Non-commutative Elimination in Ore Algebras Proves Multivariate Identities

Frédéric Chyzak, Bruno Salvy

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET AUTOMATIQUE

Non-commutative Elimination in Ore Algebras Proves Multivariate Identities

Frédéric CHYZAK, Bruno SALVY

N° 2799

Février 1996

PROGRAMME 2

Rapport de recherche

1996

ISSN 0249-6399

Non-commutative Elimination in Ore Algebras Proves Multivariate Identities

Frédéric Chyzak and Bruno Salvy

Abstract
Many identities involving special functions can be proved using the theory of ∂-finite or holonomic sequences and functions. This theory applies in particular to numerous combinatorial identities. This work presents a theoretical and algorithmic approach to the multivariate case, together with an implementation.

Preuve d'identités multivariées par élimination non-commutative dans des algèbres de Ore

Résumé
De nombreuses identités faisant intervenir des fonctions spéciales peuvent être prouvées par la théorie des suites et fonctions ∂-finies ou holonomes. Cette théorie s'applique en particulier à nombre d'identités combinatoires. Ce travail présente une approche théorique et algorithmique au cas multivarié, ainsi qu'une implantation.

NON-COMMUTATIVE ELIMINATION IN ORE ALGEBRAS PROVES MULTIVARIATE IDENTITIES

FRÉDÉRIC CHYZAK AND BRUNO SALVY

Abstract. Many identities involving special functions can be proved using the theory of ∂-finite or holonomic sequences and functions. This theory applies in particular to numerous combinatorial identities. This work presents a theoretical and algorithmic approach to the multivariate case, together with an implementation.

Introduction

Computer algebra consists in performing calculations on mathematical objects represented by a finite amount of information. A class of computer algebra objects is especially useful when it is possible to recognize whether two members of the class are identical or not. D. Zeilberger showed that a large set of combinatorial identities can be proved using properties of the class of P-finite functions and sequences and the important subclass of *holonomic* functions [33]. A function is P-finite when the set of its partial derivatives spans a finite-dimensional vector space over the rational functions. Computationally, a P-finite function is defined by a set of linear differential equations (linear relations between the partial derivatives) and a finite number of initial conditions. Proving that a P-finite function is zero requires finding a linear system it satisfies and checking that sufficiently many of its initial conditions are zero. This computation is made possible by the numerous closure properties enjoyed by the class of P-finite functions. Similarly, P-finite sequences are defined as sequences such that the set of sequences obtained by shifting the indices spans a finite-dimensional vector space over the rational functions. Identities involving such sequences are proved by computing systems of recurrences and sufficiently many initial conditions. There again, the computation of these systems is made possible by the closure properties enjoyed by the class of P-finite sequences. It is well-known that both properties are equivalent in the univariate case via generating functions. In the multivariate case, however, the equivalence does not hold in general and this has motivated L. Lipshitz to give a technical definition of P-finiteness of sequences in several variables [17].

The experience gained from an implementation of the univariate case [23] shows that the algorithms used in the differential and in the difference case are essentially identical. It is therefore natural to encompass both notions into a more general one. Our aim is to make effective operations on systems of *linear operators* constrained so that their solutions lie in a finite dimensional vector space. First approaches to the mixed differential-difference case are due to D. Zeilberger [33] and N. Takayama [26]. We use Ore polynomials and skew polynomial rings to also deal with q-equations, and numerous other linear equations in pseudo-derivatives. The solutions of these systems will be called *∂-finite*. However, none of our algorithms deals with these ∂-finite solutions. Instead, they deal with Ore polynomials which can be interpreted as operators annihilating them. Initial conditions therefore lie outside of the scope of our algorithms. Indeed, for each type of Ore polynomials, initial conditions require specific algorithms and a specific implementation. In Section 1, Ore polynomials are introduced and the algorithmic tools to work with them are provided. As N. Takayama noticed in the differential-difference case, and as was developed by A. Kandri-Rody and V. Weispfenning in the more general setting of polynomial rings of solvable type [13], Buchberger's algorithm for *Gröbner bases* can be adapted to this non-commutative context. These bases furnish normal forms and an algorithm for elimination. In Section 2, we use Gröbner bases to make some of the closure properties effective. When interpreted in terms of ∂-finite functions, these closure properties correspond to closure under addition, product and pseudo-derivative.

In Section 3, we generalize to the context of Ore polynomials the important operation of *creative telescoping*, which makes it possible to compute definite sums and integrals. In the special case of the Weyl algebra (differential equations), this algorithm is guaranteed to succeed for a subclass of systems of equations classically called *holonomic*. Moreover, if a function is ∂-finite, there exists a holonomic system annihilating it [14]. Results obtained by holonomy can also be translated to results for sequences via generating functions [33]. In the general case of Ore polynomials, we do not have a corresponding notion of holonomy. The algorithms we give for creative telescoping are therefore not guaranteed to succeed. We give two such algorithms. The first one is slow but will always terminate (successfully or detecting that the algorithm has failed), the second one is faster but may fail to terminate.

All these operations are illustrated by examples using F. Chyzak's implementation [6][1]. In conclusion, we recall the special case of Weyl algebras, where more operations are possible and we discuss envisioned extensions.

1. Non-commutative algebras of operators

1.1. **Definitions.** O. Ore [20] initiated an algebraic treatment of a very general class of linear operators now called *Ore operators*. We give a slightly restricted definition that fits our needs (see [7] for the general case). Since all algebras of interest to our study are skew algebras of operators, we adopt the convention that the words *rings* and *fields* always refer to possibly skew rings and fields. We specify *commutative ring* or *commutative field* when necessary. Moreover, all rings under consideration in this paper are of characteristic 0.

Table 1 gives examples of the type of operators we consider. All these operators share a very simple commutation rule of the variable ∂ with polynomials in x. This rule is the basis of the definition of skew polynomial rings.

Definition 1. Let $\mathbb{A}$ be an integral domain, i.e., ring with no zero-divisors. The *skew polynomial ring* $\mathbb{A}[\partial;\sigma,\delta]$ is the ring of polynomials in ∂ with coefficients in $\mathbb{A}$, with usual addition and a product defined by associativity from the following commutation rule

$$\forall a \in \mathbb{A} \qquad \partial a = \sigma(a)\partial + \delta(a). \tag{1}$$

Here, σ is an injective ring endomorphism of $\mathbb{A}$ and δ a σ-derivation, i.e., an additive endomorphism of $\mathbb{A}$ which satisfies the following Leibniz rule:

$$\forall a, b \in \mathbb{A} \qquad \delta(ab) = \sigma(a)\delta(b) + \delta(a)b. \tag{2}$$

Additionally, δ is assumed to commute with σ.

Degree in ∂ and coefficients are defined as in the commutative case, the coefficients being on the left side of the monomials.

[1]The packages mentioned in this article are available
by anonymous ftp from `ftp.inria.fr:INRIA/Projects/algo/programs`
or at the URL `http://www-rocq.inria.fr/algo/libraries/libraries.html`.

Examples of skew polynomial rings are given in Table 1. In all the cases under consideration in this table, $\mathbb{A}$ is of either form $\mathbb{K}[x]$ or $\mathbb{K}(q)[x]$ with $\mathbb{K}$ a field. By associativity, relation (1) then induces

$$\forall p \geq 1 \qquad \delta(x^p) = \delta(x) \sum_{k=0}^{p-1} \sigma(x)^k x^{p-1-k}. \tag{3}$$

Thus σ and δ are completely determined by their values on x. One reason for studying these skew polynomial rings is that operations which can be performed in them need only be implemented once and then apply equally to linear differential equations, linear difference equations or their q-analogues.

The following proposition is due to the existence of a degree function and leads to the multivariate case.

Proposition 1. [7, p. 35] *The skew polynomial ring $\mathbb{A}[\partial; \sigma, \delta]$ is an integral domain.*

By choosing appropriate integral domains $\mathbb{A}$, we can use this proposition in conjunction with Definition 1 to construct various multivariate skew polynomial rings. Several of these choices will be useful in the sequel. In particular, we have the following important special cases.

Definition 2. Let $\mathbb{K}$ be a field, $\mathbf{x} = (x_1, \ldots, x_p)$ and $\mathbf{y} = (y_1, \ldots, y_q)$ tuples of commutative variables. Let also $\boldsymbol{\partial} = (\partial_1, \ldots, \partial_r)$ be an r-tuple of variables that commute pairwise, and such that the commutation of the ∂_i's with the elements of $\mathbb{A} = \mathbb{K}(x_1, \ldots, x_p)[y_1, \ldots, y_q]$ is ruled by relations of the type (1) for corresponding σ_i's and δ_i's. The skew polynomial ring $\mathbb{A}[\partial_1; \sigma_1, \delta_1] \cdots [\partial_r; \sigma_r, \delta_r]$ is called an *Ore algebra*. It is denoted $\mathbb{K}(\mathbf{x})[\mathbf{y}][\boldsymbol{\partial}; \boldsymbol{\sigma}, \boldsymbol{\delta}]$. The special case when $p = 0$ is called a *polynomial Ore algebra*, and the special case when $q = 0$ is called a *rational Ore algebra*.

Example. Weyl algebras $\mathbb{K}[x_1, \ldots, x_n][\partial_{x_1}; 1, \partial_{x_1}] \cdots [\partial_{x_n}; 1, \partial_{x_n}]$ are a special case of polynomial Ore algebras, obtained when all operators are usual partial differentiations. (The variable ∂_{x_i} represents the usual differentiation operator with respect to x_i.)

Example. In $\mathbb{Q}(a, b)[n, x][S_n; S_n, 0][\partial_x; 1, \partial_x]$, where S_n denotes the shift operator with respect to n and ∂_x denotes differentiation with respect to x, the Jacobi polynomials $P_n^{(a,b)}(x)$ are cancelled by

$$(1 - x^2)\partial_x^2 + (b - a - (a + b + 2)x)\partial_x + n(n + a + b + 1), \tag{4}$$

Operator	$\sigma(a)$	$\delta(a)$	Commutation	Action of ∂
Differentiation	$a(x)$	$a'(x)$	$\partial x = x\partial + 1$	$f(x) \mapsto f'(x)$
Shift	$a(x+1)$	0	$\partial x = (x+1)\partial$	$f(x) \mapsto f(x+1)$
Difference	$a(x+1)$	$a(x+1) - a(x)$	$\partial x = (x+1)\partial + 1$	$f(x) \mapsto f(x+1) - f(x)$
q-Dilation	$a(qx)$	0	$\partial x = qx\partial$	$f(x) \mapsto f(qx)$
q-Difference	$a(qx)$	$a(qx) - a(x)$	$\partial x = qx\partial + (q-1)x$	$f(x) \mapsto f(qx) - f(x)$
q-Differentiation	$a(qx)$	$\frac{a(qx)-a(x)}{(q-1)x}$	$\partial x = qx\partial + 1$	$f(x) \mapsto \frac{f(qx)-f(x)}{(q-1)x}$
Eulerian operator	$a(x)$	$xa(x)$	$\partial x = x\partial + x$	$f(x) \mapsto xf'(x)$
e^t-Differentiation	$a(x)$	$xa(x)$	$\partial x = x\partial + x$	$f(t) \mapsto f'(t) \qquad (x = e^t)$
Mahlerian operator	$a(x^p)$	0	$\partial x = x^p\partial$	$f(x) \mapsto f(x^p) \qquad (p \geq 2)$

TABLE 1. Ore algebras

$$\begin{aligned}&2(n+2)(n+a+b+2)(2n+a+b+2)S_n^2\\(5)\qquad &-\left((2n+a+b+3)\left(a^2-b^2\right)+(2n+a+b+2)(2n+a+b+3)(2n+a+b+4)x\right)S_n\\&+2(n+a+1)(n+b+1)(2n+a+b+4).\end{aligned}$$

This is the only information our algorithms will use to deal with Jacobi polynomials. Initial conditions must be treated separately, if needed.

Example. The Ore algebra $\mathbb{Q}(q)[n,q^n][S_n;S_n,0]$ with the commutation rule

$$S_n n^k (q^n)^\ell = (n+1)^k q^\ell (q^n)^\ell S_n$$

is well-suited for certain q-computations. (Here q^n must be viewed as an indeterminate; a more formal description would be to introduce an indeterminate Q that would act on functions as the multiplication by q^n.)

1.2. **Operators, ideals and modules.** In this work, a skew polynomial ring $\mathbb{S}$ is interpreted as a ring of operators. This is achieved when ∂_i, σ_i and δ_i act as linear endomorphisms on a $\mathbb{K}$-algebra $\mathcal{F}$ of functions, power series, sequences, distributions, etc. Then Eq. (1) extends to the following Leibniz rule for products

$$(6)\qquad \forall f,g\in\mathcal{F}\qquad \partial_i(fg)=\sigma_i(f)\partial_i(g)+\delta_i(f)g.$$

This makes $\mathcal{F}$ an $\mathbb{S}$-algebra. The actions of the operators corresponding to important Ore algebras are given in Table 1. In the remainder of this article, we use the word "function" to denote any object on which the elements of an Ore algebra act.

This interpretation motivates the study of ideals of skew polynomial rings. Algebraically, an object of interest is the *left ideal* $\operatorname{Ann} f \subseteq \mathbb{S}$ of skew polynomials which vanish on some $f\in\mathcal{F}$. It is called the *annihilating ideal* of f. Most of the operations we consider below consist in finding elements of this ideal which satisfy special properties, or in finding elements of an ideal of operators annihilating a function related to f.

Correspondingly, the $\mathbb{S}$-module $\mathbb{S}\cdot f \simeq \mathbb{S}/\operatorname{Ann} f$ encapsulates much of the structure of the *pseudo-derivatives* $\partial_1^{\alpha_1}\cdots\partial_r^{\alpha_r}(f)$ of f. Computationally, all calculations take place in this module. Although all the algorithms we present below have an interpretation in terms of operators, the existence of a specific algebra $\mathcal{F}$ is not even needed. The algorithms can all be stated at the level of ideals $\mathfrak{I}$ of $\mathbb{S}$ and modules $\mathbb{S}/\mathfrak{I}$.

1.3. **Euclidean division.** Two algorithms allow us to perform most of our computations. The first one is left Euclidean division which leads to an extended right gcd algorithm. The second one is a suitably modified version of Buchberger's algorithm for Gröbner bases. The Euclidean algorithm is less powerful than the Gröbner bases approach, but it can be used in some Ore algebras where the latter is unavailable. In this section and the next one, we detail both these algorithms, their constraints and some of their applications.

The results in this section are due to O. Ore [20]. Recall our convention that fields may be skew. Call an *effective field* a field in which the usual ring operations are computable, and where given two non-zero elements α and β, one can compute two non-zero elements α' and β' such that $\alpha'\alpha+\beta'\beta=0$. In the commutative case, this can be done by taking $\alpha'=\beta$ and $\beta'=-\alpha$. Let $\mathbb{S}=\mathbb{K}[\partial;\sigma,\delta]$ be a skew polynomial ring over an effective field $\mathbb{K}$. Since the elements of $\mathbb{S}$ are polynomials in ∂, performing divisions on the left makes it possible to extend the usual Euclidean algorithm to compute right gcd's. Let a and b be two polynomials in $\mathbb{S}$ for which we want to compute a right gcd. Assume that the degree d_a of a in ∂ is greater than the degree d_b of b. Left-multiplying b by $\partial^{d_a-d_b}$ yields a second polynomial c of degree d_a. Let α and γ be the leading coefficients of a and c

respectively. Compute two non-zero cofactors α' and γ' such that $\alpha'\alpha+\gamma'\gamma=0$. Then $d=\alpha' a+\gamma' c$ has degree less than d_a in ∂. The same process is now applied to b and d. Repeating this process eventually yields zero. It is not difficult to prove that the last polynomial obtained before 0 is a right gcd g of a and b. (Gcd's are defined up to a non-zero constant in $\mathbb{K}$). Collecting the successive factors yields the extended gcd algorithm which produces u and v such that

$$ua+vb=g.$$

Left lcm's are also computed using this algorithm. This is achieved by considering the last identity produced by the algorithm:

$$Ua+Vb=0.$$

Once again, it is not difficult to prove that the polynomial Ua is a left lcm of a and b. This is summarized in the following theorem, which was proved by Ore [20] in the case of a commutative field $\mathbb{K}$, but readily extends to skew fields.

Theorem 1 (Ore). *Given two elements a and b in a skew polynomial ring $\mathbb{K}[\partial;\sigma,\delta]$ over an effective field $\mathbb{K}$, the Euclidean algorithm makes it possible to compute polynomials u,v,g,U,V, with U and V non-zero, such that*

$$ua+vb=g \qquad \text{and} \qquad Ua+Vb=0, \tag{7}$$

where g is a right gcd of a and b and Ua is a left lcm of a and b.

A *left Ore ring* is a ring such that for any non-zero elements a and b there exist non-zero U and V in the ring which satisfy $Ua=Vb$. As shown by the theorem above, skew polynomial rings over a field are left Ore rings. The proof of the above theorem also yields the following corollary.

Corollary 1 (Ore). *If $\mathbb{A}$ is a left Ore ring, so is $\mathbb{A}[\partial;\sigma,\delta]$.*

Call an *effective left Ore ring* a left Ore ring in which the usual ring operations are computable, as well as the pair (U,V) involved in Eq. (7). The previous corollary can also be interpreted as an elimination property as follows.

Corollary 2. *Given two elements a and b in a skew polynomial ring $\mathbb{S}=\mathbb{A}[\partial;\sigma,\delta]$ over an effective left Ore ring $\mathbb{A}$, if there exists $(u,v)\in\mathbb{S}^2$ and $\alpha\in\mathbb{A}\setminus\{0\}$ such that*

$$ua+vb=\alpha,$$

then (α,u,v) can be computed by the Euclidean algorithm.

Example. We apply this elimination on operators which define the Jacobi polynomials. Starting from (5) and a mixed difference-differential equation:

$$\begin{aligned}&(2n+a+b+2)\left(1-x^2\right)S_n\partial_x\\&\quad-(n+1)\left(a-b-(2n+a+b+2)x\right)S_n-2(n+a+1)(n+b+1),\end{aligned} \tag{8}$$

we prove that Jacobi polynomials also satisfy (4) by eliminating the differential operator ∂_x between (5) and (8) in the Ore algebra $\mathbb{Q}(a,b,n,x)[S_n;S_n,0][\partial_x;1,\partial_x]$. This Maple session and the following ones use F. Chyzak's *Mgfun* package. The first step is to load the package and create a suitable Ore algebra:

```
with(Mgfun):
A:=orealg(comm=[a,b],shift=[Sn,n],diff=[Dx,x]):
```

Using a philosophy reminiscent of Axiom's, an Ore algebra is represented internally as a table of procedures that perform its basic operations. Here `comm`, `diff` and `shift` are predefined types of Ore operators, but one could create Ore algebras with other operators.

We then enter both polynomials:

```
G:=[2*(n+2)*(n+a+b+2)*(2*n+a+b+2)*Sn^2
        -((2*n+a+b+3)*(a^2-b^2)+(2*n+a+b+2)*(2*n+a+b+3)*(2*n+a+b+4)*x)*Sn
        +2*(n+a+1)*(n+b+1)*(2*n+a+b+4),
    (2*n+a+b+2)*(1-x^2)*Dx*Sn-(n+1)*(a-b-(2*n+a+b+2)*x)*Sn-2*(n+a+1)*(n+b+1)]:
```

And we ask for a skew polynomial free of S_n, if possible:

```
skewelim(G[1],G[2],Sn,A);
```

$$-an - bn - n - n^2 + ax\partial_x + aD_x + bxD_x - bD_x + 2xD_x - D_x^2 + x^2D_x^2$$

This is precisely Eq. (4).

Another important application of the Euclidean algorithm is the construction of the field of fractions of a skew polynomial ring [20]. Calculations with these fractions are not needed in this work although they are used implicitly when the effective left Ore ring is of the form $\mathbb{A} = \mathbb{K}[\boldsymbol{\partial};\boldsymbol{\sigma},\boldsymbol{\delta}]$ (i.e., a skew polynomial ring in several ∂'s).

1.4. **Gröbner bases in skew polynomial rings.** In many common Ore algebras, elimination can be performed by computing non-commutative Gröbner bases using a suitable generalization of Buchberger's algorithm. Early work in this area in the context of Weyl algebras is due to A. Galligo [11]. N. Takayama used an analogous technique for difference-differential algebras [26].

A sufficient condition for Gröbner bases to be finite is that the algebra be a *left Noetherian ring*, which means that it does not contain any infinite strictly increasing sequence of left ideals. Unfortunately, not all Ore algebras are left Noetherian. An example is given in [31], with the Ore algebra $\mathbb{Q}[x][M;M,0]$, where M is the Mahlerian operator with commutation rule $Mx = x^pM$ for an integer $p > 1$ (see Table 1). Let $\mathfrak{I}_n$ be the left ideal generated by $(x, xM, \dots, xM^n)$. Then $xM^{n+1} \notin \mathfrak{I}_n$, and $(\mathfrak{I}_n)_{n\in\mathbb{N}}$ is an infinite strictly increasing sequence of left ideals. Therefore, not all left ideals have a finite basis[2].

The following theorem gives a sufficient condition for an Ore algebra to possess finite Gröbner bases, and states that these bases can be computed by a non-commutative analogue of Buchberger's algorithm. When this theorem applies, efficiency can be improved by suitable generalizations of the so-called "normal strategy" [9, chap. 2], "sugar strategy" [12] and by "trace lifting" [30]. Further discussion of implementation and efficiency will be part of F. Chyzak's thesis (see also [6]).

Theorem 2. *Let $\mathbb{O} = \mathbb{K}(\mathbf{x})[\mathbf{y}][\boldsymbol{\partial};\boldsymbol{\sigma},\boldsymbol{\delta}]$ be an Ore algebra over a field $\mathbb{K}$ such that $\boldsymbol{\partial}$, $\boldsymbol{\sigma}$ and $\boldsymbol{\delta}$ satisfy relations of the type*

$$\partial_i y_j = a_{i,j} y_j \partial_i + b_{i,j}, \qquad 1 \le i \le r, \quad 1 \le j \le q,$$

with $a_{i,j} \neq 0$ in $\mathbb{K}(\mathbf{x})$ and $b_{i,j}$ in $\mathbb{K}(\mathbf{x})[\mathbf{y}]$. Then $\mathbb{O}$ is left Noetherian and a non-commutative version of Buchberger's algorithm terminates.

As can be seen from Table 1, this theorem implies that many useful Ore algebras are left Noetherian.

[2] Surprisingly, this implies that Proposition 8.2 p. 35 in [7] is wrong.

Proof. The difficult part of this theorem is the case of polynomial Ore algebras (when $\mathbf{x}$ is empty). This case is treated by A. Kandri-Rody and V. Weispfenning [13, Theorem 4.7] under the name "polynomial rings of solvable type". The theorem can be obtained either by a simple refinement of their derivation, or by appealing to T. Mora's general theory of standard bases for filtered rings [19]. In particular, the last implication that noetherianity implies termination of Buchberger's algorithm is proved in [19].

We apply T. Mora's general theory of standard bases [19] to skew polynomial rings. First, $\mathbb{O}$ has the structure of a *filtered ring.* Let Γ be the semigroup of all (commutative) terms in $\mathbf{y}$ and $\boldsymbol{\partial}$. Let Γ be ordered by any total order $\prec$ compatible with the product and such that $1 \preceq \gamma$ for all $\gamma \in \Gamma$. Let $\{\mathcal{F}_\gamma\}_{\gamma\in\Gamma}$ be the sequence of $\mathbb{K}(\mathbf{x})$-vector spaces of all skew polynomials which involve only terms lower than or equal to γ. This is a filtration of $\mathbb{O}$, since for all $\gamma, \gamma' \in \Gamma$: *(i).* $\gamma \prec \gamma' \implies \mathcal{F}_\gamma \subseteq \mathcal{F}_{\gamma'}$; *(ii).* $\mathcal{F}_\gamma \mathcal{F}_{\gamma'} \subseteq \mathcal{F}_{\gamma\gamma'}$; and *(iii).* for all $p \in \mathbb{O}$, there exist $\gamma, \gamma' \in \Gamma$ such that $p \in \mathcal{F}_\gamma$ but $p \notin \mathcal{F}_{\gamma'}$.

For $p \in \mathbb{O} \setminus \{0\}$, let $\deg p$ be the leading term of p: $\deg p = \min\{\gamma \mid p \in \mathcal{F}_\gamma\}$. Next, let $\mathcal{V}_\gamma$ be the vector space of all skew polynomials of leading term strictly lower than γ, $\mathcal{G}_\gamma$ be the additive group $\mathcal{F}_\gamma/\mathcal{V}_\gamma$, and $\mathcal{G}$ be the direct sum $\bigoplus_{\gamma\in\Gamma} \mathcal{G}_\gamma$. The ring $\mathcal{G}$ is the *graded ring associated to* $\mathbb{O}$. Finally, let $\operatorname{Init} p$ be the residue class of p modulo $\mathcal{V}_{\deg p}$, and define Init on sets by $\operatorname{Init}\{p_i\}_{i\in I} = \{\operatorname{Init} p_i\}_{i\in I} \subseteq \mathcal{G}$. It is easily seen that $\mathcal{G} = \mathbb{K}[\mathbf{x},\mathbf{y}][\boldsymbol{\partial};\boldsymbol{\sigma},0]$. (In this ring, the product of monomials is a monomial.)

It follows from [13] that $\mathbb{K}(\mathbf{x})[\mathbf{y}][\boldsymbol{\partial};\boldsymbol{\sigma},\boldsymbol{\delta}]$ is a left Noetherian ring, so that $\mathbb{O}$ is itself a left Noetherian ring. The following two concepts therefore coincide for any left ideal $\mathfrak{I} \subseteq \mathbb{O}$ [19]:

1. B is a *standard set* for $\mathfrak{I}$ when $\operatorname{Init} B$ generates $\operatorname{Init} \mathfrak{I}$;
2. B is a *standard basis* for $\mathfrak{I}$ when any non-zero element f of $\mathfrak{I}$ can be written as a finite sum of products lb, for $l \in \mathbb{O}$ and $b \in B$ satisfying $\deg l \deg b \preceq \deg f$.

Such a set is called a *Gröbner basis* of the ideal $\mathfrak{I}$. With additional assumptions of effectiveness easily seen to be satisfied by $\mathbb{O}$, it follows from [19] that Buchberger's algorithm generalizes to the rings under consideration, and terminates because of noetherianity. ∎

Example. We perform the same computation as above on the Jacobi polynomials via Gröbner bases. This is achieved by defining a lexicographic order on the variables, with $S_n \prec D_x$:

```
T:=termorder(A,plex=[Sn,Dx]):
```

Next, a Gröbner basis with respect to this order is computed. It contains two polynomials: the mixed difference-differential operator (8) and the differential operator (4) which is seen by selecting only those terms without S_n:

```
remove(has,gbasis(G,T,ratpoly(rational,[a,b,x,n])),Sn);
```

$$\left[-n^2 - n - na - nb + x^2 D_x^2 + 2xD_x + bxD_x + acD_x + aD_x - bD_x - D_x^2\right]$$

Similarly, one could obtain (5) by eliminating the shift operator between (4) and (8).

2. Rational Ore algebras and ∂-finiteness

Solutions of linear recurrence or differential equations with polynomial coefficients are of particular interest to computer algebra and combinatorics, since they can be specified by a finite amount of information: the coefficients and a finite number of initial conditions. This has led D. Zeilberger to generalize the notions of P-recursive sequences and D-finite functions studied by R. Stanley [24] into a notion of P-finiteness [33]. In several variables, a function is P-finite when the vector space generated by its derivatives has finite dimension over the field of rational functions. Similarly, a sequence is P-finite when the vector space generated by its shifts has finite dimension over the field of rational functions. This has a simple translation in terms of ideals, and this translation yields a very natural generalization in the context of Ore algebras.

Definition 3. Let $\mathbb{O} = \mathbb{K}(\mathbf{x})[\boldsymbol{\partial}; \boldsymbol{\sigma}, \boldsymbol{\delta}]$ be a rational Ore algebra over a field $\mathbb{K}$. A left ideal $\mathfrak{I}$ of $\mathbb{O}$ is ∂*-finite* if $\mathbb{O}/\mathfrak{I}$ is finite-dimensional over $\mathbb{K}(\mathbf{x})$.

Functions, series, distributions, sequences, etc which are annihilated by such an ideal will also be called ∂-finite.

When $\mathfrak{I}$ is the annihilating ideal $\operatorname{Ann} f$ of a function f, the quotient $\mathbb{O}/\operatorname{Ann} f$ is isomorphic to the $\mathbb{O}$-module $\mathbb{O} \cdot f$ and this quotient is finite-dimensional if and only if the successive pseudo-derivatives $\partial_1^{\alpha_1} \cdots \partial_r^{\alpha_r}(f)$ of f span a finite-dimensional vector space over $\mathbb{K}(\mathbf{x})$.

When $\mathfrak{I}$ is a ∂-finite ideal, then $\mathbb{O}/\mathfrak{I}$ is isomorphic to the module $\mathbb{O} \cdot f$ where f is the residue class of 1 in $\mathbb{O}/\mathfrak{I}$. This f corresponds to a generic function annihilated by $\mathfrak{I}$. Thus ∂-finite ideals make it possible to express the properties and algorithms below without any reference to a specific algebra of functions. For any g annihilated by $\mathfrak{I}$, $\operatorname{Ann} g \supset \operatorname{Ann} f = \mathfrak{I}$. For instance, if $\mathfrak{I}$ is generated by $\partial_x^2 + 1$ in $\mathbb{O} = \mathbb{C}(x)[\partial_x; 0, \partial_x]$, then $\mathbb{O} \cdot f$ is isomorphic to either $\mathbb{O} \cdot \cos(x)$ or $\mathbb{O} \cdot \sin(x)$. Besides, $g = (-i\partial + 1)f$ corresponds to $e^{\pm ix}$; it is cancelled by $\mathfrak{I}$ and $\mathbb{O} \cdot g$ is a strict sub-module of $\mathbb{O} \cdot f$.

The study of ∂-finite ideals is motivated by their nice closure properties and the relative simplicity of the corresponding algorithms.

2.1. **Rectangular systems.** To simplify the proofs, we first note that ∂-finite ideals contain systems of polynomials of a special shape which we call *rectangular*.

Definition 4. A system of polynomials of a rational Ore algebra $\mathbb{O} = \mathbb{K}(\mathbf{x})[\boldsymbol{\partial}; \boldsymbol{\sigma}, \boldsymbol{\delta}]$ is *rectangular* when each ∂_i is involved in exactly one of its elements. As a consequence, each operator involves only one ∂_i.

There is no loss of generality in considering systems of this special form, as follows from the next proposition.

Proposition 2. *An ideal of a rational Ore algebra $\mathbb{O} = \mathbb{K}(\mathbf{x})[\boldsymbol{\partial}; \boldsymbol{\sigma}, \boldsymbol{\delta}]$ is ∂-finite if and only if it contains a rectangular system.*

Proof. If $\mathfrak{I}$ is a ∂-finite ideal, then for each i, $\{1, \partial_i, \partial_i^2, \dots\}$ spans a finite-dimensional vector space over $\mathbb{K}(\mathbf{x})$ in $\mathbb{O}/\mathfrak{I}$, from which follows the existence of a polynomial in ∂_i with coefficients in $\mathbb{K}(\mathbf{x})$ which becomes zero in the quotient (i.e., belongs to the ideal). Conversely, if $\mathfrak{I}$ contains a rectangular system with k_i the degree of the polynomial in ∂_i, then $\mathbb{O}/\mathfrak{I}$ is generated by $\{\partial_1^{p_1} \cdots \partial_n^{p_n}\}_{0 \le p_i < k_i}$ as a $\mathbb{K}(\mathbf{x})$-vector space. ■

A consequence of this proposition is that proving the ∂-finiteness of a "function" in a rational Ore algebra $\mathbb{O}$ reduces to proving that it is cancelled by a rectangular system of operators in $\mathbb{O}$. As an example of application, an important subclass of ∂-finite functions is often provided by *rational functions*.

Proposition 3. *Let $\mathbb{O} = \mathbb{K}(\mathbf{x})[\boldsymbol{\partial}; \boldsymbol{\sigma}, \boldsymbol{\delta}]$ be a rational Ore algebra which acts on an algebra of functions $\mathcal{F} \supseteq \mathbb{K}(\mathbf{x})$, making it an $\mathbb{O}$-module. If for all operators ∂_i of $\mathbb{O}$, the function $\gamma_i = \partial_i(1)$ is in $\mathbb{K}(\mathbf{x})$ then all rational functions of $\mathbb{K}(\mathbf{x})$ are ∂-finite with respect to $\mathbb{O}$.*

Proof. Let r be any rational function. Then $\partial_i(r) = \sigma_i(r)\gamma_i + \delta_i(r)$ is a rational function. Writing this fraction in the form (A_i/B_i) for two polynomials A_i and B_i yields an operator $B_i\partial_i - A_i$ which cancels r. ■

Another very simple example of ∂-finite "functions" is provided by hypergeometric sequences, i.e., sequences $u_{n_1,\dots,n_p}$ such that $u_{n_1,\dots,n_i+1,\dots,n_p}/u_{n_1,\dots,n_p}$ is rational for all i. The corresponding rectangular system consists only of shift operators of order one.

By (the simple part of) Theorem 2 rational Ore algebras are Noetherian. Computations of Gröbner bases therefore always terminate in rational Ore algebras. It is possible to compute a rectangular system included in a ∂-finite ideal $\mathfrak{I}$ from a Gröbner basis of $\mathfrak{I}$ (for any order) as follows. For each ∂_i in the algebra, ∂_i^k is reduced modulo this basis for $k = 0, 1, \dots$ This reduction rewrites the ∂_i^k in terms of a finite number of monomials $\partial_1^{i_1} \cdots \partial_r^{i_r}$ independent of k. The algorithm stops when a linear dependency between the remainders is detected by Gaussian elimination. Note however that in general, the ideal generated by this rectangular system is smaller than the original ideal. This may lead to calculations where the final equations have larger order than the minimal one, since inclusion is reversed on the corresponding modules.

2.2. **Closure properties.** Given two ∂-finite "functions" f and g (or equivalently two ∂-finite ideals $\mathfrak{I}$ and $\mathfrak{K}$ of a rational Ore algebra $\mathbb{O}$ and generators f and g of the $\mathbb{O}$-modules $\mathbb{O}/\mathfrak{I}$ and $\mathbb{O}/\mathfrak{K}$), we show in this section that $f+g$ is also ∂-finite, we determine sufficient conditions for fg to be ∂-finite and we show how to perform computations using specializations of f and pseudo-derivatives of f.

In each case, the problem is first translated into the language of ideals and modules, then conditions on the rational Ore algebra for the resulting ideal to exist are derived. This is then made effective by providing algorithms which construct generators of the ideal under consideration. For each operation, we give two different algorithms. One inputs and outputs rectangular systems and can be applied in Ore algebras even when Gröbner bases cannot be computed. The other one is based on Gröbner bases and returns generators of an ideal which is generally larger (hence better).

We begin with the sum.

Lemma 1. *Let $\mathfrak{I}$ and $\mathfrak{K}$ be two ∂-finite ideals in a rational Ore algebra $\mathbb{O}$. The annihilating ideal for any sum $f + g$ where f is annihilated by $\mathfrak{I}$ and g is annihilated by $\mathfrak{K}$ is also ∂-finite.*

Proof. An operator $P \in \mathbb{O}$ is applied to $f + g$ by $P(f + g) = P(f) + P(g)$. The first summand can be reduced modulo $\mathfrak{I} = \operatorname{Ann} f$, while the second summand can be reduced modulo $\mathfrak{K} = \operatorname{Ann} g$. Thus the natural algebraic setting is the direct sum $\mathbb{T} = \mathbb{O}/\mathfrak{I} \oplus \mathbb{O}/\mathfrak{K} = \mathbb{O} \cdot f \oplus \mathbb{O} \cdot g$ (over $\mathbb{K}(\mathbf{x})$), which is of finite dimension, since both ideals are ∂-finite. ∎

A rectangular system for the sum can be computed using rectangular systems for $\operatorname{Ann} f$ and $\operatorname{Ann} g$. For each ∂ in the algebra, one reduces $\partial^k f$ and $\partial^k g$ for $k = 1, 2, 3, \dots$ in the sequence $f + g, \partial f + \partial g, \partial^2 f + \partial^2 g, \dots$ This eventually yields a rectangular system for $f + g$ by Gaussian elimination.

The ∂-finite ideal obtained in this way is not necessarily as large as possible. If Gröbner bases are given for both $\operatorname{Ann} f$ and $\operatorname{Ann} g$, then a Gröbner basis of the annihilating ideal of $f + g$ can be computed by noting that $\operatorname{Ann}(f + g) = \operatorname{Ann} f \cap \operatorname{Ann} g$. Thus as in the commutative case, a basis for this ideal is obtained by eliminating a new commutative variable t in $t \operatorname{Ann} f + (1 - t) \operatorname{Ann} g$. In the univariate case, this algorithm reduces to computing a left lcm, for instance by the extended skew gcd algorithm.

Another procedure, which will also apply to other operations, consists in applying a suitable version of the FGLM algorithm [10] (more precisely by the *NewBasis* algorithm of [10], using both Gröbner bases to define the *NormalForm* function).

Example. We compute annihilators for the sum of the exponential function $f(x, y) = \exp(\mu x + \nu y)$ and of the product of Bessel functions $g(x, y) = J_\mu(x) J_\nu(y)$.

The functions f and g are defined by the rectangular systems

$$\begin{cases} f_x - \mu f = 0, \\ f_y - \nu f = 0 \end{cases} \quad \text{and} \quad \begin{cases} x^2 g_{x,x} + x g_x + (x^2 - \mu^2) g = 0, \\ y^2 g_{y,y} + y g_y + (y^2 - \nu^2) g = 0 \end{cases}$$

respectively (indices denoting differentiation). Using rectangular systems, one gets two skew polynomials of order 3 for the sum, namely

$$\begin{aligned} p_1 = & x^2(x^2 - \mu^2 + \mu^2 x^2 + \mu x)\partial_x^3 - x(3\mu^2 - \mu^3 x + \mu^3 x^3 + \mu x^3 - 2\mu x - x^2)\partial_x^2 \\ & + (x^4 - x^2 + \mu^2 x^4 - \mu^2 - \mu^4 x^2 - \mu^3 x^3 + \mu^4 - 4\mu^2 x^2)\partial_x \\ & + \mu(\mu^2 + 3\mu^3 x - \mu^2 x^4 - \mu^4 - \mu x^3 - x^4 + \mu^4 x^2 + x^2 + 2\mu^2 x^2) \end{aligned}$$

and a similar polynomial (p_3 below) where the rôles of (μ, x, ∂_x) and (ν, y, ∂_y) have been exchanged.

We now show the use of the FGLM algorithm to compute a Gröbner basis of skew polynomials vanishing on the sum $s(x, y)$. First, the algorithm reduces s, $\partial_y(s)$, $\partial_x(s)$, $\partial_y^2(s)$, $\partial_x\partial_y(s)$ and detects that they are independent. Then $\partial_x^2(s)$ is reduced and found to satisfy a linear relation with the previous ones, expressed by the following skew polynomial:

$$\begin{aligned} p_2 = & -(x^2 - \mu^2 + x^2\mu^2 + \mu x)y^2\partial_y^2 + x^2(y^2 - \nu^2 + y^2\nu^2 + \nu y)\partial_x^2 \\ & -(x^2 - \mu^2 + x^2\mu^2 + \mu x)y\partial_y + x(y^2 - \nu^2 + y^2\nu^2 + \nu y)\partial_x \\ & -\mu^2 y^2\nu^2 + x^2\nu y + x^2 y^2\nu^2 - x^2\mu^2 y^2 + x^2\mu^2\nu^2 - \mu x y^2 + \mu x\nu^2 - \mu^2\nu y. \end{aligned}$$

Next, the algorithm continues by reducing $\partial_y^3(s)$ and finds a new relation

$$\begin{aligned} p_3 = & y^2(y^2 - \nu^2 + \nu y + y^2\nu^2)\partial_y^3 - y(y^3\nu + y^3\nu^3 - y^2 - 2\nu y - \nu^3 y + 3\nu^2)\partial_y^2 \\ & + (y^4 + y^4\nu^2 - y^3\nu^3 - y^2 - y^2\nu^4 - 4y^2\nu^2 - \nu^2 + \nu^4)\partial_y \\ & + \nu(-y^4 - y^4\nu^2 + y^2 - \nu^4 + 2y^2\nu^2 + \nu^2 + y^2\nu^4 - y^3\nu + 3\nu^3 y). \end{aligned}$$

Finally, reduction of $\partial_x\partial_y^2$ produces the skew polynomial

$$p_4 = y^2\partial_x\partial_y^2 - \mu y^2\partial_y^2 + y\partial_x\partial_y - \mu y\partial_y + (y^2 - \nu^2)\partial_x - \mu(y^2 - \nu^2).$$

Thus, the computation with the FGLM algorithm returns more information than a simple rectangular system. On this example, the rectangular system $\{p_1, p_3\}$ makes it possible to rewrite any derivative of $s(x, y)$ as a linear combination of 9 derivatives, while the more accurate output $\{p_2, p_3, p_4\}$ of the FGLM algorithm yields a basis of 5 derivatives only.

The second algorithm starts from the following system in $\mathbb{K}(x, y)[t][\partial_x; 1, \partial_x][\partial_y; 1, \partial_y]$:

$$\{t(\partial_x - \mu), t(\partial_y - \nu), (1 - t)(x^2\partial_x^2 + x\partial_x + x^2 - \mu^2), (1 - t)(y^2\partial_y^2 + y\partial_y + y^2 - \nu^2)\}.$$

Eliminating the variable t yields the same basis $\{p_2, p_3, p_4\}$ as above.

In order to deal with the product, we need more information on σ_i and δ_i in (6). A sufficient condition is easily found.

Lemma 2. *Let $\mathfrak{I}$ and $\mathfrak{K}$ be two ∂-finite ideals in a rational Ore algebra $\mathbb{O} = \mathbb{K}(\mathbf{x})[\boldsymbol{\partial}; \boldsymbol{\sigma}, \boldsymbol{\delta}]$. Assume that σ_i and δ_i are polynomials in ∂_i over $\mathbb{K}(\mathbf{x})$ for all i. Then the annihilating ideal for any product fg where f is annihilated by $\mathfrak{I}$ and g is annihilated by $\mathfrak{K}$ is also ∂-finite.*

As can be seen from Table 1, this hypothesis does not represent a severe restriction on the class of Ore algebras we consider.

Again, f and g in this lemma need not be interpreted as functions but as generators of the $\mathbb{O}$-modules $\mathbb{O}/\mathfrak{I}$ and $\mathbb{O}/\mathfrak{K}$.

Proof. Let $\sigma_i = A_i(\partial_i)$ and $\delta_i = B_i(\partial_i)$ be polynomials in ∂_i for $i \in \{1,\dots,r\}$. Instead of considering sums of the form $P(f)+Q(g)$, we need to consider linear combinations of monomials of the form $P(f)Q(g)$. The natural setting for this computation is the tensor product $\mathbb{T} = \mathbb{O}/\mathfrak{I} \otimes \mathbb{O}/\mathfrak{K} = \mathbb{O}\cdot f \otimes \mathbb{O}\cdot g$ (over $\mathbb{K}(\mathbf{x})$). The application of ∂_i to products of the above type is translated into the following action which reflects (6):

$$\partial_i(P\otimes Q) = (A_i(\partial_i)P)\otimes(\partial_i Q) + (B_i(\partial_i)P)\otimes Q.$$

Computing an operator which cancels the product fg reduces to computing a polynomial which cancels $1\otimes 1$. Such a polynomial exists since $\mathbb{T}$ is finite dimensional. ∎

The algorithm to get a rectangular system which annihilates the product works as above by expressing the $\partial^k(fg)$, $k=1,2,\dots$ in the finite basis $\partial^i f\otimes\partial^j g$ and using Gaussian elimination to get an operator for each ∂ in the algebra. Once again, if Gröbner bases are given for $\operatorname{Ann} f$ and $\operatorname{Ann} g$ then a Gröbner basis of the (generally larger) annihilating ideal for fg is obtained by the FGLM algorithm of [10].

Linear combinations of pseudo-derivatives of a ∂-finite “function” are also ∂-finite.

Lemma 3. *Let $\mathfrak{I}$ be a ∂-finite ideal of a rational Ore algebra $\mathbb{O} = \mathbb{K}(\mathbf{x})[\boldsymbol{\partial};\boldsymbol{\sigma},\boldsymbol{\delta}]$. Let P be any skew polynomial in $\mathbb{O}$. Then for any f annihilated by $\mathfrak{I}$, $\operatorname{Ann} P(f)$ is also ∂-finite.*

Proof. This follows from the inclusion $\mathbb{O}\cdot Pf \subseteq \mathbb{O}\cdot f$ and the ∂-finiteness of $\operatorname{Ann} f \supset \mathfrak{I}$. ∎

The algorithm to find an operator vanishing on $P(f)$ consists in rewriting successive $\partial^\alpha P(f)$ in the finite basis formed with the pseudo-derivatives of f, and then finding a linear dependency by Gaussian elimination.

Putting all three lemmas together yields the following result for polynomial expressions in ∂-finite functions.

Proposition 4. *Let $\mathfrak{I}_1,\dots,\mathfrak{I}_n$ be ∂-finite ideals of a rational Ore algebra $\mathbb{O} = \mathbb{K}(\mathbf{x})[\boldsymbol{\partial};\boldsymbol{\sigma},\boldsymbol{\delta}]$. Assume that σ_i and δ_i are polynomials in ∂_i over $\mathbb{K}(\mathbf{x})$ for all $i\in\{1,\dots,r\}$. Let P be an element of the polynomial ring $\mathbb{O}[u_1,\dots,u_n]$ and f_i be annihilated by $\mathfrak{I}_i$, $i=1,\dots,n$. Then $P(f_1,\dots,f_n)$ is ∂-finite with respect to $\mathbb{O}$.*

In practice, one can apply the algorithms outlined above directly on $P(f_1,\dots,f_n)$, instead of decomposing into sums of products. This has the nice property of often producing equations of a lower order (i.e., larger ideals).

Example. Cassini's identity on the Fibonacci numbers reads

$$F_{n+2}F_n - F_{n+1}^2 = (-1)^n,$$

with $F_0 = F_1 = 1$ and $F_{n+2} = F_{n+1} + F_n$. In the Ore algebra $\mathbb{Q}[S_n; S_n, 0]$, the annihilating ideal $\mathfrak{I} = \operatorname{Ann} f$ of the Fibonacci numbers is generated by $S_n^2 - S_n - 1$. We consider the polynomial

$$P = S_n^2(f).f - S_n(f)^2.$$

First each of the S_n^i is reduced modulo $\mathfrak{I}$, so that P is rewritten

$$P = S_n(f).f + f^2 - S_n(f)^2.$$

Then S_nP is reduced similarly, and this yields

$$S_nP = -S_n(f).f - f^2 + S_n(f)^2.$$

Thus Gaussian elimination detects that S_n+1 annihilates P, whereas the decomposition into sums of products yields the less precise annihilator $S_n^3 - 2S_n^2 - 2S_n + 1$.

The specialization of a ∂-finite function at a point is also ∂-finite.

Proposition 5. *Let $\mathbb{O} = \mathbb{K}(\mathbf{x})(y_1, \ldots, y_q)[\boldsymbol{\partial}; \boldsymbol{\sigma}, \boldsymbol{\delta}]$ be a rational Ore algebra, the function $f(\mathbf{x}, \mathbf{y})$ be ∂-finite with respect to $\mathbb{O}$ and $a_1, \ldots, a_q$ be elements of $\mathbb{K}$. Then $g(\mathbf{x}) = f(\mathbf{x}, a_1, \ldots, a_q)$ is ∂-finite with respect to $\mathbb{O}' = \mathbb{K}(\mathbf{x})[\boldsymbol{\partial}; \boldsymbol{\sigma}, \boldsymbol{\delta}]$, and a rectangular system of $\operatorname{Ann} g$ can be computed (in $\mathbb{O}'$) from a system of generators of $\operatorname{Ann} f$.*

Again, this proposition could also be stated at the level of ∂-finite ideals.

Proof. Starting from a rectangular system for f, the algorithm simply consists in replacing $y_1, \ldots, y_q$ by $a_1, \ldots, a_q$ in the polynomials involving those ∂_i's that commute with the y_j, $j = 1, \ldots, q$ and discarding the other ones. This process does not yield trivial equations provided (left) polynomial factors are removed from the input polynomials before substitutions. ■

If a set of generators of the ideal $\operatorname{Ann} f$ is given, for instance as a Gröbner basis calculated by closure operations, a system for $\operatorname{Ann} g$ is obtained by eliminating (by a Gröbner basis computation) the ∂_j's that do not commute with $y_1, \ldots, y_q$ and then replacing $y_1, \ldots, y_q$ by $a_1, \ldots, a_q$. This system is not necessarily rectangular.

3. Creative telescoping

The main success of D. Zeilberger's theory of holonomic functions is *creative telescoping* [1, 28, 32, 33, 34, 35]. This is an algorithm to compute equations satisfied by definite sums or integrals. We now generalize this algorithm to *polynomial* Ore algebras $\mathbb{O} = \mathbb{K}[\mathbf{x}][\boldsymbol{\partial}; \boldsymbol{\sigma}, \boldsymbol{\delta}]$.

3.1. Indefinite ∂^{-1} and definite $\partial^{-1}|_\Omega$.

Let $\mathfrak{I}$ be a ∂-finite ideal and f be a generator of the module $\mathbb{O}/\mathfrak{I}$ (for instance f can be the class of 1). Let ∂_i be one of the operators of $\mathbb{O}$, and let $\mathbf{x}_i = \{x_{i_1}, \ldots, x_{i_k}\}$ denote the set of those variables x_j's of $\mathbb{O}$ which do not commute with ∂_i. (In practice this set is often reduced to $\{x_i\}$).

We assume an *indefinite* operator ∂_i^{-1} and a *definite* operator $\partial_i^{-1}|_\Omega$ exist, with the property that they commute with all the ∂_j's of the algebra, as well as with the x_j's, for $x_j \notin \mathbf{x}_i$. In addition, they satisfy

$$\partial_i^{-1}\partial_i = \partial_i\partial_i^{-1} = 1 \qquad \text{and} \qquad \partial_i^{-1}|_\Omega\partial_i = \partial_i\partial_i^{-1}|_\Omega = 0. \tag{9}$$

The indefinite operator ∂_i^{-1} corresponds to the indefinite sum or integration operator when ∂_i is the difference or differentiation operator, provided the set of functions $\mathcal{F}$ satisfies some analytic conditions. For instance, ∂_x and $\int_{-\infty}^x$ commute on $\mathbb{Q}[x]e^{-x^2}$. Similarly $\Delta_n = S_n - 1$ and $\sum_{-\infty}^{n-1}$ commute on many expressions involving binomial coefficients. The analytic conditions correspond to setting constants of integration or summation, so that Eq. (9) is satisfied. In the same cases, the definite operator $\partial_i^{-1}|_\Omega$ corresponds to the definite sum or integration operator respectively. Besides, in many cases of interest the value of the definite operator $\partial_i^{-1}|_\Omega$ on a "function" f is the value of $\partial^{-1}(f)$ when x is given a value. For instance, on the set of functions $\mathbb{Q}[x]e^{-x^2}$, $\partial_x^{-1}|_\mathbb{R} = \lim_{x\to+\infty} \int_{-\infty}^x$. Similarly $\Delta_n^{-1}|_\mathbb{Z} = \lim_{n\to+\infty} \sum_{-\infty}^{n-1}$ for many expressions involving binomial coefficients.

To compute indefinite ∂_i^{-1} or definite $\partial_i^{-1}|_\Omega$, the first step of creative telescoping consists in finding a polynomial $P \in \mathfrak{I}$ which does not contain any element of $\mathbf{x}_i$. Euclidean division by ∂_i can then be used to produce two polynomials A and B which do not contain any element of $\mathbf{x}_i$ and such that

$$P(f) = 0 = \partial_i A(f) + B(f). \tag{10}$$

Next, left multiplying by ∂_i^{-1} and using the commutation rules for ∂_i^{-1} yields

$$B\partial_i^{-1}(f) = -A(f).$$

The next step depends on whether we are computing a definite $\partial_i^{-1}|_\Omega$ or an indefinite ∂_i^{-1}. When computing a definite integral or sum over a domain on the border $\partial\Omega$ of which all the $\boldsymbol{\partial} f$ vanish (which is often the case in practice), the right-hand side of the above equation is 0, and B is therefore the polynomial we are looking for. Hence, this result is obtained by simply left multiplying Eq. (10) by $\partial_i^{-1}|_\Omega$ and using the commutation rules.

In the indefinite case, one appeals to Lemma 3 in order to compute polynomials C annihilating $A(f)$, then for such polynomials, CB is a polynomial annihilating $\partial_i^{-1} f$.

There are two difficulties with this technique, which both reside in the first step. The first one is to determine whether there exists a non-zero polynomial P in $\mathfrak{I}$ which does not contain any element of $\mathbf{x}_i$. The second one is to find such a polynomial, or better yet a basis of them when they exist.

Our approach consists in using a Gröbner basis computation to perform the elimination of the necessary x_i's. We are therefore led to work in *polynomial* Ore algebras (at least in the relevant x_i's) instead of rational Ore algebras. We then just have to compute a Gröbner basis for an appropriate elimination order. This basis contains polynomials free of the undesirable x_i's if such polynomials exist in the ideal. (See however the comments in Section 4.3.)

3.2. **Example of creative telescoping by Gröbner bases.** We illustrate the use of Gröbner bases in Ore algebras to compute annihilators of the generating function of the Jacobi polynomials

$$\sum_{n=0}^{\infty} P_n^{(a,b)}(x) y^n.$$

We create a new Ore algebra to accommodate the extra variable y:

```
with(Mgfun):
A:=orealg(comm=[a,b],shift=[Sn,n],diff=[Dx,x],diff=[Dy,y]):
```

To get the equations for $P_n^{(a,b)}(x)y^n$ one could use the equations for $P_n^{(a,b)}(x)$, define y^n as a solution of $\{S_n - y, y\partial_y - n\}$ and appeal to closure under product (Section 2.2). A more direct way consists in noting that the differential equation (4) is also satisfied by $P_n^{(a,b)}(x)y^n$, while a recurrence is obtained by changing S_n into $y^{-1}S_n$ in recurrence (5). This gives

```
G:=[2*(n+2)*(n+a+b+2)*(2*n+a+b+2)*Sn^2
        -(2*n+a+b+3)*(a^2-b^2+4*x*n^2+4*x*n*a+4*x*n*b+12*x*n+x*a^2+2*x*a*b
        +6*x*a+x*b^2+6*x*b+8*x)*Sn*y+2*(n+a+1)*(n+b+1)*(2*n+a+b+4)*y^2,
    -2*(n+a+1)*(n+b+1)*y+(n+1)*(-a+b+2*x*n+x*a+x*b+2*x)*Sn
        -(x-1)*(x+1)*(2*n+a+b+2)*Dx*Sn,
    n*(n+a+b+1)+(b-a-x*a-x*b-2*x)*Dx-(x-1)*(x+1)*Dx^2,
    y*Dy-n]:
```

To compute the sum for non-negative n, we start by eliminating n. We therefore define an appropriate order:

```
T:=termorder(A,lexdeg=[[n],[Sn,Dx,Dy]]):
```

The elimination is then obtained by a simple Gröbner basis computation:

```
GB:=gbasis(G,T,ratpoly(rational,[a,b,x,y])):
```

This basis consists of six polynomials which vanish on $P_n^{(a,b)}(x)y^n$, only the first one of which contains n. The next step of creative telescoping consists in substituting S_n by 1 in these operators (i.e., in computing the remainder of the Euclidean division by the difference operator $S_n - 1$):

```
CT:=collect(subs(Sn=1,[GB[2..6]]),[Dx,Dy],distributed,factor);
```

$$
\begin{aligned}
&[-2y(1+b)(1+a)(a+2xa+2xb-b+2x)-4y^3(yx-1)D_y^3\\
&\quad+(x-1)(x+1)(-4yb-4y-4ya-b^2-4yab+a^2)D_x\\
&\quad-y(-2yb^2+4yb^2x+12yxab+24yxb+6ya-6yb+28yx+4ya^2x\\
&\qquad+24yxa+2ya^2-6b-a^2+xa^2-b^2-6a-6ab-xb^2-4)D_y\\
&\quad-4y^2(x-1)(x+1)(a+b+3)D_xD_y\\
&\quad-4y^3(x-1)(x+1)D_xD_y^2-2y^2(-yb+ya+4yxa+4yxb+12yx-3a-6-3b)D_y^2,\\
&y(a^2+b^2+a+b-xb^2+xa-xb+xa^2)+(x-1)(x+1)(a+b-yb+ya)D_x\\
&\quad+y(ya+yxa-yxb+yb-xb-xa+b-a)D_y,\\
&(b-a-xa-xb-2x)D_x+y(2+a+b)D_y-(x-1)(x+1)D_x^2+y^2D_y^2,\\
&-2y(1+b)(1+a)-(x-1)(x+1)(b+a)D_x\\
&\quad+y(-6y-2ya-2yb-a+b+xa+2x+xb)D_y\\
&\quad-2y(x-1)(x+1)D_xD_y+2y^2(-y+x)D_y^2,\\
&-2y-2yab+3xb+3xa-b^2+a-b+xa^2+2x-2yb-2ya+a^2+xb^2\\
&\quad+(-2b-2-2a-2y^2b-6y^2+4yxa-2y^2a+8yx+4yxb)D_y\\
&\quad+2xab+2(x-1)(x+1)D_x+2y(-1+2yx-y^2)D_y^2]
\end{aligned}
$$

The whole computation takes 17 seconds[3]. It is then possible to compute a rectangular system out of these equations. From this system and initial conditions, a differential equation solver can find the generating function of the Jacobi polynomials

$$\frac{1}{\sqrt{1-2yx+y^2}\left(1-y+\sqrt{1-2yx+y^2}\right)^a\left(1+y+\sqrt{1-2yx+y^2}\right)^b}.$$

Even when solving is not possible, these equations can be used to check such a conjectured right-hand side (see below), or more importantly to proceed with further computations when no closed-form exists or is available. Here is the verification:

```
> R:=sqrt(1-2*x*y+y^2):
  P:=1/(R*(1-y+R)^a*(1+y+R)^b):
  map(simplify,map(applyopr,CT,P,A));
```

$$[0,0,0,0,0]$$

Checking the initial conditions at 0 then proves that this solution is the generating function that we were looking for.

3.3. **Takayama's algorithm for definite $\partial^{-1}|_\Omega$.** The elimination of the variables in $\mathbf{x}_i$ to perform creative telescoping is slightly stronger than what is strictly necessary. This can result in operators of order larger than necessary, or in a failure to compute the definite $\partial^{-1}|_\Omega$. It is actually sufficient to determine an element of the ideal $\mathfrak{I}$ which can be written $\partial_i A + B$, where only B needs to commute with ∂_i and where only B needs to be computed.

An elimination algorithm based on Gröbner bases for modules was developed by N. Takayama [27, 28] to solve this problem in the context of the Weyl algebra. This algorithm is readily adapted

[3] All our timings are obtained on a DecStation 3000 300X (Alpha).

to the context of Ore algebras and often results in faster computations. We now describe this algorithm.

Since the aim is to compute B, during the intermediate computations one can replace all the polynomials which can be rewritten $\partial_i C$ for some C by zero, provided these polynomials will not be multiplied by any x_j belonging to $\mathbf{x}_i$ in later computations. If the Ore algebra satisfies the hypothesis of Theorem 2 (which is necessary if we want to compute Gröbner bases), the idea is that this simplification can be achieved by computing Gröbner bases of (not finitely generated) $\mathbb{K}(\mathbf{x} \setminus \mathbf{x}_i)[\mathbf{x}_i][\boldsymbol{\partial}; \boldsymbol{\sigma}, \boldsymbol{\delta}]$-modules.

The hypothesis of Theorem 2 on the a_{ij}'s implies that all the $x_j^p \partial_i^k$ for $x_j \in \mathbf{x}_i$ can be rewritten as polynomials of lower degree in ∂_i using

$$\partial_i^k x_j^p = c x_j^p \partial_i^k + \text{lower order terms} \tag{11}$$

provided the left hand-side can be replaced by zero. The algorithm then considers the $\mathbb{K}(\mathbf{x} \setminus \mathbf{x}_i)[\mathbf{x}_i][\boldsymbol{\partial}; \boldsymbol{\sigma}, \boldsymbol{\delta}]$-sub-modules of $\mathfrak{I} + \partial_i \mathbb{O}$ consisting of polynomials of total degree less than or equal to N in $\mathbf{x}_i$ for $N = 0, 1, 2, \ldots$ A suitable generalization of Gröbner bases of these modules is computed in three steps: first the generators of $\mathfrak{I}$ are left-multiplied by powers of the elements of $\mathbf{x}_i$ to produce all of the possible operators of degree less than or equal to N in $\mathbf{x}_i$. Then ∂_i is eliminated from these operators using (11). Finally a generalized Gröbner basis for this set of operators in $\mathbb{K}(\mathbf{x} \setminus \mathbf{x}_i)[\mathbf{x}_i][\boldsymbol{\partial} \setminus \partial_i]$ is computed by the usual Buchberger algorithm, except that multiplications by elements of $\mathbf{x}_i$ are not allowed when computing syzygies or reductions.

An optimized version of the algorithm is as follows (here we denote by $G_0 \subset G$ the set of polynomials in G of degree 0 in $\mathbf{x}_i$).

 # The input is a set P of operators
 # $P_1, \ldots, P_p$ of degree $d_1, \ldots, d_p$ in $\mathbf{x}_i$
$G := \{\}$
for N from $\min(d_1, \ldots, d_p)$ while $G_0 = \{\}$ do
  $H := \{x_{i_1}^{p_1} \cdots x_{i_k}^{p_k} P_i; p_j \geq 0, \sum p_j = N - d_i, d_i \leq N, x_{i_j} \in \mathbf{x}_i\}$
  reduce H using (11)
  $G :=$ module-gbasis$(G \cup H)$
od
return G_0

It is worth noting that the reduction by (11) is usually very simple. In the case of a differential operator ∂_x, it consists in replacing monomials $p(x)\partial_x^k$ by $(-1)^k p^{(k)}(x)$. In the case of a shift operator S_n, it consists in replacing monomials $p(n)S_n^k$ by $p(n-k)$. In the case of the q-shift operator S_n, it consists in replacing monomials $p(n, q^n)S_n^k$ by $p(n-k, q^{n-k})$.

The condition under which the algorithm should be stopped can be modified depending on the context. In the Weyl algebra case, N. Takayama chooses to stop the loop when the basis generates a holonomic ideal and he proves that this always happens in finite time. We do not have such a result in the general Ore algebra case. Thus we stop the algorithm as soon as one polynomial free of $\mathbf{x}_i$ has been found and termination is not guaranteed unless there exists such a polynomial (i.e., the definite $\partial_i^{-1}|_\Omega$ can be found by creative telescoping). Termination can only be guaranteed for special cases of ideals such as holonomic ideals in the Weyl algebra.

The speed of this algorithm compared to the general one described in Section 3.1 may well make it the only practical one on large examples. However, it is worth noting that this algorithm computes in a different ideal than the general method. Thus the ideal generated by its output when stopping the loop as soon as a polynomial free of $\mathbf{x}_i$ has been found may be larger or smaller than the ideal obtained by the other algorithm. (But running the loop forever computes a sequence of ideals with is stationary on a larger ideal than the one obtained by the other algorithm.) In practice, this new algorithm often returns operators of a smaller order than the general method. This increases the speed of subsequent computations.

Example. In the same example as above, the computation now takes place in the simpler algebra

```
A:=orealg(comm=[a,b,n],diff=[Dx,x],diff=[Dy,y]):
```

It takes less than 6 seconds to find the following set of three operators which annihilate the generating function of the Jacobi polynomials:

$$\begin{cases} (x-1)(x+1)D_x^2 + (ax+bx+2x+a-b)D_x - y(a+b+2)D_y - y^2D_y^2, \\ 2y^2(x-y)D_y^2 + y(-2ay-2by-6y+2x+b-a+ax+bx)D_y \\ \qquad -(x-1)(x+1)(a+b)D_x - 2y(x-1)(x+1)D_xD_y - 2y(a+1)(b+1), \\ 4y^3(y^2-2xy+1)D_y^3 \\ \qquad -2y^2(-3y^2a-16y^2-3y^2b+6yxa+6yxb+22xy-3b-3a-6)D_y^2 \\ \qquad -y(-8y^2ab-26y^2a-52y^2-2y^2b^2-2y^2a^2-26y^2b-b^2y+32yxb+a^2y+40xy \\ \qquad\quad +5yb^2x+32yxa+14yaxb+5ya^2x-6b+a^2x-b^2x-4-a^2-6ab-6a-b^2)D_y \\ \qquad +(x-1)(x+1)(b^2y+a^2-4y-b^2-2yab+a^2y)D_x \\ \qquad -2y(1+b)(a+1)(-ay+2ax+a-by+2bx-b-6y+2x). \end{cases}$$

This is obtained with $N=2$. It is not difficult to check that the ideal generated by these operators is ∂-finite. The next iteration of the loop takes 22 more seconds and produces the same more refined basis as the general algorithm.

3.4. **Hypergeometric examples.** Apart from his general theory of holonomic identities [33], D. Zeilberger, together with H. Wilf, developed specialized algorithms for the cases of hypergeometric and q-hypergeometric identities [32] (see also [15]). It would be interesting to compare their efficiency to our approach and generalize as much as possible their good features. We now show using a few examples that the general approach outlined in this paper performs rather well in practice.

3.4.1. *An identity between Franel and Apéry numbers.*

The following identity was proved by V. Strehl [25]:

$$\sum_{k=0}^{n}\binom{n}{k}^2\binom{n+k}{k}^2 = \sum_{k=0}^{n}\binom{n}{k}\binom{n+k}{k}\sum_{j=0}^{k}\binom{k}{j}^3. \tag{12}$$

Both sides of this equation satisfy the operator

$$(n+2)^3S_n^2 - \left((n+2)^3+(n+1)^3+4(2n+3)^3\right)S_n + (n+1)^3. \tag{13}$$

(This operator was used by Apéry in his proof of the irrationality of $\zeta(3)$.)

Using the algorithm of Section 3.1, the computation is performed by *Mgfun* in 82 seconds. First, an operator of order 3 annihilating the inner sum of the right hand-side is obtained in 5 seconds; then 2 more seconds are necessary to compute operators annihilating the product by the two binomials using the technique of Section 2.2 and creative telescoping applied to these latter operators require 31 seconds to yield an operator of order 7 annihilating the right hand-side of (12). Another creative telescoping yields an operator of order 4 annihilating the left hand-side of (12) in 44 seconds. The identity is then proved by checking 11 initial conditions (the order of the operator annihilating the difference). Then taking the gcd of both operators yields (13).

A similar calculation using our version of Takayama's algorithm is performed in 11 seconds. Interestingly, the operators found by this method have a smaller order than those produced by the general algorithm. The inner sum of the right hand-side is found to satisfy an operator of order 2 in 4 seconds; then the product still takes 2 seconds and the second creative telescoping takes 2 seconds and yields an operator of order 2. The same operator is obtained by applying this algorithm to the left hand-side and the computation takes 2.5 seconds.

3.4.2. *A Rogers-Ramanujan identity.* We consider the following finite version due to Andrews of one of the famous Rogers-Ramanujan identities:

$$\sum_k \frac{q^{k^2}}{(q;q)_k(q;q)_{n-k}} = \sum_k \frac{(-1)^k q^{(5k^2-k)/2}}{(q;q)_{n-k}(q;q)_{n+k}},$$

where $(q;q)_n = (1-q)\cdots(1-q^n)$.

Using the general method of Section 3.1, it takes one second to find a second order operator annihilating the left hand-side of this identity, and 56 seconds to find a fifth order operator annihilating the right hand-side. From this a proof is easily derived as above. Our generalization of Takayama's algorithm finds *the same operators* as the general method in 1 second and 23 seconds respectively.

It was noted by P. Paule [21] that summing only the even part of the right-hand side (i.e. multiplying it by $(1+q^k)/2$) results in Zeilberger's algorithm finding an operator of order 2 for the right-hand side. Using the same trick with our algorithms, we find that Takayama's method benefits from it and yields an operator of order 3 instead of 5, while the more general algorithm yields an operator of order 6. As in the hypergeometric case, the reasons for this trick to work or not to work remain mysterious.

3.4.3. *q-Dixon identity.* The left hand-side of the identity

$$\sum_k (-1)^k q^{k(3k+1)/2} \binom{a+b}{a+k}_q \binom{a+c}{c+k}_q \binom{b+c}{b+k}_q = \frac{(q;q)_{a+b+c}}{(q;q)_a(q;q)_b(q;q)_c}$$

satisfies a system of operators in the variables S_a, S_b and S_c which can be obtained in 490 seconds using *Mgfun*. A simpler (but less complete) system is obtained using our version of Takayama's algorithm in only 70 seconds. Here is this simpler system:

$$\begin{cases} q^a(q^{b+c+1}-1)S_a - q^b(q^{a+c+1}-1)S_b + (q^a-q^b)S_aS_b, \\ q^a(q^{b+c+1}-1)S_a - q^c(q^{a+b+1}-1)S_c + (q^a-q^c)S_cS_a, \\ q^c(q^{a+b+1}-1)S_c - q^b(q^{a+c+1}-1)S_b + (q^c-q^b)S_cS_b. \end{cases}$$

It can be checked that these operators do not generate a ∂-finite ideal. One more iteration of the algorithm takes 166 seconds to produce generators of a ∂-finite ideal.

Note that the same computation could be performed in an algebra containing the differentiation operator ∂_a instead of the shift operator S_a. What happens then is that our algorithm does not produce any operator in ∂_a (no such operator exists), but only the operators in S_b and S_c.

4. Conclusion

4.1. **Holonomy.** In the context of the Weyl algebra, D. Zeilberger used I. N. Bernstein's theory of *holonomic* systems to outline an important class of "functions" enjoying numerous closure properties and for which the elimination of any x_i is always guaranteed to succeed. He extended this technique to sequences and definite summation by considering generating functions. A nice property of the Weyl algebra is that holonomy is equivalent to ∂-finiteness [14] (see also [29]). More precisely, if $\mathfrak{I}$ is a ∂-finite ideal in $\mathbb{K}(\mathbf{x})[\boldsymbol{\partial}]$, then $\mathfrak{I} \cap \mathbb{K}[\mathbf{x}][\boldsymbol{\partial}]$ is holonomic.

Unfortunately, this equivalence breaks down in the case of general Ore algebras, which is why in this paper, we have focussed on ∂-finite functions and on equations with rational functions coefficients. I. N. Bernstein's theory of holonomy [3, 4] deals with polynomial coefficients and relies on a theory of dimension for ideals and modules. In a Weyl algebra on n differentiation symbols $\partial_1, \dots, \partial_n$, holonomic modules are those of least possible Bernstein dimension, namely n. Thus it is easy to check whether an ideal is holonomic when a set of its generators has been given (via Gröbner basis computations for instance). The difficulty in the case of Ore algebras consists in finding a class of ideals of Bernstein dimension less than or equal to n closed under product. This will be the subject of future work.

4.2. **The Weyl algebra case.** It is well-known that in the special case of the Weyl algebra, many algorithms make it possible to compute equations for interesting operations. These operations apply to both univariate and multivariate cases.

In particular, algebraic functions are holonomic and an algorithm to compute differential equations from the polynomial equation exists [8]. Also, the composition of a holonomic function with algebraic functions is again holonomic and equations can be computed [17, 24].

Holonomic functions are defined as solutions of differential equations with polynomial (or equivalently, rational) coefficients. There is in fact no enlargement of the class if we allow algebraic functions as coefficients: a function that satisfies a rectangular system with algebraic coefficients is holonomic and annihilators with polynomial coefficients can be computed.

Diagonals of holonomic functions are holonomic, and this is also effective [16]. This leads to the result that the Hadamard product of two holonomic power series is again holonomic, and again equations can be computed. Also, recurrence equations satisfied by the coefficients of a holonomic power series can be computed. All these operations are implemented in the univariate case in *gfun* [23] and are or will be implemented in the multivariate case in F. Chyzak's *Mgfun* package.

4.3. **The extension/contraction problem.** The functions that we work with are naturally defined by operators in a *rational* Ore algebra $\mathbb{O}_{\mathrm{r}} = \mathbb{K}(\mathbf{x})[\boldsymbol{\partial}; \boldsymbol{\sigma}, \boldsymbol{\delta}]$, while the algorithms we use for creative telescoping need elimination of some of the x_i's. We therefore need to describe functions with operators in the smaller *polynomial* Ore algebra $\mathbb{O}_{\mathrm{p}} = \mathbb{K}[\mathbf{x}][\boldsymbol{\partial}; \boldsymbol{\sigma}, \boldsymbol{\delta}]$. Let $p_1, \dots, p_r$ be polynomials in $\mathbb{O}_{\mathrm{p}}$ generating a left ideal $\mathfrak{I} \subseteq \mathbb{O}_{\mathrm{p}}$, then they also generate an ideal $\mathfrak{K} \subseteq \mathbb{O}_{\mathrm{r}}$. However the actual ideal we are interested in is the *contraction* $\mathfrak{I}' = \mathfrak{K} \cap \mathbb{O}_{\mathrm{p}}$, which can be larger than the original ideal $\mathfrak{I}$. Unfortunately, at the moment we do not have any algorithm to compute a basis of this ideal $\mathfrak{I}'$ from a basis of $\mathfrak{K}$. In the commutative case, the contraction $\mathfrak{I} \cap \mathbb{K}[\mathbf{x}, \mathbf{y}]$ of an ideal $\mathfrak{I} \subseteq \mathbb{K}(\mathbf{x})[\mathbf{y}]$ can be computed. An algorithm [2, algorithms CONT and IDEALDIV2] is based on the calculation of the *ideal quotient* $\mathfrak{I}/f^\infty$, i.e., the set of all p such that $f^s p \in \mathfrak{I}$ for a positive integer s. This algorithm does not extend trivially to the skew case. Another algorithm for the commutative case, called the *tangent cone algorithm* [18, 19] was designed by T. Mora to overcome a similar difficulty. (More precisely, it computes a basis for $\mathbb{K}[[\mathbf{x}]] \otimes \mathfrak{I}$ when a basis for an ideal $\mathfrak{I}$ of $\mathbb{K}[\mathbf{x}]$ is known.) In the future, we hope to extend one of these algorithms to Ore algebras.

In the case of a ∂-finite function f, this extension/contraction problem means that even if we are given generators $(p_1, \dots, p_r)$ of the ideal $\mathfrak{K}_f$ of all polynomials in $\mathbb{O}_{\mathrm{r}}$ that vanish at f, the ideal $\mathfrak{I} = (p_1, \dots, p_r) \subseteq \mathbb{O}_{\mathrm{p}}$ is not necessarily an accurate description of f. Therefore, elimination of one x_k between the p_i's may lead to zero, even when $\mathfrak{K}_f \cap \mathbb{K}[\mathbf{x} \setminus x_k][\boldsymbol{\partial}; \boldsymbol{\sigma}, \boldsymbol{\delta}]$ contains a non-zero polynomial.

Example. The binomial coefficients $u_{n,k} = \binom{n}{k}$ are annihilated by the skew polynomials $P = (n+1-k)S_n-(n+1)$ and $Q = (k+1)S_k-(n-k)$ in the rational Ore algebra $\mathbb{O}_\mathrm{r} = \mathbb{K}(n,k)[S_n; S_n, 0][S_k; S_k, 0]$ built on two shift operators S_n and S_k. Any ideal larger than $\mathfrak{K} = (P, Q)$ in $\mathbb{O}_\mathrm{r}$ is $\mathbb{O}_\mathrm{r}$ itself. Pascal's triangle rule is represented by the operator $R = S_nS_k - S_k - 1$, which is easily found to be an element of $\mathfrak{K}$. Therefore $R \in \mathfrak{I}'$. However, in the difference algebra $\mathbb{O}_\mathrm{p} = \mathbb{K}[n,k][S_n; S_n, 0][S_k; S_k, 0]$, the ideal $\mathfrak{I} = (P, Q)$ does *not* contain R, although it contains $(n + 1)R$, and $(k + 1)R$, which is sufficient to make it possible to find the result R by Gröbner basis computation (with ideals in $\mathbb{K}(n)[k][S_n; S_n, 0][S_k; S_k, 0]$).

Example. Diagonals can be computed by creative telescoping. If $f(x, y)$ is a ∂-finite power series, then its diagonal is the coefficient of s^{-1} in $F(x, s) = f(s, x/s)/s$. By Cauchy's theorem, this is obtained by computing the definite integral of F with respect to s. From generators of an ideal annihilating f, it is not difficult to obtain generators of an ideal $\mathfrak{K} \subseteq \mathbb{O}_\mathrm{r} = \mathbb{K}(s,x)[\partial_s; 1, \partial_s][\partial_x; 1, \partial_x]$ annihilating F, and then one has to eliminate s. However, the success of this elimination with our algorithms requires generators of a sufficiently large ideal $\mathfrak{I} \subseteq \mathfrak{K} \cap \mathbb{O}_\mathrm{p}$, with $\mathbb{O}_\mathrm{p} = \mathbb{K}[s,x][\partial_s; 1, \partial_s][\partial_x; 1, \partial_x]$, such that $\mathfrak{I}$ contains a polynomial free of s.

For instance, to compute the diagonal of $f = 1/(1 - (x + y))$ requires finding an operator free of s in $\mathbb{O}_p$ which cancels $F = 1/(s^2 - s + x)$. The annihilating ideal of F in $\mathbb{O}_\mathrm{r}$ is $\mathfrak{K} = (P, Q)$ where $P = \partial_s(s^2-s+x) = (s^2-s+x)\partial_s + (2s-1)$ and $Q = \partial_x(s^2-s+x) = (s^2-s+x)\partial_x + 1$. Once again, any larger ideal in $\mathbb{O}_\mathrm{r}$ is $\mathbb{O}_\mathrm{r}$ itself. The operator $U = \partial_s^2 + (4x - 1)\partial_x^2 + 6\partial_x$ vanishes at F, so that $U \in \mathfrak{K}$, hence $U \in \mathfrak{K} \cap \mathbb{O}_\mathrm{p}$. However, U is *not* an element of $\mathfrak{I} = (P, Q)$ in $\mathbb{O}_\mathrm{p}$. It follows that the calculation of the diagonal of f from P and Q cannot be performed by elimination in $\mathbb{O}_\mathrm{p}$. However, if one takes the generator $R = (s^2 - s + x)\partial_s\partial_x - 2\partial_s \in \mathfrak{K}$, then the ideal $(R, Q) \subseteq \mathbb{O}_\mathrm{p}$ contains the operator U and our algorithm finds it. Furthermore, our version of Takayama's algorithm then finds the simpler $(1 - 4x)\partial_x - 2$.

Acknowledgement. This work was supported in part by the Long Term Research Project Alcom-IT (#20244) of the European Union.

References

[1] Almkvist, G., and Zeilberger, D. The method of differentiating under the integral sign. *Journal of Symbolic Computation 10* (1990), 571–591.

[2] Becker, T., and Weispfenning, V. *Groebner bases*, vol. 141 of *Graduate texts in mathematics*. Springer Verlag, 1993. A computational approach to commutative algebra (in cooperation with Heinz Kredel).

[3] Bernstein, I. N. Modules over a ring of differential operators, study of the fundamental solutions of equations with constant coefficients. *Functional Analysis and Applications 5*, 2 (1971), 1–16 (Russian); 89–101 (English translation).

[4] Bernstein, I. N. The analytic continuation of generalized functions with respect to a parameter. *Functional Analysis and Applications 6*, 4 (1972), 26–40 (Russian); 273–285 (English translation).

[5] Cartier, P. Démonstration 'automatique' d'identités et fonctions hypergéométriques. *Astérisque 206* (1992), 41–91. Séminaire Bourbaki.

[6] Chyzak, F. Holonomic systems and automatic proofs of identities. Research Report 2371, Institut National de Recherche en Informatique et en Automatique, Oct. 1994.

[7] Cohn, P. M. *Free Rings and Their Relations.* No. 2 in London Mathematical Society Monographs. Academic Press, 1971.

[8] Comtet, L. Calcul pratique des coefficients de Taylor d'une fonction algébrique. *L'Enseignement Mathématique 10* (1964), 267–270.

[9] Cox, D., Little, J., and O'Shea, D. *Ideals, Varieties, and Algorithms.* Springer Verlag, 1992. An Introduction to Computational Algebraic Geometry and Commutative Algebra.

[10] Faugère, J., Gianni, P., Lazard, D., and Mora, T. Efficient computation of zero-dimensional Gröbner bases by change of ordering. *Journal of Symbolic Computation 16* (Oct. 1993), 329–344.

[11] Galligo, A. Some algorithmic questions on ideals of differential operators. In *Proceedings EUROCAL'85* (1985), B. F. Caviness, Ed., vol. 204 of *Lecture Notes in Computer Science*, Springer Verlag, pp. 413–421.

[12] Giovini, A., Mora, T., Niesi, G., Robbiano, L., and Traverso, C. "One sugar cube, please" or selection strategies in the Buchberger algorithm. In *Symbolic and algebraic computation* (1991), S. M. Watt, Ed., ACM Press, pp. 49–54.

[13] Kandri-Rody, A., and Weispfenning, V. Non-commutative Gröbner bases in algebras of solvable type. *Journal of Symbolic Computation 9* (1990), 1–26.

[14] Kashiwara, M. On the holonomic systems of linear differential equations II. *Inventiones Mathematicae 49* (1978), 121–135.

[15] Koornwinder, T. H. On Zeilberger's algorithm and its q-analogue. *Journal of Computational and Applied Mathematics 48* (1993), 91–111.

[16] Lipshitz, L. The diagonal of a D-finite power series is D-finite. *Journal of Algebra 113* (1988), 373–378.

[17] Lipshitz, L. D-finite power series. *Journal of Algebra 122* (1989), 353–373.

[18] Mora, F. An algorithm to compute the equations of tangent cones. In *Proceedings of EUROCAM'82* (1982), J. Calmet, Ed., no. 144 in Lecture Notes in Computer Science, pp. 24–31.

[19] Mora, T. Seven variations on standard bases. Preprint 45, Università di Genova, Dipartimento di Matematica, Mar. 1988.

[20] Ore, O. Theory of non-commutative polynomials. *Annals of Mathematics 34* (1933), 480–508.

[21] Paule, P. Short and easy computer proofs of the Rogers-Ramanujan identities and of identities of similar type. *The Electronic Journal of Combinatorics 1*, R10 (July 1994), 1–9.

[22] Paule, P., and Schorn, M. A Mathematica version of Zeilberger's algorithm for proving binomial coefficient identities. Journal of Symbolic Computation, 1995. To appear.

[23] Salvy, B., and Zimmermann, P. Gfun: a Maple package for the manipulation of generating and holonomic functions in one variable. *ACM Trans. Math. Softw. 20*, 2 (1994), 163–177. Also available as Technical Report 143, INRIA.

[24] Stanley, R. P. Differentiably finite power series. *European Journal of Combinatorics 1* (1980), 175–188.

[25] Strehl, V. Binomial identities — combinatorial and algorithmic aspects. *Discrete Mathematics 136* (1994), 309–346.

[26] Takayama, N. Gröbner basis and the problem of contiguous relations. *Japan Journal of Applied Mathematics 6*, 1 (1989), 147–160.

[27] Takayama, N. An algorithm of constructing the integral of a module — an infinite dimensional analog of Gröbner basis. In *Symbolic and algebraic computation* (1990), ACM and Addison-Wesley, pp. 206–211. Proceedings of ISSAC'90, Kyoto.

[28] Takayama, N. Gröbner basis, integration and transcendental functions. In *Symbolic and algebraic computation* (1990), ACM and Addison-Wesley, pp. 152–156. Proceedings ISSAC'90, Kyoto.

[29] Takayama, N. An approach to the zero recognition problem by Buchberger algorithm. *Journal of Symbolic Computation 14* (1992), 265–282.

[30] Takayama, N. Private communication, 1995.

[31] Weispfenning, V. Finite Gröbner bases in non-Noetherian skew polynomial rings. In *Symbolic and Algebraic Computation* (New York, 1992), P. S. Wang, Ed., ACM Press, pp. 329–334. Proceedings ISSAC'92, July 1992, Berkeley, California.

[32] Wilf, H. S., and Zeilberger, D. An algorithmic proof theory for hypergeometric (ordinary and "q") multi-sum/integral identities. *Inventiones Mathematicae 108* (1992), 575–633.

[33] Zeilberger, D. A holonomic systems approach to special functions identities. *Journal of Computational and Applied Mathematics 32* (1990), 321–368.

[34] Zeilberger, D. A Maple program for proving hypergeometric identities. *SIGSAM Bulletin 25*, 3 (July 1991), 4–13.

[35] Zeilberger, D. The method of creative telescoping. *Journal of Symbolic Computation 11* (1991), 195–204.

INRIA-Rocquencourt and École Polytechnique
E-mail address: Frederic.Chyzak@inria.fr

INRIA-Rocquencourt
E-mail address: Bruno.Salvy@inria.fr