On the complexity of skew arithmetic

Joris van der Hoeven

On the complexity of skew arithmetic*

Joris van der Hoeven

LIX, CNRS
École polytechnique
91128 Palaiseau Cedex
France

Email: vdhoeven@lix.polytechnique.fr
Web: http://lix.polytechnique.fr/~vdhoeven

September 9, 2011

In this paper, we study the complexity of several basic operations on linear differential operators with polynomial coefficients. As in the case of ordinary polynomials, we show that these complexities can be expressed in terms of the cost of multiplication.

Keywords: Linear differential operators, algorithm, complexity, multiplication, local solution, division, gcd, lcm

A.M.S. subject classification: 68W30, 68Q15, 34M03, 12E15

1. Introduction

Let $\mathbb{K}$ be an effective field of constants of characteristic zero, so that all field operations can be carried out by algorithms. Given an indeterminate x and the derivation $\delta = x\,\partial$, where $\partial = \partial/\partial x$, we will study various operations in the skew ring $\mathbb{K}[x,\delta]$, such as multiplication, division, greatest common divisors, series solutions, etc. In analogy with the commutative case, we will give bounds for the computational complexities of these operations in terms of the complexity of operator multiplication.

For our complexity measures, it is convenient to assume that all field operations can be carried out in constant time $O(1)$. We will try to express the complexities of our algorithms in terms of the following standard complexities:

- The time $\mathsf{M}(n)$ required for the multiplication of two polynomials of degrees $<n$ and coefficients in $\mathbb{K}$. It is classical [6] that $\mathsf{M}(n) = O(n \log n \log\log n)$ and $\mathsf{M}(n) = O(n \log n)$ if $\mathbb{K}$ admits sufficiently many 2^p-th roots of unity [7].
- The complexity $O(r^\omega)$ of multiplying two $r \times r$ matrices with entries in $\mathbb{K}$. It is classical [13, 11, 8] that $\omega < 2.376$, although $\omega \approx 3$ in practice.

We will denote by $\mathbb{K}[x]_n$ the subset of $\mathbb{K}[x]$ of polynomials of degree $<n$. Likewise, we denote by $\mathbb{K}[x,\delta]_{n,r}$ the set of operators $L \in \mathbb{K}[x,\delta]$ of degree $\deg_x L < n$ in x and degree $\deg_\delta L < r$ in δ.

Now consider two linear differential operators $K, L \in \mathbb{K}[x,\delta]_{n,r}$. We start with studying the following complexities:

- The complexity $\mathsf{SM}(n,r)$ of multiplying K and L.
- The complexity $\mathsf{SV}(n,r)$ of applying L to a vector of r polynomials in $\mathbb{K}[x]_n$.

*. This work has been supported by the ANR-09-JCJC-0098-01 MaGiX project, as well as a Digiteo 2009-36HD grant and Région Ile-de-France.

- The cost $\mathsf{SF}(n, r)$ to compute a fundamental system of r solutions to the monic equation $(\delta^r + L)\, f = 0$ in $\mathbb{K}[[x]]$, up to order $O(x^n)$, while assuming the existence of such a fundamental system.
- Given a vector V of r truncated power series in $\mathbb{K}[x]$, the cost $\mathsf{SA}(n, r)$ of computing a monic operator in $A = \delta^r + \mathbb{K}[x, \delta]_{n,r}$ with $A(V) = O(x^n)$.

The special case $n = r$ was first studied in [15], where it was shown that $\mathsf{SM}(n, n) = O(n^\omega)$, using evaluation-interpolation techniques. The inverse bound $n^\omega = O(\mathsf{SM}(n, n))$ has been proved in [5]; this paper also contains detailed information on the constant factors involved in these bounds.

For fixed constants $\alpha,\ \beta > 0$, we notice that $\mathsf{M}(\alpha\ n) = O(\mathsf{M}(n))$, $(\beta\ r)^\omega = O(r^\omega)$, $\mathsf{SM}(\alpha\, n, \beta\, r) = O(\mathsf{SM}(n,\ r))$, etc. In this paper, we will freely use this remark without further mention. In order to simplify our complexity estimates, it will be convenient to make a few additional assumptions. First of all, we will assume that $\omega > 2$, whence in particular $\mathsf{M}(n) \log n = \mathcal{O}(n^{\omega-1})$. We will also assume that the function $\mathsf{M}(n)/n$ is increasing and that $\mathsf{SM}(n, r)/(n\, r)$ is increasing in both n and r. This will indeed be the case for the complexity bounds for $\mathsf{SM}(n, r)$ that will be given in section 3.

In section 2, we will first prove (see theorems 2 and 3) that the problems of multiplication and operator-vector application are essentially equivalent when $n \geqslant r$. In section 3, we recall the best available bounds in the case when $n \neq r$. It remains an open question whether these bounds are optimal.

In section 4, we show that the problems of computing fundamental systems of solution and its inverse can be reduced to operator multiplication modulo a logarithmic overhead (see theorems 13 and 14). This provides a dual way to perform operations on differential operators by working on their fundamental systems of solutions. In section 5, we start with the operations of exact division and division with remainder. In section 6, we consider greatest common divisors and least common multiples. Again, we will show how to express the complexities of these operations essentially in terms of the complexity $\mathsf{SM}(n,\ r)$ of multiplication (see theorems 19, 21, 23 and 26).

To the best of our knowledge, the idea to perform operations on linear differential operators *via* power series solutions was first proposed (but only partially worked out) in [3, Chapter 10]. We were independently aware of this possibility and prefer the use of fundamental systems of solutions (so as to force a clean bijection between operators and solution spaces). It is also possible to mimic classical divide and conquer algorithms for division, greatest common divisors and least common multiples, while using adjoints in order to perform the recursive operations on the appropriate side. Such algorithms were implemented inside MATHEMAGIX [18] and we plan to analyze them in a forthcoming paper.

2. EVALUATION AND INTERPOLATION

The key argument behind the proof from [15] that $\mathsf{SM}(n,\ n) = O(n^\omega)$ is the observation that an operator $L \in \mathbb{K}[x,\ \delta]_{n,r}$ is uniquely determined by its images on the vector $x^{;r} = (1, ..., x^{r-1})$. This makes it possible to use a similar evaluation-interpolation strategy for the multiplication of differential operators as in the case of FFT-multiplication of commutative polynomials. More precisely, given $L \in \mathbb{K}[x,\ \delta]_{n,r}$, let $\Phi_L^{r+n,r}$ be the matrix of the mapping $\mathbb{K}[x]_r \to \mathbb{K}[x]_{r+n};\ P \mapsto L(P)$ with respect to the bases $x^{;r}$ and $x^{;r+n}$:

$$\Phi_L^{r+n,r} = \begin{pmatrix} L(1)_0 & \cdots & L(x^{r-1})_0 \\ \vdots & & \vdots \\ L(1)_{r+n-1} & \cdots & L(x^{r-1})_{r+n-1} \end{pmatrix}.$$

LEMMA 1. *Let $L \in \mathbb{K}[x,\delta]_{n,r}$. Then*

a) We may compute $\Phi_L^{r+n,r}$ as a function of L in time $O(n\,\mathsf{M}(r)\log r)$.

b) We may recover L from $\Phi_L^{r+n,r}$ in time $O(n\,\mathsf{M}(r)\log r)$.

Proof. Consider the expansion of L with respect to x

$$L(x,\delta) \;=\; L_0(\delta) + \cdots + x^{n-1}\,L_{n-1}(\delta).$$

For all i, j, we have

$$\begin{aligned} L(x,\delta)(x^j)_{i+j} &= [x^i\,L_i(\delta)](x^j)_{i+j} \\ &= [x^{i+j}\,L_i(\delta+j)(1)]_{i+j} \\ &= L_i(j). \end{aligned}$$

In other words, $\Phi_L^{r+n,r}$ is a lower triangular band matrix

$$\Phi_L^{r+n,r} \;=\; \begin{pmatrix} L_0(0) & & \\ \vdots & \ddots & \\ L_{n-1}(0) & & L_0(r-1) \\ & \ddots & \vdots \\ & & L_{n-1}(r-1) \end{pmatrix}$$

of bandwidth $\leqslant n$. The coefficients on the i-th subdiagonal of $\Phi_L^{r+n,r}$ are exactly the result of a multipoint evaluation of L_i at 0, ..., $r-1$. It is classical [10, 14, 2] that both multipoint evaluation and the inverse operation of interpolation can be performed in time $\mathcal{O}(\mathsf{M}(r)\log r)$. Doing this for each of the polynomials $L_0, \ldots, L_{n-1}$ yields the result. □

THEOREM 2. *If $n \geqslant r$, then*

$$\mathsf{SM}(n,r) \;=\; O(\mathsf{SV}(n,r) + n\,\mathsf{M}(r)\log r) \tag{1}$$

Proof. Let $K, L \in \mathbb{K}[x,\delta]_{n,r}$ and assume that we want to compute $K\,L$. We may evaluate $L(x^{;2r})$ in time $\mathsf{SV}(\max(n, 2\,r), 2\,r) = O(\mathsf{SV}(n,r))$. We may also evaluate $K(L(x^{;2r}))$ in time $\mathsf{SV}(n+2\,r, 2\,r) = O(\mathsf{SV}(n,r))$. Using the lemma, we may recover $K\,L$ from $K(L(x^{;2r}))$ in time $O(n\,\mathsf{M}(r)\log r)$. This completes the proof. □

THEOREM 3. *If $n \geqslant r$, then*

$$\mathsf{SV}(n,r) \;=\; O(\mathsf{SM}(n,r) + n\,\mathsf{M}(r)\log r). \tag{2}$$

Proof. Assume now that we are given $K(x,\,\delta) \in \mathbb{K}[x,\,\delta]_{n,r}$, as well as a vector $V = (V_0, \ldots, V_{r-1}) \in \mathbb{K}[x]_n^r$ and that we want to evaluate $K(V) = (K(V_0), \ldots, K(V_{r-1}))$. This is equivalent to evaluating the operator $K^* = K(x, \delta - r)$ at the vector $x^r\,V$. It is classical [1] that K^* can be computed in time $O(n\,\mathsf{M}(r))$. Using the lemma, we may compute the unique operator $L \in \mathbb{K}[x,\delta]_{n+r,r}$ with $L(x^{;r}) = x^r\,V$ in time $O((n+r)\,\mathsf{M}(r)\log r) = O(n\,\mathsf{M}(r)\log r)$. We may next compute the product $K^*\,L$ in time $\mathsf{SM}(n+r, r) = O(\mathsf{SM}(n,r))$. The lemma finally allows us to evaluate $K^*\,L$ at $x^{;r}$ in time $O(n\,\mathsf{M}(r)\log r)$, thereby yielding $K(V)$. □

3. BASIC COMPLEXITY BOUNDS

Let us review the best know algorithms (asymptotically speaking, and up to constant factors) for the multiplication of linear differential operators and for the evaluation of linear differential operators at vectors of polynomials. We will treat the cases $n \geqslant r$ and $r \geqslant n$ separately.

3.1. Large degrees $n \geqslant r$

THEOREM 4. *If $n \geqslant r$, then*

$$\mathsf{SM}(n, r) \;=\; O(\mathsf{M}(n\,r)\,r).$$

Proof. Consider two operators $K = \sum_{i,j} K_{i,j}\, x^j\, \delta^i$, $L = \sum_{i,j} L_{i,j}\, x^j\, \delta^i \in \mathbb{K}[x, \delta]_{n,r}$. Then we may compute their operator product using the formula

$$K\,L \;=\; \sum_k \frac{1}{k!} \left(\frac{\partial}{\partial \delta}\right)^k K * \left(\frac{x\,\partial}{\partial x}\right)^k L, \tag{3}$$

attributed to Takayama, where

$$K * L \;=\; \sum_{i,j,i',j'} K_{i,j}\, L_{i',j'}\, x^{j+j'}\, \delta^{i+i'}$$

denotes the commutative product of K and L. Commutative products can be computed in time $O(\mathsf{M}(n\,r))$ using Kronecker substitution [12, 9], whence the result follows.

In practice, it is often possible and best to compute the commutative products using bivariate FFT multiplication. In that case several of the FFT-transforms can be shared. For instance, if $n \geqslant r$, then the most expensive step is to compute the r^2 transforms with respect to x of the r coefficients in δ of the first r derivatives $(x\,\partial/\partial x)^k\, L$. For large n, this optimization yields an algorithm of complexity $\sim(1/3)\, M(n)\, r^2$. □

THEOREM 5. *If $n \geqslant r$, then*

$$\mathsf{SV}(n, r) \;=\; O(\mathsf{M}(n\,r) \min(n, r)).$$

Proof. This is a direct consequence of theorems 3 and 4.

In practice, in the domain where FFT-multiplication is most efficient, it is better to use a more direct method to obtain this result. Given $L \in \mathbb{K}[x, \delta]_{n,r}$ and $V \in \mathbb{K}[x]^r_n$, we use the following algorithm:

- Compute the r^2 FFT-transforms of $\delta^j\, V_i$ with $i, j < r$ and the r FFT-transforms of the coefficients of L with respect to δ.
- From these values, deduce the FFT-transforms of the r entries of $L(V)$ using $2\,n - 1$ scalar $r \times r$ matrix-vector multiplications.
- Recover $L(V)$ using r inverse transforms.

This algorithm has a complexity $\sim(1/3)\, M(n)\, r^2$, for large n. □

If $n = O(r^{4-\omega})$, then the following result becomes more efficient for the evaluation of linear differential operators at vectors of polynomials:

THEOREM 6. *If $n \geqslant r$, then we have*

$$\mathsf{SV}(n, r) \;=\; O(n^2\, r^{\omega-2}).$$

Proof. We may cut both L and V into $O(n/r)$ pieces in $\mathbb{K}[x, \delta]_{r,r}$ and $\mathbb{K}[x]^r_r$. Hence $\mathsf{SV}(n, r) = O((n/r)^2\, \mathsf{SV}(r, r)) = O(n^2\, r^{\omega-2})$. Here we repeatedly use the commutation rule $L(x, \delta)\, x^k = x^k\, L(x, \delta + k)$, when considering $L(x, \delta)$ and x^k as operators. The twist $L(x, \delta) \mapsto L(x, \delta + k)$ can be computed in time $O(r\, \mathsf{M}(r))$, for $L \in \mathbb{K}[x, \delta]_{r,r}$ [1]. □

If both $n = O(r^{4-\omega})$, this yields the following more efficient algorithm for the multiplication of linear differential operators:

THEOREM 7. *If $n \geqslant r$, then we have*

$$\mathsf{SM}(n,r) \;=\; O(n^2\, r^{\omega-2}).$$

Proof. Direct consequence of theorems 2 and 6. □

We recall from [5] that $n^\omega = O(\mathsf{SM}(n,n))$. More generally, we have:

THEOREM 8. *If $n \geqslant r$, then the product of an $r \times n$ matrix and an $r \times r$ matrix with coefficients in $\mathbb{K}$ can be computed in time $O(\mathsf{SM}(n,r))$.*

Proof. By the result from [5], the problem is equivalent to the computation of $k = \lceil n/r \rceil$ operators K_0, ..., K_{k-1} in $\mathbb{K}[x,\delta]_{r,r}$ with a fixed operator $L \in \mathbb{K}[x,\delta]_{r,r}$. Setting $K = K_0 + x^{2r}\, K_1 + \cdots + x^{2r(k-1)}\, K_{k-1}$, we may compute $K\, L$ in time $O(\mathsf{SM}(n,r))$. We may directly read off the products $K_0\, L, ..., K_{k-1}\, L$ from the result. □

Remark 9. An interesting open problem at the time of writing concerned the existence of a better bound for $\mathsf{SM}(n,\, r)$ than the one given in theorem 7. In collaboration with Benoit and Bostan, we have recently been able to prove the sharper bound $\mathsf{SM}(n,\, r) = O(r^{\omega-1}\, n + r\, \mathsf{M}(n) \log n)$ for the case when $n \geqslant r$.

3.2. Large orders $r \geqslant n$

THEOREM 10. *If $r \geqslant n$, then*

$$\mathsf{SM}(n,r) \;=\; O(\mathsf{M}(r)\, n^2).$$

Proof. Let $K, L \in \mathbb{K}[x,\delta]_{n,r}$ and consider the expansion

$$L(x,\delta) \;=\; L_0(\delta) + \cdots + x^{n-1}\, L_{n-1}(\delta)$$

of L in δ. Then we have

$$\begin{aligned} K(x,\delta)\, L(x,\delta) \;&=\; \sum_{k<n} K(x,\delta)\, (x^k\, L_k(\delta)) \\ &=\; \sum_{k<n} x^k\, K(x,\delta+k)\, L_k(\delta) \end{aligned}$$

For each k, both the Taylor shift $K(x,\delta+k)$ and the product $K(x,\delta+k)\, L_k(\delta)$ can be computed in time $O(\mathsf{M}(r)\, n)$ [1]. □

THEOREM 11. *If $r \geqslant n$, then*

$$\mathsf{SV}(n,r) \;=\; O(n^{\omega-1}\, r).$$

Proof. Let $L \in \mathbb{K}[x,\delta]_{n,r}$ and $V \in \mathbb{K}[x]^r_n$. We may compute $L(x^{:n})$ in time $O(n^{\omega-1}\, r)$, since this really amounts to the computation of $O(r/n)$ matrix products of size $n \times n$. Writing $V = M\, x^{:r}$ for a constant $n \times r$ matrix M, we may thus compute $L(V) = M\, L(x^{:n})$ in time $O(n^{\omega-1}\, r)$. □

THEOREM 12. *If $r \geqslant n$, then*

$$\mathsf{SM}(n,r) \;=\; O(n^{\omega-1}\, r + n\, \mathsf{M}(r) \log r)$$

Proof. Let $K, L \in \mathbb{K}[x,\delta]_{n,r}$. By lemma 1, we may compute $\Phi_K^{2r+2n,2r+n}$ and $\Phi_L^{2r+n,2r}$ in time $O(n\,\mathsf{M}(r)\log r)$. Since both of these matrices are band matrices of bandwidths $\leqslant n$, we may compute the product

$$\Phi_{KL}^{2r+2n,2r} \;=\; \Phi_K^{2r+2n,2r+n}\,\Phi_L^{2r+n,2r}$$

in time $O(n^{\omega-1}\,r)$. Again by lemma 1, we may reconstruct $K\,L$ from $\Phi_{KL}^{2r+2n,2r}$ in time $O(n\,\mathsf{M}(r)\log r)$. □

4. Local solutions

From now on, we will assume that $n \geqslant r$. We recall that an operator $L \in \mathbb{K}[x,\partial]$ of order r is said to be *non singular* at x_0, if its leading coefficient L_r does vanishes at x_0. We will say that an operator $L \in \mathbb{K}[x,\delta]$ of order r is non singular (at the origin) if $x^{-r}\,L \in \mathbb{K}[x,\partial]$ and $x^{-r}\,L$ is non singular as an operator in ∂.

Given a non singular differential operator $L \in \mathbb{K}[x,\,\delta]_{n,r+1}$ of order r, the equation $L(H)=0$ admits a *canonical* fundamental system $H=(H_0,\ldots,H_{r-1})$ of solutions in $\mathbb{K}[[x]]^r$, with the property that $(H_i)_i=1$ and $(H_i)_j=0$ for all $i,j<r$ with $i\neq j$. Conversely, given a $\mathbb{K}$-linearly independent vector of power series $H \in \mathbb{K}[[x]]^r$, there exists a unique monic operator $L \in \delta^r + \mathbb{K}[[x]][\delta]$ of order r with $L(H)=0$. Let us show how to convert efficiently between these two representations.

THEOREM 13. *Let $L \in \mathbb{K}[x,\,\delta]_{n,r+1}$ be a differential operator of order r, which is non singular at the origin, and let H be its canonical fundamental system of solutions. Then we may compute H up to order $O(x^n)$ in time $O(\mathsf{SV}(n,r)\log n)$. In other words,*

$$\mathsf{SF}(n,r) \;=\; O(\mathsf{SM}(n,r)\log n). \tag{4}$$

Proof. Modulo multiplying L on the left by L_r^{-1}, we may assume without loss of generality that L is monic. Since L is non singular at the origin, we have $x^{-r}\,L \in \mathbb{K}[x,\partial]$. Rewritten in terms of δ, this means that L is of the form

$$\begin{aligned} L &= \Delta_r(\delta) + x\,C_{r-1}\,\Delta_{r-1}(\delta) + \cdots + x^r\,C_0\,\Delta_0(\delta).\\ \Delta_k(\delta) &= \delta\,(\delta-1)\cdots(\delta-k+1), \end{aligned}$$

for certain $C_0,\ldots,C_{r-1} \in \mathbb{K}[x]$. Setting $R = \Delta_r(\delta) - L \in x\,\mathbb{K}[x,\delta]_{n-1,r}$, we observe that R maps $\mathbb{K}[[x]]$ into $x^r\,\mathbb{K}[[x]]$. We now compute H using the "recursive" formula

$$H \;=\; \begin{pmatrix} 1 \\ \vdots \\ x^{r-1} \end{pmatrix} + \Delta_r(\delta)^{-1}(R(H)), \tag{5}$$

where

$$\Delta_r(\delta)^{-1}\left(\sum_{k\geqslant r} A_k\,x^k\right) \;=\; \sum_{k\geqslant r} \frac{A_k}{\Delta_r(k)}\,x^k.$$

The equation (5) is a schoolbook example for applying the strategy of relaxed resolution of power series equations [16, 17]. Since $\Delta_r(\delta)^{-1}$ operates coefficientwise, it can be computed in linear time. The main cost of the computation therefore reduces to the relaxed evaluation of $R(H)$. Using fast relaxed multiplication, this amounts to a cost

$$\mathsf{SF}(n,r) \;=\; 2\,\mathsf{SV}(\lceil n/2\rceil,r) + 4\,\mathsf{SV}(\lceil n/4\rceil,r) + \cdots + n\,\mathsf{SV}(1,r).$$

Using the monotonicity assumption and theorem 3, the result follows. □

In what follows, given a non zero series Y in x, we denote by $v(Y)$ its valuation. Given a vector V of elements in a $\mathbb{K}$-vector space, we will also denote by $\mathrm{Vect}(V)$ the subvector space generated by the entries of V, and

$$v^{\max}(V) \;=\; \max\{v(Y): Y \in \mathrm{Vect}(V) \setminus \{0\}\}.$$

THEOREM 14. *Let $H = (H_0, ..., H_{r-1}) \in \mathbb{K}[[x]]^r$ be $\mathbb{K}$-linearly independent. Then there exists a unique monic operator $L = \mathrm{ann}(H) \in \delta^r + \mathbb{K}[[x]][\delta]_r$ with $L(H) = 0$. Moreover, given the truncation of H at order $O(x^n)$, we may compute L at order $O(x^{n-v^{\max}(H)})$ in time $O(\mathsf{SM}(n,r)\log r)$. In other words,*

$$\mathsf{SA}(n,r) \;=\; O(\mathsf{SM}(n,r)\log r). \tag{6}$$

Proof. Modulo a triangularization of H, we may assume without loss of generality that $v(H_0) < \cdots < v(H_{r-1}) = v^{\max}(H)$. We define operators $L^{[0]}, ..., L^{[r]}$ by

$$\begin{aligned} L^{[0]} &= 1 \\ L^{[i+1]} &= \left(\delta - \frac{\delta L^{[i]}(H_i)}{L^{[i]}(H_i)}\right) L^{[i]}. \end{aligned}$$

Then $L = L^{[r]}$ annihilates H and for any other operator $\tilde{L} \in \delta^r + \mathbb{K}[x,\delta]_{n,r}$ with $\tilde{L}(H) = 0$, we would have $(\tilde{L} - L)(H) = 0$, which is in contradiction with the fact that $\dim \ker (\tilde{L} - L) < r$. Moreover, by induction over i, we observe that the coefficient of x^0 in $L^{[i]}$ is given by $(\delta - v(H_0)) \cdots (\delta - v(H_{i-1}))$ and the coefficients of $x^0, ..., x^{n-1}$ in $L^{[i]}$ can be expressed in terms of the coefficients of $x^0, ..., x^{n-1+v(H_{i-1})}$ in $H_0, ..., H_{i-1}$. In particular, the truncation of L at order $O(x^{n-v^{\max}(H)})$ is uniquely determined by the truncation of H at order $O(x^n)$.

In order to explicitly compute L up to a given order, it is more efficient to use a divide and conquer approach. More precisely, given $H \in (H_0, ..., H_{r-1}) \in \mathbb{K}[x]_n^r$ we compute $\mathrm{ann}_n(H) \in \delta^r + \mathbb{K}[x,\delta]_{n,r}$ using the following method:

- If $r = 1$, then we take $\mathrm{ann}_n(H) = \delta - (\delta H_0 / H_0) \bmod x^n$.
- Otherwise, let $r = a + b$ with $a = \lceil r/2 \rceil$.
- Compute $A := \mathrm{ann}_n(H_0, ..., H_{a-1})$.
- Evaluate $I := (A(H_a), ..., A(H_{r-1})) \bmod x^n$.
- Compute $B := \mathrm{ann}_n(I_0, ..., I_{b-1})$.
- Return $L = B\,A \bmod x^n$.

If $n > v^{\max}(H)$, then it is easy to check that $\mathrm{ann}_n(H)(H) = O(x^{n-v^{\max}(H)})$. For a fixed constant C, we thus have

$$\mathsf{SA}(n, 2r) \;\leqslant\; 2\,\mathsf{SA}(n,r) + C\,\mathsf{SM}(n,r).$$

The result now follows from the monotonicity assumption. □

Remark 15. If $\mathsf{SM}(n,r)/r^{1+\epsilon}$ is increasing in r for some $\epsilon > 0$, then the bound further simplifies to $\mathsf{SA}(n,r) = O(\mathsf{SM}(n,r))$.

Remark 16. We notice that the operator L in theorem 14 is singular if and only if $v^{\max}(H) = r - 1$, and if and only if $\{v(Y): Y \in \mathrm{Vect}(H) \setminus \{0\}\} = \{0, ..., r-1\}$.

Although a general operator $L \in \mathbb{K}[x,\delta]$ can be singular at the origin, many operations on operators (such as division and greatest common divisors) commute with translations $x \mapsto x + x_0$, and the following lemmas can be used in order to reduce to the case when L is non singular at the origin.

LEMMA 17. *Any operator $L \in \mathbb{K}[x,\delta]_{n,r}$ can be rewritten as an operator in $\mathbb{K}[x,\partial]_{n+r,r}$ in time $O(\mathsf{SM}(n,r))$. Similarly, an operator $L \in x^r\,\mathbb{K}[x,\partial]$ may be rewritten as an operator in $\mathbb{K}[x,\delta]_{n+r,r}$ in time $O(\mathsf{SM}(n,r))$.*

Proof. In [15, 5], it is shown how to perform these rewritings using matrix products, so that the result follows from theorem 8. □

LEMMA 18. *Given a non zero operator $L \in \mathbb{K}[x,\partial]_{n,r}$, we may find a point $x_0 \in \mathbb{K}$ where L is non singular in time $O(\mathsf{M}(n))$.*

Proof. Let L_k be the leading coefficient of L. Since $\deg_x L_k < n$, we have $L_k(x_0) \neq 0$ for some $x_0 \in \{0, ..., n\}$. Using fast multipoint evaluation [4], we may find such a point x_0 in time $O(\mathsf{M}(n))$. □

5. DIVISION

From the formula (3) it is clear that both the degrees in x and δ are additive for the multiplication of operators $K, L \in \mathbb{K}[x,\delta]$. In particular, if $K, L \in \mathbb{K}[x,\delta]_{n,r}$ and L is left or right divisible by K, then the quotient is again in $\mathbb{K}[x,\delta]_{n,r}$.

THEOREM 19. *Let $K, L \in \mathbb{K}[x,\delta]_{n,r}$ be such that $L = Q\,K$ for some $Q \in \mathbb{K}[x,\delta]$. Then we may compute Q in time $O(\mathsf{SM}(n,r)\log n)$.*

Proof. By lemmas 17 and 18, and modulo a shift $x \mapsto x + x_0$, we may assume without loss of generality that K and L are non singular at the origin. We now use the following algorithm:

- We first compute the canonical fundamental system of solutions H to $L(H) = 0$ up to order $O(x^{n+r})$. By theorem 13, this can be done in time $O(\mathsf{SM}(n,r)\log n)$.
- We next evaluate $I = K(H)$ and compute a $\mathbb{K}$-basis G for $\mathrm{Vect}(I)$ at order $O(x^{n+r})$. This can be done in time $O(\mathsf{SM}(n,r))$, by theorems 3 and 8, and using linear algebra. Since K is non singular, we have $v(Y) \geqslant \deg_\delta K \Rightarrow v(K(Y)) = v(Y)$ for all $Y \in \mathbb{K}[[x]]$. In particular, the $\deg_\delta Q = \deg_\delta L - \deg_\delta K$ elements of H of valuations $\deg_\delta K, ...,$ $\deg_\delta L - 1$ are mapped to set which spans a vector space of dimension $\deg_\delta Q$. This shows that $s = \dim\,(\mathrm{Vect}(I) \bmod x^r) = \deg_\delta Q$.
- We now compute the monic annihilator $\Omega = \mathrm{ann}(G)$ of G at order $O(x^n)$. This can be done in time $O(\mathsf{SM}(n,r)\log r) = O(\mathsf{SM}(n,r)\log n)$, by theorem 14.
- We return the truncation of $Q_s\,\Omega$ at order $O(x^n)$, where $Q_s = L_{\deg_\delta L}/K_{\deg_\delta K}$.

Since each of the steps can be carried out in time $O(\mathsf{SM}(n,r)\log n)$, the result follows. □

It is classical that euclidean division generalizes to the skew polynomial ring $\mathbb{K}(x)[\delta]$. In other words, given operators $A, B \in \mathbb{K}(x)[\delta]$ where $B \neq 0$, there exist unique operators $Q = \mathrm{quo}(A,B)$ and $R = \mathrm{rem}(A,B)$ in $\mathbb{K}(x)[\delta]$ with

$$A \;=\; Q\,B + R,$$

and $\deg_\delta R < \deg_\delta B$. If $A, B \in \mathbb{K}[x, \delta]$ and I is the leading term of B with respect to δ, then left multiplication of A by $I^{\deg_\delta A - \deg_\delta B + 1}$ allows us to remain in the domain $\mathbb{K}[x, \delta]$: there exist unique $Q = \mathrm{pquo}(A, B)$ and $R = \mathrm{prem}(A, B)$ in $\mathbb{K}[x, \delta]$ with

$$I^{\deg_\delta A - \deg_\delta B + 1} A \;=\; Q\, B + R, \tag{7}$$

and $\deg_\delta R < \deg_\delta B$. The operators Q and R are usually called pseudo-quotients and pseudo-remainders. In some cases, a non trivial polynomial can be factored out in the relation (7). Let J be monic, of maximal degree, such that $J^{-1} Q\, B, J^{-1} R \in \mathbb{K}[x, \delta]$. Then we call $J^{-1} Q = \mathrm{quo}^*(A, B)$ and $J^{-1} R = \mathrm{rem}^*(A, B)$ the "simplified" pseudo-quotient and pseudo-remainder of A and B.

LEMMA 20. *Let $H = (H_0, \ldots, H_{r-1}) \in \mathbb{K}[[x]]^r$ be $\mathbb{K}$-linearly independent and define $p = v^{\max}(\mathrm{Vect}(H)) + 1$. Given $G \in (x^p\, \mathbb{K}[[x]])^r$, there exists a unique operator $L \in \mathbb{K}[[x]][\delta]_r$ of order $< r$ with $L(H) = G$ and we may compute its first n terms with respect to x in time $O(\mathsf{SM}(n + p, r) \log n)$.*

Proof. Let $\alpha_i = v(H_i)$ for each i. Modulo a base change, we may assume without loss of generality that $\alpha_0 < \cdots < \alpha_{r-1}$. Let $\Phi : \mathbb{K}[[x]]^r \to \mathbb{K}[[x]]^r$ be the operator with

$$\Phi(V_0, \ldots, V_{r-1}) \;=\; (x^{\alpha_0} V_0, \ldots, x^{\alpha_{r-1}} V_{r-1}),$$

and let Φ^{-1} denote the inverse operator. Let $\Psi : \mathbb{K}[[x]][\delta]_r \to \mathbb{K}[[x]]^r$ be the operator with

$$\Psi(K) \;=\; \Phi^{-1}(K(\Phi(1))).$$

Writing $K = \sum_{i,k} K_{i,k}\, x^k\, \delta^i$ and $\Psi(K)_{i,k} = (\Psi(K)_i)_k$, we have

$$\begin{pmatrix} \Psi(K)_{0,k} \\ \vdots \\ \Psi(K)_{r-1,k} \end{pmatrix} = \begin{pmatrix} 1 & k+\alpha_0 & \cdots & (k+\alpha_0)^{r-1} \\ \vdots & \vdots & & \vdots \\ 1 & k+\alpha_{r-1} & \cdots & (k+\alpha_{r-1})^{r-1} \end{pmatrix} \begin{pmatrix} K_{0,k} \\ \vdots \\ K_{r-1,k} \end{pmatrix}.$$

In other words, Ψ and its inverse Ψ^{-1} operate coefficientwise and n coefficients can be computed in time $O(r^\omega\, n)$.

Putting $H_i = x^{\alpha_i} + E_i$ with $E_i = o(x^{\alpha_i})$ for each i, we may rewrite the equation $L(H) = G$ as

$$L \;=\; \Psi^{-1}(\Phi^{-1}(G - L(E)))$$

and we observe that the coefficient of x^k in the righthand side of (8) only depends on earlier coefficients of $1, \ldots, x^{k-1}$ in L. In particular, we may solve the equation using a relaxed algorithm. Then the main cost is concentrated in the relaxed evaluation of $L(E)$. As in the proof of theorem 13, this evaluation can be done in time $O(\mathsf{SM}(n + p, r) \log n)$. □

THEOREM 21. *Let $K, L \in \mathbb{K}[x, \delta]_{n,r}$ with $n \geqslant r$ and $s = \deg_\delta K > 0$. Skew pseudo-division of L by K and simplification yields a relation*

$$A\, L \;=\; Q\, K + R,$$

where A, $Q = \mathrm{quo}^*(L, K)$, $R = \mathrm{rem}^*(L, K) \in \mathbb{K}[x, \delta]$. If $n' \geqslant n$ is such that $A, Q, R \in \mathbb{K}[x, \delta]_{n',r}$, then A, Q and R can be computed in time $O(\mathsf{SM}(n', r) \log n')$.

Proof. Modulo a shift $x \mapsto x + x_0$, we may assume without loss of generality that K and L are non singular at the origin. We now use the following algorithm:

- We compute the canonical fundamental system H of solutions to $K(H) = 0$ up to order $O(x^{2n'+r})$. This requires a time $O(\mathsf{SM}(n', s) \log n')$.

- We compute $G = L(H)$ with $R(H) = A\,G$ up to order $O(x^{2n'+r})$. This requires a time $O(\mathsf{SM}(n', r))$.
- We determine the operator $\Omega \in \mathbb{K}[[x]][\delta]_s$ with $\Omega(H) = x^s\,G$ up to order $O(x^{2n'+r})$. The lemma shows that this can be done in time $O(\mathsf{M}(n', s)\log n')$.
- By theorem 14, we have $R = x^{-s}\,A\,\Omega$ and $x^{-s}\,\Omega$ is known up to order $O(x^{2n'})$. Now $x^{-s}\,\Omega_0$, ..., $x^{-s}\,\Omega_{s-1}$ are truncated rational functions, for which the degrees of the numerators and denominators are bounded by n'. Using rational function reconstruction [9], we may thus compute $N_k/D_k = x^{-s}\,\Omega_k$ with $\gcd(N_k, D_k) = 1$ in time $s\,O(\mathsf{M}(n)\log n)$. Taking $A = \operatorname{lcm}(D_0, ..., D_{s-1})$, we find R.
- Once A and R are known, we compute Q using the algorithm from theorem 19.

The total complexity of this algorithm is bounded by $O(\mathsf{SM}(n', r)\log n')$. □

Remark 22. In the above proof, we have assumed that n' is known beforehand. In general, we may still apply the above algorithm for a trial value n^*. Then the algorithm may either fail (for instance, if $\deg \operatorname{lcm}(D_0, ..., D_{s-1}) \geqslant n^*$), or return the triple (A, Q, R) under the assumption that $A, Q, R \in \mathbb{K}[x, \delta]_{n^*,r}$. We may then check whether the triple is correct in time $O(\mathsf{SM}(n^*, r))$. Applying this procedure for successive guesses $n^* = n, 2\,n, 4\,n, ...$, the algorithm ultimately succeeds for an n^* with $n^* \leqslant 2\,n'$. Using the monotonicity hypothesis, the total running time thus remains bounded by $O(\mathsf{SM}(n^*, r)\log n^*) = O(\mathsf{SM}(n', r)\log n')$.

6. EUCLIDEAN OPERATIONS

It is classical that greatest common divisors and least common multiples exist in the skew euclidean domain $\mathbb{K}(x)[\delta]$: given two operators $K, L \in \mathbb{K}(x)[\delta]$, the greatest common divisor $\Gamma = \gcd(K, L)$ and the least common multiple $\Lambda = \operatorname{lcm}(K, L)$ are the unique monic operators with

$$\begin{aligned} \mathbb{K}(x)[\delta]\,\Gamma &= \mathbb{K}(x)[\delta]\,K + \mathbb{K}(x)[\delta]\,L \\ \mathbb{K}(x)[\delta]\,\Lambda &= \mathbb{K}(x)[\delta]\,K \cap \mathbb{K}(x)[\delta]\,L. \end{aligned}$$

Assume now that K, $L \in \mathbb{K}[x, \delta]$ and let A and B be monic polynomials of minimal degrees, such that $A\,\Gamma$ and $B\,\Lambda$ are in $\mathbb{K}[x, \delta]$. Then we call $\Gamma^* = \operatorname{gcd}^*(K, L) = A\,\Gamma$ and $\Lambda^* = \operatorname{lcm}^*(K, L) = B\,\Lambda$ the (simplified) pseudo-gcd and pseudo-lcm of K and L.

THEOREM 23. *Let $K, L \in \mathbb{K}[x, \delta]_{n,r}$ and $n' \geqslant n$ be such that $\Gamma^* = \operatorname{gcd}^*(K, L) \in \mathbb{K}[x, \delta]_{n',r}$. Assume that K, L and $\operatorname{lcm}^*(K, L)$ are non singular at the origin. Then we may compute Γ^* in time $O(\mathsf{SM}(n', r)\log n')$.*

Proof. We compute Γ^* using the following algorithm:

- We compute the canonical fundamental systems of solutions G and H to $K(G) = 0$ and $L(H) = 0$ at order $O(x^{2n'+2r})$. This can be done in time $O(\mathsf{SM}(n', r)\log n')$.
- Using linear algebra, we compute a basis B for $V = \operatorname{Vect}(G) \cap \operatorname{Vect}(H)$ at order $O(x^{2n'+2r})$. This can be done in time $O(n'\,r^{\omega-1})$. Since $\Lambda^* = \operatorname{lcm}^*(K, L)$ is non singular, we have $\dim\,([\operatorname{Vect}(G) + \operatorname{Vect}(H)] \bmod x^{2r}) = \deg_\delta \Lambda^* = \deg_\delta G + \deg_\delta H - \deg_\delta \Gamma^*$. Hence, $s = \dim\,(V \bmod x^{2r}) = \dim\,(\operatorname{Vect}(G) \bmod x^{2r}) + \dim\,(\operatorname{Vect}(H) \bmod x^{2r}) - \dim\,([\operatorname{Vect}(G) + \operatorname{Vect}(H)] \bmod x^{2r}) = \deg_\delta \Gamma^*$.
- We compute $\Omega = \operatorname{ann}(B) = \gcd(K, L)$ at order $O(x^{2n'})$. By theorem 14, this can be done in time $O(\mathsf{SM}(n', r)\log n')$.

- We compute Γ^* from $\Omega \bmod x^{2n'}$ using rational function reconstruction.

This algorithm requires a running time $O(\mathsf{SM}(n', r) \log n')$. □

Remark 24. In the above proof, we have again assumed that n' is known beforehand. Below, we will discuss ways to check the correctness of the computed result for a trial value n^*, after which a similar strategy as in remark 22 can be applied. During the relaxed computation of G and H, we may also check whether $V = \varnothing$ at each next coefficient. In the particular case when $\Gamma = 1$, the running time of the algorithm will then be bounded by $O(\mathsf{SM}(n^*, r) \log n^*)$, where n^* is the smallest order at which common solutions no longer exist. This kind of early termination only works for this very special case.

Remark 25. Notice that Γ^* might be singular at the origin, even if K, L and $\operatorname{lcm}^*(K, L)$ are not. This happens for instance when K is the minimal annihilator of the vector $(1, x)$ and L the minimal annihilator of the vector (e^x, x), so that $\Gamma = \delta - 1$.

THEOREM 26. *Let $K, L \in \mathbb{K}[x, \delta]_{n,r}$ and $n' \geqslant n$ be such that $\Lambda^* = \operatorname{lcm}^*(K, L) \in \mathbb{K}[x, \delta]_{n',r}$. If K, L and $\operatorname{lcm}^*(K, L)$ are non singular at the origin, then we may compute Λ^* in time $O(\mathsf{SM}(n', r) \log n')$.*

Proof. Similar to the proof of theorem 23, by taking $V = \operatorname{Vect}(K) + \operatorname{Vect}(L)$ instead of $V = \operatorname{Vect}(K) \cap \operatorname{Vect}(L)$. □

Remark 27. The above algorithms can be generalized to gcds and lcms of more than two operands. This is usually more efficient than the repeated computation of gcds or lcms of pairs.

The assumption that $\operatorname{lcm}^*(K, L)$ should be non singular is still a bit unsatisfactory in theorems 23 and 26, even though the probability that a randomly chosen point is singular is infinitesimal. If we drop this assumption, then we still have $s \geqslant \deg_\delta \Gamma^*$ in the proof of theorem 23. Consequently, "candidate" pseudo-gcds Γ^* found by the algorithm are genuine pseudo-gcds whenever Γ^* pseudo-divides both K and L. Using the division algorithms from the previous section, this can be checked in time $O(\mathsf{SM}(n' r, r) \log n')$ in the case of gcds and $O(\mathsf{SM}(n r, r) \log n')$ in the case of lcms.

An alternative way to check whether candidate gcds and lcms are correct is to compute Bezout and Ore relations. More precisely, given $K, L \in \mathbb{K}(x)[\delta]$ with $L \notin \mathbb{Q}(x)\, K$, there exist operators $A, B, C, D \in \mathbb{K}(x)[\delta]$ with

$$\begin{pmatrix} \Gamma \\ 0 \end{pmatrix} = \begin{pmatrix} A & B \\ C & D \end{pmatrix} \begin{pmatrix} K \\ L \end{pmatrix},$$

$\deg_\delta A\, K, \deg_\delta B\, L < \deg_\delta \Lambda$ and $C\, K = -D\, L = \Lambda$. The 2×2 matrix at the righthand side will be called the Euclidean matrix $\mathrm{E} = \operatorname{Eucl}(K, L)$ of K and L. In a similar way as above, we may define a (simplified) pseudo-Euclidean matrix $\mathrm{E}^* = \operatorname{Eucl}^*(K, L)$ with entries A^*, B^*, C^*, D^* in $\mathbb{K}[x, \delta]$, whenever $K, L \in \mathbb{K}[x, \delta]$. We will say that $\operatorname{Eucl}(K, L)$ is non singular at x_0, if the denominators of A, B, C and D do not vanish at x_0.

THEOREM 28. *Let $K, L \in \mathbb{K}[x, \delta]_{n,r}$ and $n' \geqslant n$ be such that $\mathrm{E}^* = \operatorname{Eucl}^*(K, L) \in \mathbb{K}[x, \delta]^{2\times 2}_{n',r}$. If K, L, $\operatorname{lcm}^*(K, L)$ and $\operatorname{Eucl}(K, L)$ are non singular at the origin, then we may compute Λ^* in time $O(\mathsf{SM}(n', r) \log n')$.*

Proof. Assuming n' known, we compute $\operatorname{Eucl}(K, L) = \left(\begin{smallmatrix} A & B \\ C & D \end{smallmatrix}\right)$ at order $O(x^{2n'})$ as follows:

- We compute the canonical fundamental systems of solutions G and H to $K(G) = 0$ and $L(H) = 0$ at order $O(x^{2n'+3r})$.

- We compute a basis X for $\mathrm{Vect}(G) \cap \mathrm{Vect}(H)$ at order $O(x^{2n'+3r})$, together with bases $\hat{G}$ and $\hat{H}$ for the supplements of $\mathrm{Vect}(X)$ in $\mathrm{Vect}(G)$ resp. $\mathrm{Vect}(H)$. We also compute $\Gamma = \mathrm{ann}(X)$ at order $O(x^{2n'+2r})$.
- We solve the systems $A\big(K\big(\hat{H}\big)\big) = \Gamma\big(\hat{H}\big)$ and $B(L(\hat{G})) = \Gamma(\hat{G})$ in A resp. B at order $O(x^{2n'})$, using lemma 20.
- We compute a basis Y for $\mathrm{Vect}(G) + \mathrm{Vect}(H)$ at order $O(x^{2n'+2r})$, as well as bases $\tilde{H}$ and $\tilde{G}$ for the vector spaces $\mathrm{Vect}(K(Y))$ resp. $\mathrm{Vect}(L(Y))$ at order $O(x^{2n'+2r})$.
- We compute $C = K^{-1}_{\deg_\delta K} \mathrm{ann}(\tilde{H})$ and $D = -L_{\deg_\delta L} \mathrm{ann}(\tilde{G})$ at order $O(x^{2n'})$.

We finally compute E^* from A, B, C and D using rational function reconstruction. The complexity analysis and the remainder of the proof is done in a similar way as in the proofs of theorems 21 and 23. □

With the above techniques, we may at least verify whether computed pseudo-gcds or pseudo-lcms are correct. For a fully deterministic algorithm, we still need a way to find a point where $\mathrm{lcm}^*(K, L)$ is non singular. This can be done by brute force. Let us state the result for pseudo-gcds; similar deterministic results hold for pseudo-lcms and pseudo-Euclidean matrices.

THEOREM 29. *Let $K, L \in \mathbb{K}[x,\delta]_{n,r}$ and $n' \geqslant n$ be such that $\Gamma^* = \mathrm{gcd}^*(K, L) \in \mathbb{K}[x,\delta]_{n',r}$. Then we may compute Γ^* in time $O(\mathsf{SM}(n'r, r) \log n' + n'(\mathsf{M}(n)r + r^\omega))$.*

Proof. Let $k = \deg_\delta K$, $l = \deg_\delta L$, and assume first that we know n'. Then, at $n'+1$ distinct random points where K and L are non singular, we compute canonical fundamental systems of solutions G and H at order $O(x^{k+l})$. This can be done in time $O(n'(\mathsf{M}(n)r + r^\omega))$. We now pick a point at which the dimension of $(\mathrm{Vect}(G) + \mathrm{Vect}(H)) \bmod x^{k+l}$ is maximal and apply the algorithm from theorem 23 in order to find Γ^*. If n' is unknown, then we use a sequence of guesses $n' = n, 2n, 4n, \ldots$, as in the previous proofs. □

BIBLIOGRAPHY

[1] A.V. Aho, K. Steiglitz and J.D. Ullman. Evaluating polynomials on a fixed set of points. *SIAM Journ. of Comp.*, 4:533–539, 1975.

[2] A. Borodin and R.T. Moenck. Fast modular transforms. *Journal of Computer and System Sciences*, 8:366–386, 1974.

[3] A. Bostan. *Algorithmique efficace pour des opérations de base en calcul formel*. PhD thesis, École polytechnique, 2003.

[4] A. Bostan and É. Schost. Polynomial evaluation and interpolation on special sets of points. *Journal of Complexity*, 21(4):420–446, August 2005. Festschrift for the 70th Birthday of Arnold Schönhage.

[5] Alin Bostan, Frédéric Chyzak and Nicolas Le Roux. Products of ordinary differential operators by evaluation and interpolation. In J. Rafael Sendra and Laureano González-Vega, editors, *ISSAC*, pages 23–30. Linz/Hagenberg, Austria, July 2008. ACM.

[6] D.G. Cantor and E. Kaltofen. On fast multiplication of polynomials over arbitrary algebras. *Acta Informatica*, 28:693–701, 1991.

[7] J.W. Cooley and J.W. Tukey. An algorithm for the machine calculation of complex Fourier series. *Math. Computat.*, 19:297–301, 1965.

[8] D. Coppersmith and S. Winograd. Matrix multiplication via arithmetic progressions. In *Proc. of the 19^{th} Annual Symposium on Theory of Computing*, pages 1–6. New York City, may 25–27 1987.

[9] J. von zur Gathen and J. Gerhard. *Modern Computer Algebra*. Cambridge University Press, 2-nd edition, 2002.

[10] R.T. Moenck and A. Borodin. Fast modular transforms via division. In *Thirteenth annual IEEE symposium on switching and automata theory*, pages 90–96. Univ. Maryland, College Park, Md., 1972.

[11] V. Pan. *How to multiply matrices faster*, volume 179 of *Lect. Notes in Math.* Springer, 1984.

[12] V. Pan and D. Bini. *Polynomial and matrix computations*. Birkhauser, 1994.

[13] V. Strassen. Gaussian elimination is not optimal. *Numer. Math.*, 13:352–356, 1969.

[14] V. Strassen. Die Berechnungskomplexität von elementarsymmetrischen Funktionen und von Interpolationskoeffizienten. *Numer. Math.*, 20:238–251, 1973.

[15] J. van der Hoeven. FFT-like multiplication of linear differential operators. *JSC*, 33(1):123–127, 2002.

[16] J. van der Hoeven. Relax, but don't be too lazy. *JSC*, 34:479–542, 2002.

[17] J. van der Hoeven. Relaxed multiplication using the middle product. In Manuel Bronstein, editor, *Proc. ISSAC '03*, pages 143–147. Philadelphia, USA, August 2003.

[18] J. van der Hoeven, G. Lecerf, B. Mourain et al. Mathemagix. 2002. `http://www.mathemagix.org`.