282-289.

[9] Loos, R., *Generalized Polynomial Remainder Sequence,* in Computer Algebra, Symbolic and Algebraic Computation, Buchberger, B., Collins, G., Loos, R., (eds.), Springer-Verlag, Wien-New York, (1982), 115-137.

[10] Mishra, B., Algorithmic Algebra Texts and Monographs in Computer Science, Springer-Verlag, 1993.

[11] Ore, O., *Theory of Non-Commutative Polynomials,* Annals of Mathematics, **34**, (1933), 480-508.

[12] Perron, O., Algebra I die Grundlagen, Berlin: de Gruyter & Co., 1951.

[13] Rubald, M., *Algorithms for Polynomials over a Real Algebraic Number Field,* PhD Thesis, Department of Computer Science, University of Wisconsin, 1973.