D-modules and Cohomology of Varieties

Uli Walther

Mathematical Sciences Research Institute, Berkeley, CA and
Department of Mathematics, Purdue University, West Lafayette, IN

Gröbner bases over polynomial rings have been used for many years in computational algebra, and the other chapters in this book bear witness to this fact. In the mid-eighties some important steps were made in the theory of Gröbner bases in non-commutative rings, notably in rings of differential operators. This chapter is about some of the applications of this theory to problems in commutative algebra and algebraic geometry.

Our interest in rings of differential operators and D-modules stems from the fact that some very interesting objects in algebraic geometry and commutative algebra have a *finite* module structure over an appropriate ring of differential operators. The prime example is the ring of regular functions on the complement of an affine hypersurface. A more general object is the Čech complex associated to a set of polynomials, and its cohomology, the local cohomology modules of the variety defined by the vanishing of the polynomials. More advanced topics are restriction functors and de Rham cohomology.

With these goals in mind, we shall study applications of Gröbner bases theory in the simplest ring of differential operators, the Weyl algebra, and develop algorithms that compute various invariants associated to a polynomial f. These include the Bernstein-Sato polynomial $b_f(s)$, the set of differential operators $J(f^s)$ which annihilate the germ of the function f^s (where s is a new variable), and the ring of regular functions on the complement of the variety of f.

For a family $f_1, \ldots, f_r$ of polynomials we study the associated Čech complex as a complex in the category of modules over the Weyl algebra. The algorithms are illustrated with examples. We also give an indication what other invariants associated to polynomials or varieties are known to be computable at this point and list some open problems in the area.

Acknowledgments

It is with great pleasure that I acknowledge the help of A. Leykin, M. Stillman and H. Tsai while writing this chapter. The D-module routines used or mentioned here have all been written by them and I would like to thank them for this marvelous job. I also would like to thank D. Grayson for help on *Macaulay 2* and D. Eisenbud and B. Sturmfels for inviting me to contribute to this volume.

1 Introduction

1.1 Local Cohomology – Definitions

Let R be a commutative Noetherian ring (always associative, with identity) and M an R-module. For $f \in R$ one defines a *Čech complex* of R-modules

$$\check{C}^\bullet(f) = (0 \to \underbrace{R}_{\text{degree } 0} \hookrightarrow \underbrace{R[f^{-1}]}_{\text{degree } 1} \to 0) \tag{1.1}$$

where the injection is the natural map sending $g \in R$ to $\frac{g}{1} \in R[f^{-1}]$ and "degree" refers to cohomological degree. For a family $f_1, \ldots, f_r \in R$ one defines

$$\check{C}^\bullet(f_1, \ldots, f_r) = \bigotimes_1^r \check{C}^\bullet(f_i), \tag{1.2}$$

and for an R-module M one sets

$$\check{C}^\bullet(M; f_1, \ldots, f_r) = M \otimes_R \check{C}^\bullet(f_1, \ldots, f_r). \tag{1.3}$$

The i-th (algebraic) *local cohomology functor* with respect to $f_1, \ldots, f_r$ is the i-th cohomology functor of $\check{C}^\bullet(-; f_1, \ldots, f_r)$. If $I = R \cdot (f_1, \ldots, f_r)$ then this functor agrees with the i-th right derived functor of the functor $H^0_I(-)$ which sends M to the I-torsion $\bigcup_{k=1}^\infty (0 :_M I^k)$ of M and is denoted by $H^i_I(-)$. This means in particular, that $H^\bullet_I(-)$ depends only on the (radical of the) ideal generated by the f_i. Local cohomology was introduced by A. Grothendieck [13] as an algebraic analog of (classical) relative cohomology. For instance, if X is a scheme, Y is a closed subscheme and $U = X \setminus Y$ then there is a long exact sequence

$$\cdots \to H^i(X, \mathfrak{F}) \to H^i(U, \mathfrak{F}) \to H^{i+1}_Y(X, \mathfrak{F}) \to \cdots$$

for all quasi-coherent sheaves $\mathfrak{F}$ on X. (To make sense of this one has to generalize the definition of local cohomology to be the right derived functor of $H^0_Y(-) : \mathfrak{F} \to (U \to \{f \in \mathfrak{F}(U) : \operatorname{supp}(f) \subseteq Y \cap U\})$.) An introduction to algebraic local cohomology theory may be found in [8].

The *cohomological dimension of I in R*, denoted by $\operatorname{cd}(R, I)$, is the smallest integer c such that $H^i_I(M) = 0$ for all $i > c$ and all R-modules M. If R is the coordinate ring of an affine variety X and $I \subseteq R$ is the defining ideal of the Zariski closed subset $Y \subseteq X$ then the *local cohomological dimension of Y in X* is defined as $\operatorname{cd}(R, I)$. It is not hard to show that if X is smooth, then the integer $\dim(X) - \operatorname{cd}(R, I)$ depends only on Y but neither on X nor on the embedding $Y \hookrightarrow X$.

1.2 Motivation

As one sees from the definition of local cohomology, the modules $H^i_I(R)$ carry information about the sections of the structure sheaf on Zariski open sets, and hence about the topology of these open sets. This is illustrated by the following examples. Let $I \subseteq R$ and $c = \mathrm{cd}(R, I)$. Then I cannot be generated by fewer than c elements – in other words, $\mathrm{Spec}(R) \setminus \mathrm{Var}(I)$ cannot be covered by fewer than c affine open subsets (i.e., $\mathrm{Var}(I)$ cannot be cut out by fewer than c hypersurfaces). In fact, no ideal J with the same radical as I will be generated by fewer than c elements, [8].

Let $H^i_{\mathrm{Sing}}(-;\mathbb{C})$ stand for the i-th singular cohomology functor with complex coefficients. The classical Lefschetz Theorem [12] states that if $X \subseteq \mathbb{P}^n_{\mathbb{C}}$ is a variety in projective n-space and Y a hyperplane section of X such that $X \setminus Y$ is smooth, then $H^i_{\mathrm{Sing}}(X;\mathbb{C}) \to H^i_{\mathrm{Sing}}(Y;\mathbb{C})$ is an isomorphism for $i < \dim(X) - 1$ and injective for $i = \dim(X) - 1$. The Lefschetz Theorem has generalizations in terms of local cohomology, called Theorems of Barth Type. For example, let $Y \subseteq \mathbb{P}^n_{\mathbb{C}}$ be Zariski closed and $I \subseteq R = \mathbb{C}[x_0, \ldots, x_n]$ the defining ideal of Y. Then $H^i_{\mathrm{Sing}}(\mathbb{P}^n_{\mathbb{C}};\mathbb{C}) \to H^i_{\mathrm{Sing}}(Y;\mathbb{C})$ is an isomorphism for $i < n - \mathrm{cd}(R, I)$ and injective if $i = n - \mathrm{cd}(R, I)$ ([15], Theorem III.7.1).

A consequence of the work of Ogus and Hartshorne ([37], 2.2, 2.3 and [15], Theorem IV.3.1) is the following. If $I \subseteq R = \mathbb{C}[x_0, \ldots, x_n]$ is the defining ideal of a complex smooth variety $Y \subseteq \mathbb{P}^n_{\mathbb{C}}$ then, for $i < n - \mathrm{codim}(Y)$,

$$\dim_{\mathbb{C}} \mathrm{soc}_R(H^0_{\mathfrak{m}}(H^{n-i}_I(R))) = \dim_{\mathbb{C}} H^i_x(\tilde{Y};\mathbb{C})$$

where $H^i_x(\tilde{Y};\mathbb{C})$ stands for the i-th singular cohomology group of the affine cone $\tilde{Y}$ over Y with support in the vertex x of $\tilde{Y}$ and with coefficients in $\mathbb{C}$ (and $\mathrm{soc}_R(M)$ denotes the socle $(0:_M (x_0, \ldots, x_n)) \subseteq M$ for any R-module M), [24]. These iterated local cohomology modules have a special structure (cf. Subsection 4.3).

Local cohomology relates to the connectedness of the underlying spaces as is shown by the following facts. If Y is a complete intersection of positive dimension in $\mathbb{P}^n_{\mathbb{C}}$, then Y cannot be disconnected by the removal of closed subsets of codimension 2 in Y or higher, [7]. This is a consequence of the so-called Hartshorne-Lichtenbaum vanishing theorem, see [8].

In a similar spirit one can show that if $(A, \mathfrak{m})$ is a complete local domain of dimension n and $f_1, \ldots, f_r$ are elements of the maximal ideal with $r + 2 \leq n$, then $\mathrm{Var}(f_1, \ldots, f_r) \setminus \{\mathfrak{m}\}$ is connected, [7].

In fact, as we will discuss to some extent in Section 5, over the complex numbers the complex $\check{C}^\bullet(R; f_1, \ldots, f_r)$ for $R = \mathbb{C}[x_1, \ldots, x_n]$ determines the Betti numbers $\dim_{\mathbb{C}}(H^i_{\mathrm{Sing}}(\mathbb{C}^n \setminus \mathrm{Var}(f_1, \ldots, f_r);\mathbb{C}))$.

1.3 The Masterplan

The cohomological dimension has been studied by many authors. For an extensive list of references and some open questions we recommend to consult the very nice survey article [17].

It turns out that for the determination of $\mathrm{cd}(R, I)$ it is in fact enough to find a test to decide whether or not the local cohomology module $H_I^i(R) = 0$ for given i, R, I. This is because $H_I^i(R) = 0$ for all $i > c$ implies $\mathrm{cd}(R, I) \leq c$ (see [14], Section 1).

Unfortunately, calculations are complicated by the fact that $H_I^i(M)$ is rarely finitely generated as R-module, even for very nice R and M. In this chapter we show how in an important class of examples one may still carry out explicit computations, by enlarging R.

We shall assume that $I \subseteq R_n = K[x_1, \ldots, x_n]$ where K is a computable field containing the rational numbers. (By a *computable field* we mean a subfield K of $\mathbb{C}$ such that K is described by a finite set of data and for which addition, subtraction, multiplication and division as well as the test whether the result of any of these operations is zero in the field can be executed by the Turing machine. For example, K could be $\mathbb{Q}[\sqrt{2}]$ stored as a 2-dimensional vector space over $\mathbb{Q}$ with an appropriate multiplication table.)

The ring of K-linear differential operators $D(R, K)$ of the commutative K-algebra R is defined inductively: one sets $D_0(R, K) = R$, and for $i > 0$ defines

$$D_i(R, K) = \{P \in \mathrm{Hom}_K(R, R) : Pr - rP \in D_{i-1}(R, K) \forall r \in R\}.$$

Here, $r \in R$ is interpreted as the endomorphism of R that multiplies by r.

The local cohomology modules $H_I^i(R_n)$ have a natural structure of finitely generated left $D(R_n, K)$-modules (see for example [20,24]). The basic reason for this finiteness is that in this case $R_n[f^{-1}]$ is a cyclic $D(R_n, K)$-module, generated by f^a for $\mathbb{Z} \ni a \ll 0$ (compare [5]):

$$R_n[f^{-1}] = D(R_n, K) \bullet f^a. \tag{1.4}$$

Using this finiteness we employ the theory of Gröbner bases in $D(R_n, K)$ to develop algorithms that give a presentation of $H_I^i(R_n)$ and $H_{\mathfrak{m}}^i(H_I^j(R_n))$ for all triples $i, j \in \mathbb{N}$, $I \subseteq R_n$ in terms of generators and relations over $D(R_n, K)$ (where $\mathfrak{m} = R_n \cdot (x_1, \ldots, x_n)$), see Section 4. This also leads to an algorithm for the computation of the invariants

$$\lambda_{i,j}(R_n/I) = \dim_K \mathrm{soc}_{R_n}(H_{\mathfrak{m}}^i(H_I^{n-j}(R_n)))$$

introduced in [24].

At the basis for the computation of local cohomology are algorithms that compute the localization of a $D(R_n, K)$-module at a hypersurface $f \in R_n$. That means, if the left module $M = D(R_n, K)^d/L$ is given by means of a finite number of generators for the left module $L \subseteq D(R_n, K)^d$ then we want to compute a finite number of generators for the left module $L' \subseteq D(R_n, K)^{d'}$ which satisfies

$$D(R_n, K)^{d'}/L' \cong (D(R_n, K)^d/L) \otimes_{R_n} R_n[f^{-1}],$$

which we do in Section 3.

Let L be a left ideal of $D(R_n, K)$. The computation of the localization of $M = D(R_n, K)/L$ at $f \in R_n$ is closely related to the $D(R_n, K)[s]$-module $\mathcal{M}_f$ generated by

$$\overline{1} \otimes 1 \otimes f^s \in M \otimes_{R_n} R_n[f^{-1}, s] \otimes f^s \tag{1.5}$$

and the minimal polynomial $b_f(s)$ of s on the quotient of $\mathcal{M}_f$ by its submodule $\mathcal{M}_f \cdot f$ generated over $D(R_n, K)[s]$ by $\overline{1} \otimes f \otimes f^s$, cf. Section 3. Algorithms for the computation of these objects have been established by T. Oaku in a sequence of papers [30–32].

Astonishingly, the roots of $b_f(s)$ prescribe the exponents a that can be used in the isomorphism (1.4) between $R_n[f^{-1}]$ and the $D(R_n, K)$-module generated by f^a. Moreover, any good exponent a can be used to transform $\mathcal{M}_f$ into $M \otimes R_n[f^{-1}]$ by a suitable "plugging in" procedure.

Thus the strategy for the computation of local cohomology will be to compute $\mathcal{M}_f$ and a good a for each $f \in \{f_1, \ldots, f_r\}$, and then assemble the Čech complex.

1.4 Outline of the Chapter

The next section is devoted to a short introduction of results on the Weyl algebra $D(R_n, K)$ and D-modules as they apply to our work. We start with some remarks on the theory of Gröbner bases in the Weyl algebra.

In Section 3 we investigate Bernstein-Sato polynomials, localizations and the Čech complex. The purpose of that section is to find a presentation of $M \otimes R_n[f^{-1}]$ as a cyclic $D(R_n, K)$-module if $M = D(R_n, K)/L$ is a given holonomic D-module (for a definition and some properties of holonomic modules, see Subsection 2.3 below).

In Section 4 we describe algorithms that for arbitrary i, j, k, I determine the structure of $H^k_I(R), H^i_{\mathfrak{m}}(H^j_I(R))$ and find $\lambda_{i,j}(R/I)$. The final section is devoted to comments on implementations, efficiency, discussions of other topics, and open problems.

2 The Weyl algebra and Gröbner Bases

D-modules, that is, rings or sheaves of differential operators and modules over these, have been around for several decades and played prominent roles in representation theory, some parts of analysis and in algebraic geometry. The founding fathers of the theory are M. Sato, M. Kashiwara, T. Kawai, J. Bernstein, and A. Beilinson. The area has also benefited much from the work of P. Deligne, J.-E. Björk, J.-E. Roos, B. Malgrange and Z. Mebkhout. The more computational aspects of the theory have been initiated by T. Oaku and N. Takayama.

The simplest example of a ring of differential operators is given by the Weyl algebra, the ring of K-linear differential operators on R_n. In characteristic zero, this is a finitely generated K-algebra that resembles the ring of polynomials in $2n$ variables but fails to be commutative.

2.1 Notation

Throughout we shall use the following notation: K will denote a computable field of characteristic zero and $R_n = K[x_1, \ldots, x_n]$ the ring of polynomials over K in n variables. The K-linear differential operators on R_n are then the elements of

$$D_n = K\langle x_1, \partial_1, \ldots, x_n, \partial_n \rangle,$$

the *n-th Weyl algebra* over K, where the symbol x_i denotes the operator "multiply by x_i" and ∂_i denotes the operator "take partial derivative with respect to x_i". We therefore have in D_n the relations

$$\begin{aligned} x_i x_j &= x_j x_i \quad \forall\, 1 \le i, j \le n, \\ \partial_i \partial_j &= \partial_j \partial_i \quad \forall\, 1 \le i, j \le n, \\ x_i \partial_j &= \partial_j x_i \quad \forall\, 1 \le i \ne j \le n, \\ \text{and } \; x_i \partial_i + 1 &= \partial_i x_i \quad \forall\, 1 \le i \le n. \end{aligned}$$

The last relation is nothing but the *product* (or *Leibniz*) *rule*, $xf' + f = (xf)'$. We shall use multi-index notation: $x^\alpha \partial^\beta$ denotes the monomial

$$x_1{}^{\alpha_1} \cdots x_n{}^{\alpha_n} \cdot \partial_1{}^{\beta_1} \cdots \partial_n{}^{\beta_n}$$

and $|\alpha| = \alpha_1 + \ldots + \alpha_n$.

In order to keep the product $\partial_i x_i \in D_n$ and the application of $\partial_i \in D_n$ to $x_i \in R_n$ apart, we shall write $\partial_i \bullet (g)$ to mean the result of the action of ∂_i on $g \in R_n$. So for example, $\partial_i x_i = x_i \partial_i + 1 \in D_n$ but $\partial_i \bullet x_i = 1 \in R_n$. The action of D_n on R_n takes precedence over the multiplication in R_n (and is of course compatible with the multiplication in D_n), so for example $\partial_2 \bullet (x_1) x_2 = 0 \cdot x_2 = 0 \in R_n$.

$\mathfrak{m}$ will stand for the maximal ideal $R_n \cdot (x_1, \ldots, x_n)$ of R_n, Δ will denote the maximal left ideal $D_n \cdot (\partial_1, \ldots, \partial_n)$ of D_n and I will stand for the ideal $R_n \cdot (f_1, \ldots, f_r)$ in R_n. Every D_n-module becomes an R_n-module via the embedding $R_n \hookrightarrow D_n$ as $D_0(R_n, K)$.

All tensor products in this chapter will be over R_n and all D_n-modules (resp. ideals) will be left modules (resp. left ideals) unless specified otherwise.

2.2 Gröbner Bases in D_n

This subsection is a severely shortened version of Chapter 1 in [39] (and we strongly recommend that the reader take a look at this book). The purpose is to see how Gröbner basis theory applies to the Weyl algebra.

The elements in D_n allow a *normally ordered expression.* Namely, if $P \in D_n$ then we can write it as

$$P = \sum_{(\alpha,\beta)\in E} c_{\alpha,\beta} x^\alpha \partial^\beta$$

where E is a finite subset of $\mathbb{N}^{2n}$. Thus, as K-vector spaces there is an isomorphism

$$\Psi : K[x, \xi] \to D_n$$

(with $\xi = \xi_1, \ldots, \xi_n$) sending $x^\alpha \xi^\beta$ to $x^\alpha \partial^\beta$. We will assume that every $P \in D_n$ is normally ordered.

We shall say that $(u, v) \in \mathbb{R}^{2n}$ is a *weight vector for* D_n if $u + v \geq 0$, that is $u_i + v_i \geq 0$ for all $1 \leq i \leq n$. We set the *weight* of the monomial $x^\alpha \partial^\beta$ under (u, v) to be $u \cdot \alpha + v \cdot \beta$ (scalar product). The weight of an operator is then the maximum of the weights of the nonzero monomials appearing in the normally ordered expression of P. If (u, v) is a weight vector for D_n, there is an associated graded ring $\mathrm{gr}_{(u,v)}(D_n)$ with

$$\mathrm{gr}^r_{(u,v)}(D_n) = \frac{\{P \in D_n : w(P) \leq r\}}{\{P \in D_n : w(P) < r\}}.$$

So $\mathrm{gr}_{(u,v)}(D_n)$ is the K-algebra on the symbols $\{x_i : 1 \leq i \leq n\} \cup \{\partial_i : u_i + v_i = 0\} \cup \{\xi_i : u_i + v_i > 0\}$. Here all variables commute with each other except ∂_i and x_i for which the Leibniz rule holds.

Each $P \in D_n$ has an *initial form* or *symbol* $\mathrm{in}_{(u,v)}(P)$ in $\mathrm{gr}_{(u,v)}(D_n)$ defined by taking all monomials in the normally ordered expression for P that have maximal weight, and replacing all ∂_i with $u_i + v_i > 0$ by the corresponding ξ_i.

The inequality $u_i + v_i \geq 0$ is needed to assure that the product of the initial forms of two operators equals the initial form of their product: one would not want to have $\mathrm{in}(\partial_i \cdot x_i) = \mathrm{in}(x_i \cdot \partial_i + 1) = 1$.

A weight of particular importance is $-u = v = (1, \ldots, 1)$, or more generally $-u = v = (1, \ldots, 1, 0, \ldots, 0)$. In these cases $\mathrm{gr}_{(u,v)}(D_n) \cong D_n$. On the other hand, if $u + v$ is componentwise positive, then $\mathrm{gr}_{(u,v)}(D_n)$ is commutative (compare the initial forms of $\partial_i x_i$ and $x_i \partial_i$) and isomorphic to the polynomial ring in $2n$ variables corresponding to the symbols of $x_1, \ldots, x_n, \partial_1, \ldots, \partial_n$.

If L is a left ideal in D_n we write $\mathrm{in}_{(u,v)}(L)$ for $\{\mathrm{in}_{(u,v)}(P) : P \in L\}$. This is a left ideal in $\mathrm{gr}_{(u,v)}(D_n)$. If $G \subset L$ is a finite set we call it a (u, v)-*Gröbner basis* if the left ideal of $\mathrm{gr}_{(u,v)}(D_n)$ generated by the initial forms of the elements of G agrees with $\mathrm{in}_{(u,v)}(L)$.

A *multiplicative monomial order on* D_n is a total order $\prec$ on the normally ordered monomials such that

1. $1 \prec x_i \partial_i$ for all i, and
2. $x^\alpha \partial^\beta \prec x^{\alpha'} \partial^{\beta'}$ implies $x^{\alpha+\alpha''} \partial^{\beta+\beta''} \prec x^{\alpha'+\alpha''} \partial^{\beta'+\beta''}$ for all $\alpha'', \beta'' \in \mathbb{N}^n$.

A multiplicative monomial order is a *term order* if 1 is the (unique) smallest monomial. Multiplicative monomial orders, and more specifically term orders, clearly abound.

Multiplicative monomial orders (and hence term orders) allow the construction of initial forms just like weight vectors. Now, however, the initial forms are always monomials, and always elements of $K[x,\xi]$ (due to the total order requirement on $\prec$). One defines Gröbner bases for multiplicative monomial orders analogously to the weight vector case.

For our algorithms we have need to compute weight vector Gröbner bases, and this can be done as follows. Suppose (u,v) is a weight vector on D_n and $\prec$ a term order. Define a multiplicative monomial order $\prec_{(u,v)}$ as follows:

$$x^\alpha\partial^\beta \prec_{(u,v)} x^{\alpha'}\partial^{\beta'} \Leftrightarrow [(\alpha-\alpha')\cdot u+(\beta-\beta')\cdot v<0] \text{ or } \left[(\alpha-\alpha')\cdot u+(\beta-\beta')\cdot v=0 \text{ and } x^\alpha\partial^\beta \prec x^{\alpha'}\partial^{\beta'}\right].$$

Note that $\prec_{(u,v)}$ is a term order precisely when (u,v) is componentwise non-negative.

Theorem 1 ([39], Theorem 1.1.6.). *Let L be a left ideal in D_n, (u,v) a weight vector for D_n, $\prec$ a term order and G a Gröbner basis for L with respect to $\prec_{(u,v)}$. Then*

1. *G is a Gröbner basis for L with respect to (u,v).*
2. *$\mathrm{in}_{(u,v)}(G)$ is a Gröbner basis for $\mathrm{in}_{(u,v)}(L)$ with respect to $\prec$.* □

We end this subsection with the remarks that Gröbner bases with respect to multiplicative monomial orders can be computed using the Buchberger algorithm adapted to the non-commutative situation (thus, Gröbner bases with respect to weight vectors are computable according to the theorem), and that the computation of syzygies, kernels, intersections and preimages in D_n works essentially as in the commutative algebra $K[x,\xi]$. For precise statements of the algorithms we refer the reader to [39].

2.3 *D*-modules

A good introduction to D-modules are the book by J.-E. Björk, [5], the nice introduction [9] by S. Coutinho, and the lecture notes by J. Bernstein [4]. In this subsection we list some properties of localizations of R_n that are important for module-finiteness over D_n. Most of this section is taken from Section 1 in [5].

Let $f \in R_n$. Then the R_n-module $R_n[f^{-1}]$ has a structure as left D_n-module via the extension of the action $\bullet$:

$$x_i \bullet (\frac{g}{f^k}) = \frac{x_i g}{f^k}, \qquad \partial_i \bullet (\frac{g}{f^k}) = \frac{\partial_i \bullet (g) f - k\partial_i \bullet (f) g}{f^{k+1}}.$$

This may be thought of as a special case of localizing a D_n-module: if M is a D_n-module and $f \in R_n$ then $M \otimes_{R_n} R_n[f^{-1}]$ becomes a D_n-module via the *product rule*

$$x_i \bullet (m \otimes \frac{g}{f^k}) = m \otimes (\frac{x_i g}{f^k}), \quad \partial_i \bullet (m \otimes \frac{g}{f^k}) = m \otimes \partial_i \bullet (\frac{g}{f^k}) + \partial_i m \otimes \frac{g}{f^k}.$$

Of particular interest are the *holonomic* modules which are those finitely generated D_n-modules M for which $\operatorname{Ext}^j_{D_n}(M, D_n)$ vanishes unless $j = n$. This innocent looking definition has surprising consequences, some of which we discuss now.

The holonomic modules form a full Abelian subcategory of the category of left D_n-modules, closed under the formation of subquotients. Our standard example of a holonomic module is

$$R_n = D_n/\Delta.$$

This equality may require some thought – it pictures R_n as a D_n-module generated by $1 \in R_n$. It is particularly noteworthy that not all elements of R_n are killed by Δ – quite impossible if D_n were commutative.

Holonomic modules are always cyclic and of finite length over D_n. These fundamental properties are consequences of the *Bernstein inequality.* To understand this inequality we associate with the D_n-module $M = D_n/L$ the Hilbert function $q_L(k)$ with values in the integers which counts for each $k \in \mathbb{N}$ the number of monomials $x^\alpha \partial^\beta$ with $|\alpha| + |\beta| \leq k$ whose cosets in M are K-linearly independent. The filtration $k \mapsto K \cdot \{x^\alpha \partial^\beta \mod L : |\alpha| + |\beta| \leq k\}$ is called the *Bernstein filtration.* The Bernstein inequality states that $q_L(k)$ is either identically zero (in which case $M = 0$) or asymptotically a polynomial in k of degree between n and $2n$. This degree is called the *dimension of* M. A holonomic module is one of dimension n, the minimal possible value for a nonzero module.

This characterization of holonomicity can be used quite easily to check with *Macaulay 2* that R_n is holonomic. Namely, let's say $n = 3$. Start a *Macaulay 2* session with

```
i1 :  load "Dloadfile.m2"

i2 : D = QQ[x,y,z,Dx,Dy,Dz, WeylAlgebra => {x=>Dx, y=>Dy, z=>Dz}]

o2 = D

o2 : PolynomialRing

i3 : Delta = ideal(Dx,Dy,Dz)

o3 = ideal (Dx, Dy, Dz)

o3 : Ideal of D
```

The first of these commands loads the D-module library by A. Leykin, M. Stillman and H. Tsai, [23]. The second line defines the base ring $D_3 = \mathbb{Q}\langle x, y, z, \partial_x, \partial_y, \partial_z\rangle$, while the third command defines the D_3-module $D_3/\Delta \cong R_3$.

As one can see, *Macaulay 2* thinks of `D` as a ring of polynomials. This is using the vector space isomorphism Ψ from Subsection 2.2. Of course, two elements are multiplied according to the Leibniz rule. All Weyl algebra ideals and modules are by default left ideals and left modules in *Macaulay 2*.

If we don't specify any further what order is to be used in the Weyl algebra, then *Macaulay 2* automatically picks a version of `grlex` (to see the relevant part of the output, one would scroll to the right on a computer):

```
i4 : sendgg ggPush D

o4 =

o4 : String

i5 : look

WeylAlgebra(Frac(ZZ)[x,y,z,Dx,Dy,Dz; Degrees => {1, 1, 1, 1, 1, 1}; Mo ···
```

These two commands allow to gather some extra information on an object.

To compute the initial ideal of Δ with respect to the weight that associates 1 to each ∂ and to each variable, execute

```
i6 : DeltaBern = inw(Delta,{1,1,1,1,1,1})

o6 = ideal (Dz, Dy, Dx)

o6 : Ideal of QQ [x, y, z, Dx, Dy, Dz]
```

The command `inw` can be used with any weight vector for D_n as second argument. One notes that the output is not an ideal in a Weyl algebra any more, but in a ring of polynomials, as it should. The dimension of R_3, which is the dimension of the variety associated to `DeltaBern`, is computed by

```
i7 : dim DeltaBern

o7 = 3
```

As this is equal to $n = 3$, the ideal Δ is holonomic.

Let $R_n[f^{-1}, s] \otimes f^s$ be the free $R_n[f^{-1}, s]$-module generated by the symbol f^s. Using the action $\bullet$ of D_n on $R_n[f^{-1}, s]$ we define an action $\bullet$ of $D_n[s]$ on $R_n[f^{-1}, s] \otimes f^s$ by setting

$$
\begin{aligned}
s \bullet \left(\frac{g(x,s)}{f^k} \otimes f^s\right) &= \frac{sg(x,s)}{f^k} \otimes f^s, \\
x_i \bullet \left(\frac{g(x,s)}{f^k} \otimes f^s\right) &= \frac{x_i g(x,s)}{f^k} \otimes f^s, \\
\partial_i \bullet \left(\frac{g(x,s)}{f^k} \otimes f^s\right) &= \left(\partial_i \bullet \left(\frac{g(x,s)}{f^k}\right) + s\partial_i \bullet (f) \cdot \frac{g(x,s)}{f^{k+1}}\right) \otimes f^s.
\end{aligned}
$$

The last rule justifies the choice for the symbol of the generator.

Writing $M = D_n/L$ and denoting by $\overline{1}$ the coset of $1 \in D_n$ in M, this action extends to an action of $D_n[s]$ on

$$\mathcal{M}_f^L = D_n[s] \bullet (\overline{1} \otimes 1 \otimes f^s) \subseteq M \otimes_{R_n} \left(R_n[f^{-1}, s] \otimes f^s\right) \tag{2.1}$$

by the product rule for all left D_n-modules M. The interesting bit about $\mathcal{M}_f^L$ is the following fact. If $M = D_n/L$ is holonomic then there is a nonzero polynomial $b(s)$ in $K[s]$ and an operator $P(s) \in D_n[s]$ such that

$$P(s) \bullet (\overline{1} \otimes f \otimes f^s) = \overline{1} \otimes b(s) \otimes f^s \tag{2.2}$$

in $\mathcal{M}_f^L$. This entertaining equality, often written as

$$P(s)\left(\overline{1} \otimes f^{s+1}\right) = \overline{b(s)} \otimes f^s,$$

says that $P(s)$ is roughly equal to division by f. The unique monic polynomial that divides all other polynomials $b(s)$ satisfying an identity of this type is called the *Bernstein* (or also *Bernstein-Sato*) *polynomial* of L and f and denoted by $b_f^L(s)$. Any operator $P(s)$ that satisfies (2.2) with $b(s) = b_f(s)$ we shall call a *Bernstein operator* and refer to the roots of $b_f^L(s)$ as *Bernstein roots* of f on D_n/L. It is clear from (2.2) and the definitions that $b_f^L(s)$ is the minimal polynomial of s on the quotient of $\mathcal{M}_f^L$ by $D_n[s] \bullet (\overline{1} \otimes f \otimes f^s)$.

The Bernstein roots of the polynomial f are somewhat mysterious, but related to other algebro-geometric invariants as, for example, the monodromy of f (see [28]) and the log-canonical threshold (see [21]). For a long time it was also unclear how to compute $b_f(s)$ for given f. In [52] many interesting examples of Bernstein-Sato polynomials are worked out by hand, while in [1,6,27,40] algorithms were given that compute $b_f(s)$ under certain conditions on f. The general algorithm we are going to explain was given by T. Oaku. Here is a classical example.

Example 2. Let $f = \sum_{i=1}^n {x_i}^2$ and $M = R_n$ with $L = \Delta$. One can check that

$$\sum_{i=1}^n {\partial_i}^2 \bullet (\overline{1} \otimes 1 \otimes f^{s+1}) = \overline{1} \otimes 4(s+1)(\frac{n}{2} + s) \otimes f^s$$

and hence that $\frac{1}{4}\sum_{i=1}^n {\partial_i}^2$ is a Bernstein operator while the Bernstein roots of f are -1 and $-\frac{n}{2}$ and the Bernstein polynomial is $(s+1)(s+\frac{n}{2})$.

Example 3. Although in the previous example the Bernstein operator looked a lot like the polynomial f, this is not often the case and it is usually hard to guess Bernstein operators. For example, one has

$$\left(\frac{1}{27}\,{\partial_y}^3 + \frac{y}{6}{\partial_x}^2\partial_y + \frac{x}{8}{\partial_x}^3\right)(x^2+y^3)^{s+1} = (s+\frac{5}{6})(s+1)(s+\frac{7}{6})(x^2+y^3)^s.$$

In the case of non-quasi-homogeneous polynomials, there is usually no resemblance between f and any Bernstein operator.

A very important property of holonomic modules is the (somewhat counter-intuitive) fact that any localization of a holonomic module $M = D_n/L$ at a single element (and hence at any finite number of elements) of R_n is holonomic ([5], 1.5.9) and in particular cyclic over D_n, generated by $\overline{1} \otimes f^a$ for sufficiently small $a \in \mathbb{Z}$. As a special case we note that localizations of R_n are holonomic, and hence finitely generated over D_n. Coming back to the Čech complex we see that the complex $\check{C}^\bullet(M; f_1, \ldots, f_r)$ consists of holonomic D_n-modules whenever M is holonomic.

As a consequence, local cohomology modules of R_n are D_n-modules and in fact holonomic. To see this it suffices to know that the maps in the Čech complex are D_n-linear, which we will explain in Section 4. Since the category of holonomic D_n-modules and their D_n-linear maps is closed under subquotients, holonomicity of $H^k_I(R_n)$ follows.

For similar reasons, $H^i_{\mathfrak{m}}(H^j_I(R_n))$ is holonomic for $i, j \in \mathbb{N}$ (since $H^j_I(R_n)$ is holonomic). These modules, investigated in Subsections 4.2 and 4.3, are rather special R_n-modules and seem to carry some very interesting information about $\mathrm{Var}(I)$, see [10,51].

The fact that R_n is holonomic and every localization of a holonomic module is as well, provides motivation for us to study this class of modules. There are, however, more occasions where holonomic modules show up. One such situation arises in the study of linear partial differential equations. More specifically, the so-called GKZ-systems (which we will meet again in the final chapter) provide a very interesting class of objects with fascinating combinatorial and analytic properties [39].

3 Bernstein-Sato Polynomials and Localization

We mentioned in the introduction that for the computation of local cohomology the following is an important algorithmic problem to solve.

Problem 4. Given $f \in R_n$ and a left ideal $L \subseteq D_n$ such that $M = D_n/L$ is holonomic, compute the structure of the module $D_n/L \otimes R_n[f^{-1}]$ in terms of generators and relations.

This section is about solving Problem 4.

3.1 The Line of Attack

Recall for a given D_n-module $M = D_n/L$ the action of $D_n[s]$ on the tensor product $M \otimes_{R_n} (R_n[f^{-1}, s] \otimes f^s)$ from Subsection 2.3. We begin with defining an ideal of operators:

Definition 5. Let $J^L(f^s)$ stand for the ideal in $D_n[s]$ that kills $\overline{1} \otimes 1 \otimes f^s \in (D_n/L) \otimes_{R_n} R_n[f^{-1}, s] \otimes f^s$.

It turns out that it is very useful to know this ideal. If $L = \Delta$ then there are some obvious candidates for generators of $J^L(f^s)$. For example, there are $f\partial_i - \partial_i \bullet (f)s$ for all i. However, unless f is smooth, these will not generate $J^\Delta(f^s)$. For a more general L, there is a similar set of (somewhat less) obvious candidates, but again finding all elements of $J^\Delta(f^s)$ is far from elementary, even for smooth f.

In order to find $J^L(f^s)$, we will consider the module $(D_n/L)\otimes R_n[f^{-1}, s]\otimes f^s$ over the ring $D_{n+1} = D_n\langle t, \partial_t\rangle$ by defining an appropriate action of t and ∂_t on it. It is then not hard to compute the ideal $J^L_{n+1}(f^s) \subseteq D_{n+1}$ consisting of all operators that kill $\overline{1} \otimes 1 \otimes f^s$, see Lemma 8. In Proposition 9 we will then explain how to compute $J^L(f^s)$ from $J^L_{n+1}(f^s)$.

This construction gives an answer to the question of determining a presentation of $D_n \bullet (\overline{1} \otimes f^a)$ for "most" $a \in K$, which we make precise as follows.

Definition 6. We say that a property depending on $a \in K^m$ *holds for a in very general position*, if there is a countable set of hypersurfaces in K^m such that the property holds for all a not on any of the exceptional hypersurfaces.

It will turn out that for $a \in K$ in very general position $J^L(f^s)$ "is" the annihilator for f^a: we shall very explicitly identify a countable number of exceptional values in K such that if a is not equal to one of them, then $J^L(f^s)$ evaluates under $s \mapsto a$ to the annihilator inside D_n of $\overline{1} \otimes f^a$.

For $a \in \mathbb{Z}$ we have of course $D_n \bullet (\overline{1} \otimes f^a) \subseteq M \otimes R_n[f^{-1}]$ but the inclusion may be strict (e.g., for $L = \Delta$ and $a = 0$). Proposition 14 shows how $(D_n/L) \otimes R_n[f^{-1}]$ and $J^L(f^s)$ are related.

3.2 Undetermined Exponents

Consider $D_{n+1} = D_n\langle t, \partial_t\rangle$, the Weyl algebra in $x_1, \ldots, x_n$ and the new variable t. B. Malgrange [28] has defined an action $\bullet$ of D_{n+1} on $(D_n/L) \otimes R_n[f^{-1}, s]\otimes f^s$ as follows. We require that x_i acts as multiplication on the first factor, and for the other variables we set (with $\overline{P} \in D_n/L$ and $g(x, s) \in R_n[s]$)

$$\begin{aligned}
\partial_i \bullet (\overline{P} \otimes \frac{g(x,s)}{f^k} \otimes f^s) &= \left(\overline{P} \otimes \left(\partial_i \bullet (\frac{g(x,s)}{f^k}) + \frac{s\partial_i \bullet (f) g(x,s)}{f^{k+1}}\right) + \overline{\partial_i P} \otimes \frac{g(x,s)}{f^k}\right) \otimes f^s, \\
t \bullet (\overline{P} \otimes \frac{g(x,s)}{f^k} \otimes f^s) &= \overline{P} \otimes \frac{g(x,s+1)f}{f^k} \otimes f^s, \\
\partial_t \bullet (\overline{P} \otimes \frac{g(x,s)}{f^k} \otimes f^s) &= \overline{P} \otimes \frac{-sg(x,s-1)}{f^{k+1}} \otimes f^s.
\end{aligned}$$

One checks that this actually defines a left D_{n+1}-module structure (i.e., $\partial_t t$ acts like $t\partial_t + 1$) and that $-\partial_t t$ acts as multiplication by s.

Definition 7. We denote by $J^L_{n+1}(f^s)$ the ideal in D_{n+1} that annihilates the element $\overline{1} \otimes 1 \otimes f^s$ in $(D_n/L) \otimes R_n[f^{-1}, s] \otimes f^s$ with D_{n+1} acting as defined above. Then we have an induced morphism of D_{n+1}-modules $D_{n+1}/J^L_{n+1}(f^s) \to (D_n/L) \otimes R_n[f^{-1}, s] \otimes f^s$ sending $P + J^L_{n+1}(f^s)$ to $P \bullet (\overline{1} \otimes 1 \otimes f^s)$.

We say that an ideal $L \subseteq D_n$ is *f-saturated* if $f \cdot P \in L$ implies $P \in L$ and we say that D_n/L is *f-torsion free* if L is f-saturated. R_n and all its localizations are examples of f-torsion free modules for arbitrary f.

The following lemma is a modification of Lemma 4.1 in [28] where the special case $L = D_n \cdot (\partial_1, \ldots, \partial_n), D_n/L = R_n$ is considered (compare also [46]).

Lemma 8. *Suppose that $L = D_n \cdot (P_1, \ldots, P_r)$ is f-saturated. With the above definitions, $J^L_{n+1}(f^s)$ is the ideal generated by $f - t$ together with the images of the P_j under the automorphism ϕ of D_{n+1} induced by $x_i \mapsto x_i \forall i$, $t \mapsto t - f$.*

Proof. The automorphism sends ∂_i to $\partial_i + \partial_i \bullet (f)\partial_t$ and ∂_t to ∂_t. So if we write P_j as a polynomial $P_j(\partial_1, \ldots, \partial_n)$ in the ∂_i with coefficients in $K[x_1, \ldots, x_n]$, then

$$\phi(P_j) = P_j(\partial_1 + \partial_1 \bullet (f)\partial_t, \ldots, \partial_n + \partial_n \bullet (f)\partial_t).$$

One checks that $(\partial_i + \partial_i \bullet (f)\partial_t) \bullet (\overline{Q} \otimes 1 \otimes f^s) = \overline{\partial_i Q} \otimes 1 \otimes f^s$ for all $Q \in D_{n+1}$, so that $\phi(P_j(\partial_1, \ldots, \partial_n)) \bullet (\overline{1} \otimes 1 \otimes f^s) = \overline{P_j(\partial_1, \ldots, \partial_n)} \otimes 1 \otimes f^s = 0$. By definition, $f \bullet (\overline{1} \otimes 1 \otimes f^s) = t \bullet (\overline{1} \otimes 1 \otimes f^s)$. So $t - f \in J^L_{n+1}(f^s)$ and $\phi(P_j) \in J^L_{n+1}(f^s)$ for $j = 1, \ldots, r$.

Conversely let $P \bullet (\overline{1} \otimes 1 \otimes f^s) = 0$. The proof that $P \in \phi(J^L_{n+1} + D_{n+1} \cdot t)$ relies on an elimination idea and has some Gröbner basis flavor. We have to show that $P \in D_{n+1} \cdot (\phi(P_1), \ldots, \phi(P_r), t - f)$. We may assume, that P does not contain any power of t since we can eliminate t using $f - t$. Now rewrite P in terms of ∂_t and the $\partial_i + \partial_i \bullet (f)\partial_t$. Say, $P = \sum_{\alpha,\beta} \partial_t^\alpha x^\beta Q_{\alpha,\beta}(\partial_1 + \partial_1 \bullet (f)\partial_t, \ldots, \partial_n + \partial_n \bullet (f)\partial_t)$, where the $Q_{\alpha,\beta} \in K[y_1, \ldots, y_n]$ are polynomial expressions. Then

$$P \bullet (\overline{1} \otimes 1 \otimes f^s) = \sum_{\alpha,\beta} \partial_t^\alpha \bullet (\overline{x^\beta Q_{\alpha,\beta}(\partial_1, \ldots, \partial_n)} \otimes 1 \otimes f^s).$$

Let $\overline{\alpha}$ be the largest $\alpha \in \mathbb{N}$ for which there is a nonzero $Q_{\alpha,\beta}$ occurring in $P = \sum_{\alpha,\beta} \partial_t^\alpha x^\beta Q_{\alpha,\beta}(\partial_1 + \partial_1 \bullet (f)\partial_t, \ldots, \partial_n + \partial_n \bullet (f)\partial_t)$. We show that the sum of terms that contain $\partial_t^{\overline{\alpha}}$ is in $D_{n+1} \cdot \phi(L)$ as follows. In order for $P \bullet (\overline{1} \otimes 1 \otimes f^s)$ to vanish, the sum of terms with the highest s-power, namely $s^{\overline{\alpha}}$, must vanish. Hence $\sum_\beta x^\beta Q_{\overline{\alpha},\beta}(\partial_1, \ldots, \partial_n) \otimes (-1/f)^{\overline{\alpha}} \otimes f^s \in L \otimes R_n[f^{-1}, s] \otimes f^s$ as $R_n[f^{-1}, s]$ is $R_n[s]$-flat. It follows, that $\sum_\beta x^\beta Q_{\overline{\alpha},\beta}(\partial_1, \ldots, \partial_n) \in L$ (L is f-saturated!) and hence $\sum_\beta \partial_t^{\overline{\alpha}} x^\beta Q_{\overline{\alpha},\beta}(\partial_1 + \partial_1 \bullet (f)\partial_t, \ldots, \partial_n + \partial_n \bullet (f)\partial_t) \in D_{n+1} \cdot \phi(L)$ as announced.

So by the first part, $P - \sum_\beta \partial_t^{\overline{\alpha}} x^\beta Q_{\overline{\alpha},\beta}(\partial_1 + \partial_1 \bullet (f)\partial_t, \ldots, \partial_n + \partial_n \bullet (f)\partial_t)$ kills $\overline{1} \otimes 1 \otimes f^s$, but is of smaller degree in ∂_t than P was.

The claim follows by induction on $\overline{\alpha}$. ∎

If we identify $D_n[-\partial_t t]$ with $D_n[s]$ then $J^L_{n+1}(f^s) \cap D_n[-\partial_t t]$ is identified with $J^L(f^s)$ since, as we observed earlier, $-\partial_t t$ multiplies by s on $\mathcal{M}^L_f$. As we pointed out in the beginning, the crux of our algorithms is to calculate $J^L(f^s) = J^L_{n+1}(f^s) \cap D_n[s]$. We shall deal with this computation now.

In Theorem 19 of [32], T. Oaku showed how to construct a generating set for $J^L(f^s)$ in the case $L = D_n \cdot (\partial_1, \dots, \partial_n)$. Using his ideas we explain how one may calculate $J \cap D_n[-\partial_t t]$ whenever $J \subseteq D_{n+1}$ is any given ideal, and as a corollary develop an algorithm that for f-saturated D_n/L computes $J^L(f^s) = J^L_{n+1}(f^s) \cap D_n[-\partial_t t]$.

We first review some work of Oaku. On D_{n+1} we define the weight vector w by $w(t) = 1, w(\partial_t) = -1, w(x_i) = w(\partial_i) = 0$ and we extend it to $D_{n+1}[y_1, y_2]$ by $w(y_1) = -w(y_2) = 1$. If $P = \sum_i P_i \in D_{n+1}[y_1, y_2]$ and all P_i are monomials, then we will write $(P)^h$ for the operator $\sum_i P_i \cdot y_1^{d_i}$ where $d_i = \max_j(w(P_j)) - w(P_i)$ and call it the y_1-*homogenization* of P.

Note that the Buchberger algorithm preserves homogeneity in the following sense: if a set of generators for an ideal is given and these generators are homogeneous with respect to the weight above, then any new generator for the ideal constructed with the classical Buchberger algorithm will also be homogeneous. (This is a consequence of the facts that the y_i commute with all other variables and that $\partial_t t = t\partial_t + 1$ is homogeneous of weight zero.) This homogeneity is very important for the following result of Oaku:

Proposition 9. *Let $J = D_{n+1} \cdot (Q_1, \dots, Q_r)$. Let I be the left ideal in $D_{n+1}[y_1]$ generated by the y_1-homogenizations $(Q_i)^h$ of the Q_i, relative to the weight w above, and set $\tilde{I} = D_{n+1}[y_1, y_2] \cdot (I, 1 - y_1 y_2)$. Let G be a Gröbner basis for $\tilde{I}$ under a monomial order that eliminates y_1, y_2. For each $P \in G \cap D_{n+1}$ set $P' = t^{-w(P)}P$ if $w(P) < 0$ and $P' = \partial_t^{w(P)} P$ if $w(P) \geq 0$. Set $G_0 = \{P' : P \in G \cap D_{n+1}\}$. Then $G_0 \subseteq D_n[-\partial_t t]$ generates $J \cap D_n[-\partial_t t]$.*

Proof. This is in essence Theorem 18 of [32]. (See the remarks in Subsection 2.2 on how to compute such Gröbner bases.) □

As a corollary to this proposition we obtain an algorithm for the computation of $J^{\Delta}(f^s)$:

Algorithm 10 (Parametric Annihilator).
Input: $f \in R_n$; $L \subseteq D_n$ such that L is f-saturated.
Output: Generators for $J^L(f^s)$.

1. For each generator Q_i of $D_{n+1} \cdot (L, t)$ compute the image $\phi(Q_i)$ under $x_i \mapsto x_i$, $t \mapsto t - f$, $\partial_i \mapsto \partial_i + \partial_i \bullet (f)\partial_t$, $\partial_t \mapsto \partial_t$.
2. Homogenize all $\phi(Q_i)$ with respect to the new variable y_1 relative to the weight w introduced before Proposition 9.
3. Compute a Gröbner basis for the ideal

$$D_{n+1}[y_1, y_2] \cdot ((\phi(Q_1))^h, \dots, (\phi(Q_r))^h, 1 - y_1 y_2)$$

in $D_{n+1}[y_1, y_2]$ using an order that eliminates y_1, y_2.

4. Select the operators $\{P_j\}_1^b$ in this basis which do not contain y_1, y_2.
5. For each P_j, $1 \leq j \leq b$, if $w(P_j) > 0$ replace P_j by $P'_j = \partial_t^{w(P_j)} P_j$. Otherwise replace P_j by $P'_j = t^{-w(P_j)} P_j$.
6. Return the new operators $\{P'_j\}_1^b$.

End.

The output will be operators in $D_n[-\partial_t t]$ which is naturally identified with $D_n[s]$ (including the action on $\mathcal{M}_f^L$). This algorithm is in effect Proposition 7.1 of [31].

In *Macaulay 2*, one can compute the parametric annihilator ideal (for $R_n = \Delta$) by the command `AnnFs`:

```
i8 : D = QQ[x,y,z,w,Dx,Dy,Dz,Dw,
           WeylAlgebra => {x=>Dx, y=>Dy, z=>Dz, w=>Dw}];

i9 : f = x^2+y^2+z^2+w^2

      2    2    2    2
o9 = x  + y  + z  + w

o9 : D

i10 : AnnFs(f)

                                                                        ...
o10 = ideal (w*Dz - z*Dw, w*Dy - y*Dw, z*Dy - y*Dz, w*Dx - x*Dw, z*Dx   ...
                                                                        ...

o10 : Ideal of QQ [x, y, z, w, Dx, Dy, Dz, Dw, $s, WeylAlgebra => {x = ...
```

If we want to compute $J^L(f^s)$ for more general L, we have to use the command `AnnIFs`:

```
i11 : L=ideal(x,y,Dz,Dw)

o11 = ideal (x, y, Dz, Dw)

o11 : Ideal of D

i12 : AnnIFs(L,f)

                                      1          1
o12 = ideal (y, x, w*Dz - z*Dw, -*z*Dz + -*w*Dw - $s)
                                      2          2

o12 : Ideal of QQ [x, y, z, w, Dx, Dy, Dz, Dw, $s, WeylAlgebra => {x = ...
```

It should be emphasized that saturatedness of L with respect to f is a must for `AnnIFs`.

3.3 The Bernstein-Sato polynomial

Knowing $J^L(f^s)$ allows us to get our hands on the Bernstein-Sato polynomial of f on M:

Corollary 11. *Suppose L is a holonomic ideal in D_n (i.e., D_n/L is holonomic). The Bernstein polynomial $b_f^L(s)$ of f on (D_n/L) satisfies*

$$(b_f^L(s)) = \big(D_n[s] \cdot (J^L(f^s), f)\big) \cap K[s]. \tag{3.1}$$

Moreover, if L is f-saturated then $b_f^L(s)$ can be computed with Gröbner basis computations.

Proof. By definition of $b_f^L(s)$ we have $(b_f^L(s) - P_f^L(s) \cdot f) \bullet (\overline{1} \otimes 1 \otimes f^s) = 0$ for a suitable $P_f^L(s) \in D_n[s]$. Hence $b_f^L(s)$ is in $K[s]$ and in $D_n[s](J^L(f^s), f)$. Conversely, if $b(s)$ is in this intersection then $b(s)$ satisfies an equality of the type of (2.2) and hence is a multiple of $b_f^L(s)$.

If we use an elimination order for which $\{x_i, \partial_i\}_1^n \gg s$ in $D_n[s]$, then if $J^L(f^s)$ is known, $b_f^L(s)$ will be (up to a scalar factor) the unique element in the reduced Gröbner basis for $D_n[s] \cdot (J^L(f^s), f)$ that contains no x_i nor ∂_i. Since we assume L to be f-saturated, $J^L(f^s)$ can be computed according to Proposition 9. □

We therefore arrive at the following algorithm for the Bernstein-Sato polynomial [30].

Algorithm 12 (Bernstein-Sato polynomial).
Input: $f \in R_n$; $L \subseteq D_n$ such that D_n/L is holonomic and f-torsion free.
Output: The Bernstein polynomial $b_f^L(s)$.

1. Determine $J^L(f^s)$ following Algorithm 10.
2. Find a reduced Gröbner basis for the ideal $J^L(f^s) + D_n[s] \cdot f$ using an elimination order for x and ∂.
3. Pick the unique element $b(s) \in K[s]$ contained in that basis and return it.

End.

We illustrate the algorithm with two examples. We first recall f which was defined at the end of the previous subsection.

```
i13 : f

       2    2    2    2
o13 = x  + y  + z  + w

o13 : D
```

Now we compute the Bernstein-Sato polynomial.

```
i14 : globalBFunction(f)

        2
o14 = $s  + 3$s + 2

o14 : QQ [$s]
```

`globalBFunction` is the interior name in *Macaulay 2* for the Bernstein-Sato polynomial of f on R_n. We also take a look at the Bernstein-Sato polynomial of a cubic:

```
i15 : g=x^3+y^3+z^3+w^3

       3    3    3    3
o15 = x  + y  + z  + w

o15 : D
```

```
i16 : factorBFunction globalBFunction(g)

               8              5                 7        4
o16 = ($s + -)($s + 2)($s + -)($s + 1)($s + -)($s + -)
               3              3                 3        3

o16 : Product
```

In *Macaulay 2* one can also find $b_f^L(s)$ for more general L. We will see in the following remark what the appropriate commands are.

Remark 13. It is clear that $s+1$ is always a factor of any Bernstein-Sato polynomial on R_n, but this is not necessarily the case if $L \neq \Delta$. For example, $b_f^L(s) = s$ for $n = 1$, $f = x$ and $L = x\partial_x + 1$ (in which case $D_1/L \cong R_1[x^{-1}]$, generated by $\frac{1}{x}$). In particular, it is not true that the roots of $b_f^L(s)$ are negative for general holonomic L.

If L is equal to Δ, and if f is nice, then the Bernstein roots are all between $-n$ and 0 [45]. But for general f very little is known besides a famous theorem of Kashiwara that states that $b_f^{\Delta}(s)$ factors over $\mathbb{Q}$ [19] and all roots are negative.

For L arbitrary, the situation is more complicated. The Bernstein-Sato polynomial of any polynomial f on the D_n-module generated by $\overline{1} \otimes f^a$ with $a \in K$ is related to that of f on D_n/L by a simple shift, and so the Bernstein roots of f on the D_n-module generated by the function germ f^a, $a \in K$, are still all in K by [19]. Localizing other modules however can easily lead to nonrational roots. As an example, consider

```
i17 : D1 = QQ[x,Dx,WeylAlgebra => {x=>Dx}];

i18 : I1 = ideal((x*Dx)^2+1)

               2  2
o18 = ideal(x Dx  + x*Dx + 1)

o18 : Ideal of D1
```

This is input defined over the rationals. Even localizing D_1/I_1 at a very simple f leads to nonrational roots:

```
i19 : f1 = x;

i20 : b=globalB(I1, f1)

                                       2
o20 = HashTable{Boperator => - x*Dx  + 2Dx*$s + Dx}
                                     2
                Bpolynomial => $s  + 2$s + 2

o20 : HashTable
```

`globalB` is the routine to be used if a Bernstein-Sato polynomial is suspected to fail to factor over $\mathbb{Q}$. If $b_f^L(s)$ does factor over $\mathbb{Q}$, one can also use the routine `DlocalizeAll` to be discussed below. It would be very interesting to determine rules that govern the splitting field of $b_f^L(s)$ in general.

3.4 Specializing Exponents

In this subsection we investigate the result of substituting $a \in K$ for s in $J^L(f^s)$. Recall that the Bernstein polynomial $b_f^L(s)$ will exist (i.e., be nonzero) if D_n/L is holonomic. As outlined in the previous subsection, $b_f^L(s)$ can be computed if D_n/L is holonomic and f-torsion free. The following proposition (Proposition 7.3 in [31], see also Proposition 6.2 in [19]) shows that replacing s by an exponent in very general position leads to a solution of the localization problem.

Proposition 14. *If L is holonomic and $a \in K$ is such that no element of $\{a-1, a-2, \ldots\}$ is a Bernstein root of f on L then we have D_n-isomorphisms*

$$(D_n/L) \otimes_{R_n} \left(R_n[f^{-1}] \otimes f^a\right) \cong \left(D_n[s]/J^L(f^s)\right)|_{s=a} \cong D_n \bullet (\overline{1} \otimes 1 \otimes f^a). \quad (3.2)$$

□

One notes in particular that if any $a \in \mathbb{Z}$ satisfies the conditions of the proposition, then so does every integer smaller than a. This motivates the following

Definition 15. The *stable integral exponent of f on L* is the smallest integral root of $b_f^L(s)$, and denoted a_f^L.

In terms of this definition,

$$\left(D_n/J^L(f^s)\right)|_{s=a_f^L} \cong (D_n/L) \otimes_{R_n} R_n[f^{-1}],$$

and the presentation corresponds to the generator $\overline{1} \otimes f^{a_f^L}$. If $L = \Delta$ then Kashiwara's result tells us that $b_f^L(s)$ will factor over the rationals, and thus it is very easy to find the stable integral exponent. If we localize a more general module, the roots may not even be K-rational anymore as we saw at the end of the previous subsection.

The following lemma deals with the question of finding the smallest integer root of a polynomial. We let $|s|$ denote the complex absolute value.

Lemma 16. *Suppose that in the situation of Corollary 11*

$$b_f^L(s) = s^d + b_{d-1}s^{d-1} + \ldots + b_0,$$

and define $B = \max_i\{|b_i|^{1/(d-i)}\}$. The smallest integer root of $b_f^L(s)$ is an integer between $-2B$ and $2B$.

If in particular $L = D_n \cdot (\partial_1, \ldots, \partial_n)$, it suffices to check the integers between $-b_{d-1}$ and -1.

Proof. Suppose $|s_0| = 2B\rho$ where B is as defined above and $\rho > 1$. Assume also that s_0 is a root of $b_f^L(s)$. We find

$$\begin{aligned}(2B\rho)^d = |s_0|^d = |-\sum_0^{d-1} b_i {s_0}^i| &\leq \sum_0^{d-1} B^{d-i}|s_0|^i \\ &= B^d \sum_0^{d-1} (2\rho)^i \leq B^d((2\rho)^d - 1),\end{aligned}$$

using $\rho \geq 1$. By contradiction, s_0 is not a root.

The final claim is a consequence of Kashiwara's work [19] where he proves that if $L = D_n \cdot (\partial_1, \ldots, \partial_n)$ then all roots of $b_f^L(s)$ are rational and negative, and hence $-b_{n-1}$ is a lower bound for each single root. $\square$

Combining Proposition 14 with Algorithms 10 and 12 we therefore obtain

Algorithm 17 (Localization).
INPUT: $f \in R_n$; $L \subseteq D_n$ such that D_n/L is holonomic and f-torsion free.
OUTPUT: Generators for an ideal J such that $(D_n/L) \otimes_{R_n} R_n[f^{-1}] \cong D_n/J$.

1. Determine $J^L(f^s)$ following Algorithm 10.
2. Find the Bernstein polynomial $b_f^L(s)$ using Algorithm 12.
3. Find the smallest integer root a of $b_f^L(s)$.
4. Replace s by a in all generators for $J^L(f^s)$ and return these generators.

End.

Algorithms 12 and 17 are Theorems 6.14 and Proposition 7.3 in [31].

Example 18. For $f = x^2 + y^2 + z^2 + w^2$, we found a stable integral exponent of -2 in the previous subsection. To compute the annihilator of f^{-2} using *Macaulay 2*, we use the command `Dlocalize` which automatically uses the stable integral exponent. We first change the current ring back to the ring `D` which we used in the previous subsection:

```
i21 : use D

o21 = D

o21 : PolynomialRing
```

Here is the module to be localized.

```
i22 : R = (D^1/ideal(Dx,Dy,Dz,Dw))

o22 = cokernel {0} | Dx Dy Dz Dw |

                                   1
o22 : D - module, quotient of D
```

The localization then is obtained by running

```
i23 : ann2 = relations Dlocalize(R,f)

o23 = {0} | wDz-zDw wDy-yDw zDy-yDz wDx-xDw zDx-xDz yDx-xDy xDx+yDy+zD ···

                     1       10
o23 : Matrix D  <--- D
```

The output `ann2` is a 1×10 matrix whose entries generate $\mathrm{ann}_{D_4}(f^{-2})$.

Remark 19. The computation of the annihilator of f^a for values of a such that $a-k$ is a Bernstein root for some $k \in \mathbb{N}^+$ can be achieved by an appropriate syzygy computation. For example, we saw above that the Bernstein-Sato polynomial of $f = x^2+y^2+z^2+w^2$ on R_4 is $(s+1)(s+2)$. So evaluation of $J^L(f^s)$ at -1 does not necessarily yield $\mathrm{ann}_{D_4}(f^{-1})$, as will be documented in the next remark. On the other hand, evaluation at -2 gives $\mathrm{ann}_{D_4}(f^{-2})$. It is not hard to see that $\mathrm{ann}_{D_4}(f^{-1}) = \{P \in D_n : Pf \in \mathrm{ann}_{D_n}(f^{-2})\}$ because $D_4 \bullet f^{-1} = D_4 f \bullet f^{-2} \subseteq D_4 \bullet f^{-2}$. So we set:

```
i24 : F = matrix{{f}}

o24 = {0} | x2+y2+z2+w2 |

              1        1
o24 : Matrix D  <--- D
```

To find $\mathrm{ann}_{D_4}(f^{-1})$, we use the command `modulo` which computes relations: `modulo(M,N)` computes for two matrices M, N the set of (vectors of) operators P such that $P \cdot M \subseteq \mathrm{im}(N)$.

```
i25 : ann1 = gb modulo(F,ann2)

o25 = {2} | wDz-zDw wDy-yDw zDy-yDz Dx^2+Dy^2+Dz^2+Dw^2 wDx-xDw zDx-xD ···

o25 : GroebnerBasis
```

Interesting is in particular the generator $\partial_2^2+\partial_y^2+\partial_z^2+\partial_w^2$. To see the quotient of $D_4 \bullet f^{-2}$ by $D_4 \bullet f^{-1}$ we execute

```
i26 : gb((ideal ann2) + (ideal F))

o26 = {0} | w z y x |

o26 : GroebnerBasis
```

which shows that $D_4 \bullet f^{-2}$ is an extension of $D_4/D_4(x,y,z,w)$ by $D_4 \bullet f^{-1}$. This is not surprising, since $(0,0,0,0)$ is the only singularity of f and hence the difference between $D_4 \bullet f^{-2}$ and $D_4 \bullet f^{-1}$ must be supported at the origin.

It is perhaps interesting to note that for more complicated (but still prime) f the quotient $\frac{D_n \bullet f^a}{D_n \bullet f^{a+1}}$ can be greater than a simple D_n-module. For example, let $f = x^3+y^3+z^3+w^3$ and $a = a_f^{\Delta} = -2$. A computation similar to the quadric case above shows that here $\frac{D_n \bullet f^a}{D_n \bullet f^{a+1}}$ is a (x,y,z,w)-torsion module (supported at the singular locus of f) isomorphic to $(D_4/D_4 \cdot (x,y,z,w))^6$. The socle elements of the quotient are the degree 2 polynomials in x,y,z,w.

Example 20. Here we show how with *Macaulay 2* one can get more information from the localization procedure.

```
i27 : D = QQ[x,y,z,Dx,Dy,Dz, WeylAlgebra => {x=>Dx, y=>Dy, z=>Dz}];

i28 : Delta = ideal(Dx,Dy,Dz);

o28 : Ideal of D
```

We now define a polynomial and compute the localization of R_3 at the polynomial.

```
i29 : f=x^3+y^3+z^3;
```

```
i30 : I1=DlocalizeAll(D^1/Delta,f,Strategy=>Oaku)
localize: Warning: Oaku's strategy requires
             module to be f-saturated

o30 = HashTable{GeneratorPower => -2                                   ···
                LocMap => {0} | x6+2x3y3+y6+2x3z3+2y3z3+z6 |
                LocModule => cokernel {0} | 1/3xDx+1/3yDy+1/3zDz+2 z2D ···
                                        2       5       4
                Bfunction => ($s + 1) ($s + -)($s + -)($s + 2)
                                                3       3
                                   1         1         1             2        ···
                annFS => ideal (-*x*Dx + -*y*Dy + -*z*Dz - $s, z Dy -  ···
                                   3         3         3
                                 2       3          2        4       1   2    ···
                Boperator => --*y*z*Dx Dy*Dz - --*y*z*Dy Dz + ---*z Dx ···
                                81                 81               243       ···

o30 : HashTable

i31 : I2=DlocalizeAll(D^1/Delta,f)

o31 = HashTable{GeneratorPower => -2                                   ···
                LocMap => {0} | x6+2x3y3+y6+2x3z3+2y3z3+z6 |
                LocModule => cokernel {0} | xDx+yDy+zDz+6 z2Dy-y2Dz z2 ···
                                                   2     2        2        1
                IntegrateBfunction => ($s + 1) ($s) ($s + -)($s + -)
                                                                  3        3

o31 : HashTable
```

The last two commands both compute the localization of R_3 at f but follow different localization algorithms. The former uses our Algorithm 17 while the latter follows [36].

The output of the command `DlocalizeAll` is a hashtable, because it contains a variety of data that pertain to the map $R_n \hookrightarrow R_n[f^{-1}]$. `LocMap` gives the element that induces the map on the D_n-module level (by right multiplication). `LocModule` gives the localized module as cokernel of the displayed matrix. `Bfunction` is the Bernstein-Sato polynomial and `annFS` the generic annihilator $J^L(f^s)$. `Boperator` displays a Bernstein operator and the stable integral exponent is stored in `GeneratorPower`.

The warning `Oaku's ...saturated` refers to the fact that Algorithm 17 requires the ideal L to be f-saturated. This property is not checked by *Macaulay 2*, so the user needs to make sure it holds. For example, this is always the case if D_n/L is a localization of R_n. One can check the saturation property in *Macaulay 2*, but it is a rather involved computation. This can be circumvented by omitting the option `Strategy=>Oaku`, in which case the localization algorithm of [36] is used. In terms of complexity, using the Oaku strategy is much better behaved.

One can address the entries of a hashtable. For example, executing

```
i32 : I1.LocModule

o32 = cokernel {0} | 1/3xDx+1/3yDy+1/3zDz+2 z2Dy-y2Dz z2Dx-x2Dz y2Dx-x ···

                             1
o32 : D - module, quotient of D
```

one can see that $R_3[f^{-1}]$ is isomorphic to the cokernel of the `LocModule` entry which (for either localization method) is

$$D_3/\ D_3\cdot(\ x\partial_x + y\partial_y + z\partial_z + 6,\ \ z^2\partial_y - y^2\partial_z,\ \ x^3\partial_y + y^3\partial_y + y^2z\partial_z + 6y^2,\\ z^2\partial_x - x^2\partial_z,\ \ y^2\partial_x - x^2\partial_y,\ \ x^3\partial_z + y^3\partial_z + z^3\partial_z + 6z^2).$$

The first line of the hashtable `I1` shows that $R_3[f^{-1}]$ is generated by f^{-2} over D_3, while `I1.LocMap` shows that the natural inclusion $D_3/\Delta = R_3 \hookrightarrow R_3[f^{-1}] = D_3/J^{\Delta}(f^s)|_{s=a_f^{\Delta}}$ is given by right multiplication by f^2, shown as the third entry of the hashtable `I1`. It is perhaps useful to point out that the fourth entry of hashtable `I2` is a relative of the Bernstein-Sato polynomial of f, and is used for the computation of the so-called restriction functor (compare with [33,47]).

Remark 21. Plugging in bad values a for s (such that $a-k$ *is* a Bernstein root for some $k \in \mathbb{N}^+$) can have unexpected results. Consider the case $n=1$, $f=x$. Then $J^{\Delta}(f^s) = D_1\cdot(s+1-\partial_1 x_1)$. Hence $b_f^{\Delta}(s) = s+1$ and -1 is the unique Bernstein root. According to Proposition 14,

$$\left(D_1[s]/J^{\Delta}(f^s)\right)|_{s=a} \cong R_1[{x_1}^{-1}]\otimes {x_1}^a \cong D_1\bullet {x_1}^a$$

for all $a\in K\setminus\mathbb{N}$. For $a\in\mathbb{N}^+$, we also have $D_1[s]/J^{\Delta}(f^s)|_{s=a}\cong D_1\bullet x^a$, but this is of course not $R_1[{x_1}^{-1}]$ but just R_1.

For $a=0$ however, $\left(D_1[s]/J^{\Delta}(f^s)\right)|_{s=a}$ has x_1-torsion! It equals in fact what is called the Fourier transform of $R_1[{x_1}^{-1}]$ and fits into an exact sequence

$$0\to H^1_{x_1}(R_1)\to\mathcal{F}(R_1[{x_1}^{-1}])\to R_1\to 0.$$

Remark 22. If D_n/L is holonomic but has f-torsion, then $(D_n/L)\otimes R_n[f^{-1}]$ and $((D_n/L)/H^0_{(f)}(D_n/L))\otimes R_n[f^{-1}]$ are of course isomorphic. So if we knew how to find $M/H^0_f(M)$ for holonomic modules M, our localization algorithm could be generalized to all holonomic modules. There are two different approaches to the problem of f-torsion, presented in [33] and in [42,43]. The former is based on homological methods and restriction to the diagonal while the latter aims at direct computation of those $P\in D_n$ for which $f^kP\in L$ for some k.

There is also another direct method for localizing $M=D_n/L$ at f that works in the situation where the nonholonomic locus of M is contained in the variety of f (irrespective of torsion). It was proved by Kashiwara, that $M[f^{-1}]$ is then holonomic, and in [36] an algorithm based on integration is given that computes a presentation for it.

4 Local Cohomology Computations

The purpose of this section is to present algorithms that compute for given $i,j,k\in\mathbb{N}, I\subseteq R_n$ the structure of the local cohomology modules $H^k_I(R_n)$ and $H^i_{\mathfrak{m}}(H^j_I(R_n))$, and the invariants $\lambda_{i,j}(R_n/I)$ associated to I.

In particular, the algorithms detect the vanishing of local cohomology modules.

4.1 Local cohomology

We will first describe an algorithm that takes in a finite set of polynomials $\{f_1, \ldots, f_r\} \subset R_n$ and returns a presentation of $H_I^k(R_n)$ where $I = R_n \cdot (f_1, \ldots, f_r)$. In particular, if $H_I^k(R_n)$ is zero, then the algorithm will return the zero presentation.

Definition 23. Let Θ_k^r be the set of k-element subsets of $1, \ldots, r$ and for $\theta \in \Theta_k^r$ write F_θ for the product $\prod_{i\in\theta} f_i$.

Consider the Čech complex $\check{C}^\bullet = \check{C}^\bullet(f_1, \ldots, f_r)$ associated to $f_1, \ldots, f_r$ in R_n,

$$0 \to R_n \to \bigoplus_{\theta\in\Theta_1^r} R_n[{F_\theta}^{-1}] \to \bigoplus_{\theta\in\Theta_2^r} R_n[{F_\theta}^{-1}] \to \cdots \to R_n[(f_1 \cdots f_r)^{-1}] \to 0. \tag{4.1}$$

Its k-th cohomology group is $H_I^k(R_n)$. The map

$$M_k : \left(\check{C}^k = \bigoplus_{\theta\in\Theta_k^r} R_n[{F_\theta}^{-1}]\right) \to \left(\bigoplus_{\theta'\in\Theta_{k+1}^r} R_n[{F_{\theta'}}^{-1}] = \check{C}^{k+1}\right) \tag{4.2}$$

is the sum of maps

$$R_n[(f_{i_1} \cdots f_{i_k})^{-1}] \to R_n[(f_{j_1} \cdots f_{j_{k+1}})^{-1}] \tag{4.3}$$

which are zero if $\{i_1, \ldots, i_k\} \not\subseteq \{j_1, \ldots, j_{k+1}\}$, or send $\frac{1}{1}$ to $\frac{1}{1}$ (up to sign). With $D_n/\Delta \cong R_n$, identify $R_n[(f_{i_1} \cdots f_{i_k})^{-1}]$ with $D_n/J^\Delta((f_{i_1} \cdots f_{i_k})^s)|_{s=a}$ and $R_n[(f_{j_1} \cdots f_{j_{k+1}})^{-1}]$ with $D_n/J^\Delta((f_{j_1} \cdots f_{j_{k+1}})^s)|_{s=a'}$ where a, a' are sufficiently small integers. By Proposition 14 we may assume that $a = a' \le 0$. Then the map (4.3) is in the nonzero case multiplication from the right by $(f_l)^{-a}$ where $l = \{j_1, \ldots, j_{k+1}\} \backslash \{i_1, \ldots, i_k\}$, again up to sign. For example, consider the inclusion

$$D_2/D_2 \cdot (\partial_x x, \partial_y) = R_2[x^{-1}] \hookrightarrow R_2[(xy)^{-1}] = D_2/D_2 \cdot (\partial_x x, \partial_y y).$$

Since $\frac{1}{x} = \frac{y}{xy}$, the inclusion on the level of D_2-modules maps $P + \operatorname{ann}(x^{-1})$ to $Py + \operatorname{ann}((xy)^{-1})$.

It follows that the matrix representing the map $\check{C}^k \to \check{C}^{k+1}$ in terms of D_n-modules is very easy to write down once the annihilator ideals and Bernstein polynomials for all k- and $(k+1)$-fold products of the f_i are known: the entries are 0 or $\pm f_l^{-a}$ where f_l is the new factor. These considerations give the following

Algorithm 24 (Local cohomology).
INPUT: $f_1, \dots, f_r \in R_n; k \in \mathbb{N}$.
OUTPUT: $H^k_I(R_n)$ in terms of generators and relations as finitely generated D_n-module where $I = R_n \cdot (f_1, \dots, f_r)$.

1. Compute the annihilator ideal $J^{\Delta}((F_\theta)^s)$ and the Bernstein polynomial $b^{\Delta}_{F_\theta}(s)$ for all $(k-1)$-, k- and $(k+1)$-fold products F_θ of $f_1, \dots, f_r$ following Algorithms 10 and 12 (so θ runs through $\Theta^r_{k-1} \cup \Theta^r_k \cup \Theta^r_{k+1}$).
2. Compute the stable integral exponents $a^{\Delta}_{F_\theta}$, let a be their minimum and replace s by a in all the annihilator ideals.
3. Compute the two matrices M_{k-1}, M_k representing the D_n-linear maps $\check{C}^{k-1} \to \check{C}^k$ and $\check{C}^k \to \check{C}^{k+1}$ as explained above.
4. Compute a Gröbner basis G for the kernel of the composition
$$\bigoplus_{\theta \in \Theta^r_k} D_n \twoheadrightarrow \bigoplus_{\theta \in \Theta^r_k} D_n / J^{\Delta}({F_\theta}^s)|_{s=a} \xrightarrow{M_k} \bigoplus_{\theta' \in \Theta^r_{k+1}} D_n / J^{\Delta}({F_{\theta'}}^s)|_{s=a}.$$
5. Compute a Gröbner basis G_0 for the preimage in $\bigoplus_{\theta \in \Theta^r_k} D_n$ of the module
$$\operatorname{im}(M_{k-1}) \subseteq \bigoplus_{\theta \in \Theta^r_k} D_n / J^{\Delta}((F_\theta)^s)|_{s=a} \twoheadleftarrow \bigoplus_{\theta \in \Theta^r_k} D_n$$
under the indicated projection.
6. Compute the remainders of all elements of G with respect to G_0.
7. Return these remainders and G_0.

End.

The nonzero elements of G generate the quotient $G/G_0 \cong H^k_I(R_n)$ so that in particular $H^k_I(R_n) = 0$ if and only if all returned remainders are zero.

Example 25. Let I be the ideal in $R_6 = K[x_1, \dots, x_6]$ that is generated by the 2×2 minors f, g, h of the matrix $\begin{pmatrix} x_1 & x_2 & x_3 \\ x_4 & x_5 & x_6 \end{pmatrix}$. Then $H^i_I(R_6) = 0$ for $i < 2$ and $H^2_I(R_6) \neq 0$ because I is a height 2 prime, and $H^i_I(R_6) = 0$ for $i > 3$ because I is 3-generated, so the only open case is $H^3_I(R_6)$. This module in fact does not vanish, and our algorithm provides a proof of this fact by direct calculation. The *Macaulay 2* commands are as follows.

```
i33 : D= QQ[x,y,z,u,v,w,Dx,Dy,Dz,Du,Dv,Dw, WeylAlgebra =>
             {x=>Dx, y=>Dy, z=>Dz, u=>Du, v=>Dv, w=>Dw}];

i34 : Delta=ideal(Dx,Dy,Dz,Du,Dv,Dw);

o34 : Ideal of D

i35 : R=D^1/Delta;

i36 : f=x*v-u*y;

i37 : g=x*w-u*z;

i38 : h=y*w-v*z;
```

These commands define the relevant rings and polynomials. The following three compute the localization of R_6 at f:

```
i39 : Rf=DlocalizeAll(R,f,Strategy => Oaku)
localize: Warning: Oaku's strategy requires
             module to be f-saturated

o39 = HashTable{Bfunction => ($s + 1)($s + 2)                             ...
                Boperator => - Dy*Du + Dx*Dv
                GeneratorPower => -2
                LocMap => {0} | y2u2-2xyuv+x2v2 |
                LocModule => cokernel {0} | Dw Dz xDu+yDv yDy-uDu xDy+ ...
                annFS => ideal (Dw, Dz, x*Du + y*Dv, y*Dy - u*Du, x*Dy ...

o39 : HashTable
```

of $R_6[f^{-1}]$ at g:

```
i40 : Rfg=DlocalizeAll(Rf.LocModule,g, Strategy => Oaku)
localize: Warning: Oaku's strategy requires
             module to be f-saturated

o40 = HashTable{Bfunction => ($s)($s + 1)                                 ...
                Boperator => - Dz*Du + Dx*Dw
                GeneratorPower => -1
                LocMap => {0} | -zu+xw |
                LocModule => cokernel {0} | DzDv-DyDw xDu+yDv+zDw zDz- ...
                                                                          ...
                annFS => ideal (Dz*Dv - Dy*Dw, x*Du + y*Dv + z*Dw, z*D ...

o40 : HashTable
```

and of $R_6[(fg)^{-1}]$ at h:

```
i41 : Rfgh=DlocalizeAll(Rfg.LocModule,h, Strategy => Oaku)
localize: Warning: Oaku's strategy requires
             module to be f-saturated

o41 = HashTable{Bfunction => ($s - 1)($s + 1)                             ...
                Boperator => - Dz*Dv + Dy*Dw
                GeneratorPower => -1
                LocMap => {0} | -zv+yw |
                LocModule => cokernel {0} | xDu+yDv+zDw zDz-uDu-vDv-2  ...
                                                                          ...
                annFS => ideal (x*Du + y*Dv + z*Dw, z*Dz - u*Du - v*Dv ...

o41 : HashTable
```

From the output of these commands one sees that $R_6[(fgh)^{-1}]$ is generated by $\frac{1}{f^2gh}$. This follows from considering the stable integral exponents of the three localization procedures, encoded in the hashtable entry `GeneratorPower`: for example,

```
i42 : Rf.GeneratorPower

o42 = -2
```

shows that the generator for $R_6[f^{-1}]$ is f^{-2}. Now we compute the annihilator of $H^3_I(R_6)$. From the Čech complex it follows that $H^3_I(R_6)$ is the quotient of the output of `Rfgh.LocModule` (isomorphic to $R_6[(fgh)^{-1}]$) by the submodules generated by f^2, g and h. (These submodules represent $R_6[(gh)^{-1}]$, $R_6[(fh)^{-1}]$ and $R_6[(fg)^{-1}]$ respectively.)

```
i43 : Jfgh=ideal relations Rfgh.LocModule;

o43 : Ideal of D

i44 : JH3=Jfgh+ideal(f^2,g,h);

o44 : Ideal of D
```

```
i45 : JH3gb=gb JH3

o45 = {0} | w z uDu+vDv+wDw+4 xDu+yDv+zDw yDy-uDu-wDw-1 xDy+uDv uDx+vD ···

o45 : GroebnerBasis
```

So JH3 is the ideal of D_3 generated by

$$\begin{gathered} w,\ z,\ u\partial_u + v\partial_v + w\partial_w + 4,\ x\partial_u + y\partial_v + z\partial_w,\ y\partial_y - u\partial_u - w\partial_w - 1, \\ x\partial_y + u\partial_v,\ u\partial_x + v\partial_y + w\partial_z,\ y\partial_x + v\partial_u,\ x\partial_x - v\partial_v - w\partial_w - 1, \\ v^2,\ uv,\ yv,\ u^2,\ yu + xv,\ xu,\ y^2,\ xy, \\ x^2,\ xv\partial_v + 2x,\ v\partial_y\partial_u + w\partial_z\partial_u - v\partial_x\partial_v - w\partial_x\partial_w - 3\partial_x \end{gathered}$$

which form a Gröbner basis. This proves that $H^3_I(R) \neq 0$, because 1 is not in the Gröbner basis of JH3. (There are also algebraic and topological proofs to this account. Due to Hochster, and Bruns and Schwänzl, they are quite ingenious and work only in rather special situations.)

From our output one can see that $H^3_I(R_6)$ is (x, y, z, u, v, w)-torsion as JH3 contains $(x, y, z, u, v, w)^2$. The following sequence of commands defines a procedure testmTorsion which as the name suggests tests a module D_n/L for being $\mathfrak{m}$-torsion. We first replace the generators of L with a Gröbner basis. Then we pick the elements of the Gröbner basis not using any ∂_i. If now the left over polynomials define an ideal of dimension 0 in R_n, the ideal was $\mathfrak{m}$-torsion and otherwise not.

```
i46 : testmTorsion = method();

i47 : testmTorsion (Ideal):= (L) -> (
           LL = ideal generators gb L;
           n = numgens (ring (LL)) // 2;
           LLL = ideal select(first entries gens LL, f->(
              l = apply(listForm f, t->drop(t#0,n));
              all(l, t->t==toList(n:0))
              ));
           if dim inw(LLL,toList(apply(1..2*n,t -> 1))) == n
           then true
           else false
           )

o47 = --Function[stdio:49:58]--

o47 : Function
```

If we apply testmTorsion to JH3 we obtain

```
i48 : testmTorsion(JH3)

o48 = true
```

Further inspection shows that the ideal JH3 is in fact the annihilator of the fraction $\frac{f}{wzx^2y^2u^2v^2}$ in $R_6[(xyzuvw)^{-1}]/R_6 \cong D_6/D_6 \cdot (x, y, z, u, v, w)$, and that this fraction generates $D_6/D_6 \cdot (x_1, \ldots, x_6)$. Since $D_6/D_6 \cdot (x_1, \ldots, x_6)$ is isomorphic to $E_{R_6}(R_6/R_6 \cdot (x_1, \ldots, x_6))$, the injective hull of $R_6/R_6 \cdot (x_1, \ldots, x_6) = K$ in the category of R_6-modules, we conclude that $H^3_I(R_6) \cong E_{R_6}(K)$. (In the next subsection we will display a way to use *Macaulay 2* to find the length of an $\mathfrak{m}$-torsion module.)

In contrast, let I be defined as generated by the three minors, but this time over a field of finite characteristic. Then $H^3_I(R_6)$ is zero because Peskine and

Szpiro proved using the Frobenius functor [38] that R_6/I Cohen-Macaulay implies that $H^k_I(R_6)$ is nonzero only if $k = \mathrm{codim}(I)$.

Also opposite to the above example, but in any characteristic, is the following calculation. Let I be the ideal in $K[x, y, z, w]$ describing the twisted cubic: $I = R_4 \cdot (f, g, h)$ with $f = xz - y^2$, $g = yw - z^2$, $h = xw - yz$. The projective variety V_2 defined by I is isomorphic to the projective line. It is of interest to determine whether V_2 and other Veronese embeddings of the projective line are complete intersections. The set-theoretic complete intersection property can occasionally be ruled out with local cohomology techniques: if V is of codimension c in the affine variety X and $H^{c+k}_{I(V)}(O(X)) \neq 0$ for any positive k then V cannot be a set-theoretic complete intersection. In the case of the twisted cubic, it turns out hat $H^3_I(R_4) = 0$ as can be seen from the following computation:

```
i49 : D=QQ[x,y,z,w,Dx,Dy,Dz,Dw,WeylAlgebra => {x=>Dx, y=>Dy, z=>Dz, w= ···

i50 : f=y^2-x*z;

i51 : g=z^2-y*w;

i52 : h=x*w-y*z;

i53 : Delta=ideal(Dx,Dy,Dz,Dw);

o53 : Ideal of D

i54 : R=D^1/Delta;

i55 : Rf=DlocalizeAll(R,f,Strategy => Oaku)
localize: Warning: Oaku's strategy requires
             module to be f-saturated

o55 = HashTable{GeneratorPower => -1                                    ···
                LocMap => {0} | y2-xz |
                LocModule => cokernel {0} | Dw xDy+2yDz yDx+1/2zDy xDx ···
                                                        1               ···
                annFS => ideal (Dw, x*Dy + 2y*Dz, y*Dx + -*z*Dy, x*Dx  ···
                                                        2               ···
                                                3
                Bfunction => ($s + 1)($s + -)
                                                2
                              1   2
                Boperator => -*Dy  - Dx*Dz
                              4

o55 : HashTable

i56 : Rfg=DlocalizeAll(Rf.LocModule,g, Strategy => Oaku);
localize: Warning: Oaku's strategy requires
             module to be f-saturated

i57 : Rfgh=DlocalizeAll(Rfg.LocModule,h, Strategy => Oaku);
localize: Warning: Oaku's strategy requires
             module to be f-saturated

i58 : Ifgh=ideal relations Rfgh.LocModule;

o58 : Ideal of D

i59 : IH3=Ifgh+ideal(f,g,h);

o59 : Ideal of D
```

```
i60 : IH3gb=gb IH3

o60 = {0} | 1 |

o60 : GroebnerBasis
```

It follows, that we cannot conclude from local cohomological considerations that V_2 is not a set-theoretic complete intersection. This is not an accident but typical, as the second vanishing theorem of Hartshorne, Speiser, Huneke and Lyubeznik shows [14,16,18]: if a homogeneous ideal $I \subseteq R_n$ describes an geometrically connected projective variety of positive dimension then $H_I^{n-1}(R_n) = H_I^n(R_n) = 0$.

4.2 Iterated local cohomology

Recall that $\mathfrak{m} = R_n \cdot (x_1, \ldots, x_n)$. As a second application of Gröbner basis computations over the Weyl algebra we show now how to compute the $\mathfrak{m}$-torsion modules $H^i_{\mathfrak{m}}(H^j_I(R_n))$. Note that we cannot apply Lemma 8 to $D_n/L = H^j_I(R_n)$ since $H^j_I(R_n)$ may well contain some torsion.

$\check{C}^j(R_n; f_1, \ldots, f_r)$ denotes the j-th module in the Čech complex to R_n and $\{f_1, \ldots, f_r\}$. Let $\check{C}^{\bullet,\bullet}$ be the double complex

$$\check{C}^{i,j} = \check{C}^i(R_n; x_1, \ldots, x_n) \otimes_{R_n} \check{C}^j(R_n; f_1, \ldots, f_r),$$

with vertical maps $\phi^{\bullet,\bullet}$ induced by the identity on the first factor and the usual Čech maps on the second, and horizontal maps $\xi^{\bullet,\bullet}$ induced by the Čech maps on the first factor and the identity on the second. Now $\check{C}^{i,j}$ is a direct sum of modules $R_n[g^{-1}]$ where $g = x_{\alpha_1} \cdots x_{\alpha_i} \cdot f_{\beta_1} \cdots f_{\beta_j}$. So the whole double complex can be rewritten in terms of D_n-modules and D_n-linear maps using Algorithm 17:

$$\begin{tikzcd}
\check{C}^{i-1,j+1} \arrow[r, "\xi^{i-1,j+1}"] & \check{C}^{i,j+1} \arrow[r, "\xi^{i,j+1}"] & \check{C}^{i+1,j+1} \\
\check{C}^{i-1,j} \arrow[u, "\phi^{i-1,j}"'] \arrow[r, "\xi^{i-1,j}"] & \check{C}^{i,j} \arrow[u, "\phi^{i,j}"'] \arrow[r, "\xi^{i,j}"] & \check{C}^{i+1,j} \arrow[u, "\phi^{i+1,j}"'] \\
\check{C}^{i-1,j-1} \arrow[u, "\phi^{i-1,j-1}"'] \arrow[r, "\xi^{i-1,j-1}"] & \check{C}^{i,j-1} \arrow[u, "\phi^{i,j-1}"'] \arrow[r, "\xi^{i,j-1}"] & \check{C}^{i+1,j-1} \arrow[u, "\phi^{i+1,j-1}"']
\end{tikzcd}$$

Since $\check{C}^i(R_n; x_1, \ldots, x_n)$ is R_n-flat, the column cohomology of $\check{C}^{\bullet,\bullet}$ at (i,j) is $\check{C}^i(R_n; x_1, \ldots, x_n) \otimes_{R_n} H^j_I(R_n)$ and the induced horizontal maps in the j-th row are simply the maps in the Čech complex $\check{C}^\bullet(H^j_I(R_n); x_1, \ldots, x_n)$. It follows that the row cohomology of the column cohomology at (i_0, j_0) is $H^{i_0}_{\mathfrak{m}}(H^{j_0}_I(R_n))$, the object of our interest.

We have, denoting by $X_{\theta'}$ in analogy to F_θ the product $\prod_{i \in \theta'} x_i$, the following

Algorithm 26 (Iterated local cohomology).
INPUT: $f_1, \ldots, f_r \in R_n; i_0, j_0 \in \mathbb{N}$.
OUTPUT: $H^{i_0}_{\mathfrak{m}}(H^{j_0}_I(R_n))$ in terms of generators and relations as finitely generated D_n-module where $I = R_n \cdot (f_1, \ldots, f_r)$.

1. For $i = i_0 - 1, i_0, i_0 + 1$ and $j = j_0 - 1, j_0, j_0 + 1$ compute the annihilators $J^{\Delta}((F_\theta \cdot X_{\theta'})^s)$, Bernstein polynomials $b^{\Delta}_{F_\theta \cdot X_{\theta'}}(s)$, and stable integral exponents $a^{\Delta}_{F_\theta \cdot X_{\theta'}}$ of $F_\theta \cdot X_{\theta'}$ where $\theta \in \Theta^r_j, \theta' \in \Theta^n_i$.
2. Let a be the minimum of all $a^{\Delta}_{F_\theta \cdot X_{\theta'}}$ and replace s by a in all the annihilators computed in the previous step.
3. Compute the matrices to the D_n-linear maps $\phi^{i,j} : \check{C}^{i,j} \to \check{C}^{i,j+1}$ and $\xi^{k,l} : \check{C}^{k,l} \to \check{C}^{k+1,l}$, for $(i,j) \in \{(i_0, j_0), (i_0+1, j_0-1), (i_0, j_0-1), (i_0 - 1, j_0)\}$ and $(k,l) \in \{(i_0, j_0), (i_0 - 1, j_0)\}$.
4. Compute a Gröbner basis G for the module
$$D_n \cdot G = \ker(\phi^{i_0,j_0}) \cap \left[(\xi^{i_0,j_0})^{-1}(\mathrm{im}(\phi^{i_0+1,j_0-1}))\right] + \mathrm{im}(\phi^{i_0,j_0-1})$$
and a Gröbner basis G_0 for the module
$$D_n \cdot G_0 = \xi^{i_0-1,j_0}(\ker(\phi^{i_0-1,j_0})) + \mathrm{im}(\phi^{i_0,j_0-1}).$$
5. Compute the remainders of all elements of G with respect to G_0.
6. Return these remainders together with G_0.

End.

Note that $(D_n \cdot G)/(D_n \cdot G_0)$ is isomorphic to

$$\frac{\ker\left(\frac{\ker(\phi^{i_0,j_0})}{\mathrm{im}(\phi^{i_0,j_0-1})} \xrightarrow{\xi^{i_0,j_0}} \frac{\ker(\phi^{i_0+1,j_0})}{\mathrm{im}(\phi^{i_0+1,j_0-1})}\right)}{\xi^{i_0-1,j_0}\left(\frac{\ker(\phi^{i_0-1,j_0})}{\mathrm{im}(\phi^{i_0-1,j_0-1})}\right)} \cong H^{i_0}_{\mathfrak{m}}(H^{j_0}_I(R_n)).$$

The elements of G will be generators for $H^{i_0}_{\mathfrak{m}}(H^{j_0}_I(R_n))$ and the elements of G_0 generate the extra relations that are not syzygies.

The algorithm can of course be modified to compute any iterated local cohomology group $H^j_J(H^i_I(R_n))$ for $J \supseteq I$ by replacing the generators $x_1, \ldots, x_n$ for $\mathfrak{m}$ by those for J. Moreover, the iteration depth can also be increased by considering "tricomplexes" etc. instead of bicomplexes.

Again we would like to point out that with the methods of [33] or [36] one could actually compute first $H^i_I(R_n)$ and from that $H^j_J(H^i_I(R_n))$, but probably that is quite a bit more complex a computation.

4.3 Computation of Lyubeznik Numbers

G. Lyubeznik proved in [24] that if K is a field, $R = K[x_1, \ldots, x_n]$, $I \subseteq R$, $\mathfrak{m} = R \cdot (x_1, \ldots, x_n)$ and $A = R/I$ then $\lambda_{i,j}(A) = \dim_K \mathrm{soc}_R\, H^i_{\mathfrak{m}}(H^{n-j}_I(R))$

is invariant under change of presentation of A. In other words, it only depends on A and i, j but not the projection $R \twoheadrightarrow A$. Lyubeznik proved that $H^i_{\mathfrak{m}}(H^j_I(R_n))$ is in fact an injective $\mathfrak{m}$-torsion R_n-module of finite socle dimension $\lambda_{i,n-j}(A)$ and so isomorphic to $(E_{R_n}(K))^{\lambda_{i,n-j}(A)}$ where $E_{R_n}(K)$ is the injective hull of K over R_n. We are now in a position to compute these invariants of R_n/I in characteristic zero..

Algorithm 27 (Lyubeznik numbers).
Input: $f_1, \ldots, f_r \in R_n; i, j \in \mathbb{N}$.
Output: $\lambda_{i,n-j}(R_n/R_n \cdot (f_1, \ldots, f_r))$.

1. Using Algorithm 26 find $g_1, \ldots, g_l \in {D_n}^d$ and $h_1, \ldots, h_e \in {D_n}^d$ such that $H^i_{\mathfrak{m}}(H^j_I(R_n))$ is isomorphic to $D_n \cdot (g_1, \ldots, g_l)$ modulo $H = D_n \cdot (h_1, \ldots, h_e)$.
2. Assume that after a suitable renumeration g_1 is not in H. If such a g_1 cannot be chosen, quit.
3. Find a monomial $m \in R_n$ such that $m \cdot g_1 \notin H$ but $x_i m g_1 \in H$ for all x_i.
4. Replace H by $D_n m g_1 + H$ and reenter at Step 2.
5. Return $\lambda_{i,n-j}(R_n/I)$, the number of times Step 3 was executed.

End.

The reason that this works is as follows. We know that $(D_n \cdot g_1 + H)/H$ is $\mathfrak{m}$-torsion (as $H^i_{\mathfrak{m}}(H^j_I(R_n))$ is) and so it is possible (with trial and error, or a suitable syzygy computation) to find the monomial m in Step 3. The element $mg_1 \mod H \in D_n/H$ has annihilator equal to $\mathfrak{m}$ over R_n and therefore generates a D_n-module isomorphic to $D_n/D_n \cdot \mathfrak{m} \cong E_{R_n}(K)$. The injection

$$(D_n \cdot mg_1 + H)/H \hookrightarrow (D_n \cdot (g_1, \ldots, g_l) + H)/H$$

splits as map of R_n-modules because $E_{R_n}(K)$ is injective and so the cokernel $D_n \cdot (g_1, \ldots, g_l)/D_n \cdot (mg_1, h_1, \ldots, h_e)$ is isomorphic to $(E_{R_n}(K))^{\lambda_{i,n-j}(A)-1}$.

Reduction of the g_i with respect to a Gröbner basis of the new relation module and repetition will lead to the determination of $\lambda_{i,n-j}(A)$.

Assume that D_n/L is an $\mathfrak{m}$-torsion module. For example, we could have $D_n/L \cong H^i_{\mathfrak{m}}(H^j_I(R_n))$. Here is a procedure that finds by trial and error the monomial socle element m of Step 3 in Algorithm 26.

```
i61 : findSocle = method();

i62 : findSocle(Ideal, RingElement):= (L,P) -> (
          createDpairs(ring(L));
          v=(ring L).dpairVars#0;
          myflag = true;
          while myflag do (
        w = apply(v,temp -> temp*P % L);
        if all(w,temp -> temp == 0) then myflag = false
        else (
             p = position(w, temp -> temp != 0);
             P = v#p * P;)
        );
          P
          )

o62 = --Function[stdio:74:85]--

o62 : Function
```

For example, application of this socle search to the ideal describing $H^3_I(R_6)$ of Example 25 gives

```
i63 : D=ring JH3

o63 = D

o63 : PolynomialRing
```

(As D was until now the differential operators on $\mathbb{Q}[x, y, z, w]$.)

```
i64 : findSocle(JH3,1_D)

o64 = x*v

o64 : D
```

One can then repeat the socle search and kill the newly found element as suggested in the explanation above:

```
i65 : findLength = method();

i66 : findLength(Ideal):= (I) -> (
          l = 0;
          while (I != ideal(1_(ring(I)))) do (
        l = l+1;
        s = findSocle(I,1_(ring(I)));
        I = I + ideal(s););
          l
          )

o66 = --Function[stdio:90:96]--

o66 : Function
```

Applied to JH3 of the previous subsection this yields

```
i67 : findLength(JH3)

o67 = 1
```

and hence JH3 does indeed describe a module isomorphic to $E_{R_6}(K)$.

5 Implementation, Examples, Questions

5.1 Implementations and Optimizing

The Algorithms 10, 12 and 17 have first been implemented by T. Oaku and N. Takayama using the package Kan [41] which is a postscript language for computations in the Weyl algebra and in polynomial rings. In *Macaulay 2* Algorithms 10, 12 and 17 as well as Algorithm 24 have been implemented by A. Leykin, M. Stillman and H. Tsai. They additionally implemented a wealth of D-module routines that relate to topics which we cannot all cover in this chapter. These include homomorphisms between holonomic modules and extension functors, restriction functors to linear subspaces, integration (de Rham) functors to quotient spaces and others. For further theoretical information the reader is referred to [33–35,39,44,47–49].

Computation of Gröbner bases in many variables is in general a time and space consuming enterprise. In commutative polynomial rings the worst case performance for the number of elements in reduced Gröbner bases is doubly exponential in the number of variables and the degrees of the generators. In the (relatively) small Example 25 above R_6 is of dimension 6, so that the intermediate ring $D_{n+1}[y_1, y_2]$ contains 16 variables. In view of these facts the following idea has proved useful.

The general context in which Lemma 8 and Proposition 14 were stated allows successive localization of $R_n[(fg)^{-1}]$ in the following way. First one computes $R_n[f^{-1}]$ according to Algorithm 17 as quotient $D_n/J^{\Delta}(f^s)|_{s=a}$, $\mathbb{Z} \ni a \ll 0$. Then $R_n[(fg)^{-1}]$ may be computed using Algorithm 17 again since $R_n[(fg)^{-1}] \cong R_n[g^{-1}] \otimes_{R_n} D_n/J^{\Delta}(f^s)|_{s=a}$. (Note that all localizations of R_n are automatically f-torsion free for $f \in R_n$ so that Algorithm 17 can be used.) This process may be iterated for products with any finite number of factors. Of course the exponents for the various factors might be different. This requires some care when setting up the Čech complex. In particular one needs to make sure that the maps $\check{C}^k \to \check{C}^{k+1}$ can be made explicit using the f_i. (In our Example 25, this is precisely how we proceeded when we found `Rfgh`!)

Remark 28. One might hope that for all holonomic fg-torsion free modules $M = D_n/L$ we have (with $M \otimes R_n[g^{-1}] \cong D_n/L'$):

$$a_f^L = \min\{s \in \mathbb{Z} : b_f^L(s) = 0\} \leq \min\{s \in \mathbb{Z} : b_f^{L'}(s) = 0\} = a_f^{L'}. \qquad (5.1)$$

This hope is unfounded. Let $R_5 = K[x_1, \dots, x_5]$, $f = x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2$. One may check that then $b_f^{\Delta}(s) = (s+1)(s+5/2)$. Hence $R_5[f^{-1}] = D_5 \bullet f^{-1}$, let $L = \ker(D_5 \to D_5 \bullet f^{-1})$. Set $g = x_1$. Then $b_g^{\Delta}(s) = s+1$, let $L' = \ker(D_5 \to D_5 \bullet g^{-1})$.

Then $b_f^{L'}(s) = (s+1)(s+2)(s+5/2)$ and $b_g^L(s) = (s+1)(s+3)$ because of the following computations.

```
i68 : erase symbol x;

i69 : erase symbol Dx;
```

These two commands essentially clear the history of the variables `x` and `Dx` and make them available for future computations.

```
i70 : D = QQ[x_1..x_5, Dx_1..Dx_5, WeylAlgebra =>
           apply(toList(1..5), i -> x_i => Dx_i)];

i71 : f = x_1^2 + x_2^2 + x_3^2 + x_4^2 +x_5^2;

i72 : g = x_1;

i73 : R = D^1/ideal(Dx_1,Dx_2,Dx_3,Dx_4,Dx_5);
```

As usual, these commands defined the base ring, two polynomials and the D_5-module R_5. Now we compute the respective localizations.

```
i74 : Rf =DlocalizeAll(R,f,Strategy => Oaku);
localize: Warning: Oaku's strategy requires
             module to be f-saturated

i75 : Bf = Rf.Bfunction

                          5
o75 = ($s + 1)($s + -)
                          2

o75 : Product
```

```
i76 : Rfg = DlocalizeAll(Rf.LocModule,g,Strategy => Oaku);
localize: Warning: Oaku's strategy requires
             module to be f-saturated

i77 : Bfg = Rfg.Bfunction

o77 = ($s + 3)($s + 1)

o77 : Product

i78 : Rg = DlocalizeAll(R,g,Strategy => Oaku);
localize: Warning: Oaku's strategy requires
             module to be f-saturated

i79 : Bg = Rg.Bfunction

o79 = ($s + 1)

o79 : Product

i80 : Rgf = DlocalizeAll(Rg.LocModule,f,Strategy => Oaku);
localize: Warning: Oaku's strategy requires
             module to be f-saturated

i81 : Bgf = Rgf.Bfunction

             5
o81 = ($s + -)($s + 2)($s + 1)
             2

o81 : Product
```

The output shows that $R_n[(fg)^{-1}]$ is generated by $f^{-2}g^{-1}$ or $f^{-1}g^{-3}$ but not by $f^{-1}g^{-2}$ and in particular not by $f^{-1}g^{-1}$. This can be seen from the various Bernstein-Sato polynomials: as for example the smallest integral root of Bf is -1 and that of Bfg is -3, $R_3[f^{-1}]$ is generated by f^{-1} and $R_3[(fg)^{-1}]$ by $f^{-1}g^{-3}$. This example not only disproves the above inequality (5.1) but also shows the inequality to be wrong if $\mathbb{Z}$ is replaced by $\mathbb{R}$ (as $-3 < \min(-5/2, -1)$).

Nonetheless, localizing $R_n[(fg)^{-1}]$ as $(R_n[f^{-1}])[g^{-1}]$ is heuristically advantageous, apparently for two reasons. For one, it allows the exponents of the various factors to be distinct which is useful for the subsequent cohomology computation: it helps to keep the degrees of the maps small. So in Example 25 we can write $R_6[(fg)^{-1}]$ as $D_6 \bullet (f^{-1}g^{-2})$ instead of $D_6 \bullet (fg)^{-2}$. Secondly, since the computation of Gröbner bases is potentially doubly exponential it seems to be advantageous to break a big problem (localization at a product) into several "easy" problems (successive localization).

An interesting case of this behaviour is our Example 25. If we compute $R_n[(fgh)^{-1}]$ as $((R_n[f^{-1}])[g^{-1}])[h^{-1}]$, the calculation uses approximately 6MB and lasts a few seconds using *Macaulay 2*. If one tries to localize R_n at the product of the three generators at once, *Macaulay 2* runs out of memory on all machines the author has tried this computation on.

5.2 Projects of the Future

This is a list of theoretical and implementational questions that the author finds important and interesting.

Prime characteristic. In [25], G. Lyubeznik gave an algorithm for deciding whether or not $H^i_I(R) = 0$ for any given $I \subseteq R = K[x_1, \ldots, x_n]$ where K is a computable field of positive characteristic. His algorithm is built on entirely different methods than the ones used in this chapter and relies on the Frobenius functor. The implementation of this algorithm would be quite worthwhile.

Ambient spaces different from $\mathbb{A}^n_K$. If A equals $K[x_1, \ldots, x_n]$, $I \subseteq A$, $X = \mathrm{Spec}(A)$ and $Y = \mathrm{Spec}(A/I)$, knowledge of $H^i_I(A)$ for all $i \in \mathbb{N}$ answers of course the question about the local cohomological dimension of Y in X. If $W \subseteq X$ is a smooth variety containing Y then Algorithm 24 for the computation of $H^i_I(A)$ also leads to a determination of the local cohomological dimension of Y in W. Namely, if J stands for the defining ideal of W in X so that $R = A/J$ is the affine coordinate ring of W and if we set $c = \mathrm{ht}(J)$, then it can be shown that $H^{i-c}_I(R) = \mathrm{Hom}_A(R, H^i_I(A))$ for all $i \in \mathbb{N}$. As $H^i_I(A)$ is I-torsion (and hence J-torsion), $\mathrm{Hom}_A(R, H^i_I(A))$ is zero if and only if $H^i_I(A) = 0$. It follows that the local cohomological dimension of Y in W equals $\mathrm{cd}(A, I) - c$ and in fact $\{i \in \mathbb{N} : H^i_I(A) \neq 0\} = \{i \in \mathbb{N} : H^{i-c}_I(R) \neq 0\}$.

If however $W = \mathrm{Spec}(R)$ is not smooth, no algorithms for the computation of either $H^i_I(R)$ or $\mathrm{cd}(R, I)$ are known, irrespective of the characteristic of the base field. It would be very interesting to have even partial ideas for computations in that case.

De Rham cohomology. In [33,47] algorithms are given to compute de Rham (in this case equal to singular) cohomology of complements of complex affine hypersurfaces and more general varieties. In [49] an algorithm is given to compute the multiplicative (cup product) structure, and in [48] the computation of the de Rham cohomology of open and closed sets in projective space is explained. Some of these algorithms have been implemented while others are still waiting.

For example, de Rham cohomology of complements of hypersurfaces, and partially the cup product routine, are implemented.

Example 29. Let $f = x^3 + y^3 + z^3$ in R_3. One can compute with *Macaulay 2* the de Rham cohomology of the complement of $\mathrm{Var}(f)$, and it turns out that the cohomology in degrees 0 and 1 is 1-dimensional, in degrees 3 and 4 2-dimensional and zero otherwise – here is the input:

```
i82 : erase symbol x;
```

Once `x` gets used as a subscripted variable, it's hard to use it as a nonsubscripted variable. So let's just erase it.

```
i83 : R = QQ[x,y,z];

i84 : f=x^3+y^3+z^3;

i85 : H=deRhamAll(f);
localize: Warning: Oaku's strategy requires
            module to be f-saturated
```

`H` is a hashtable with the entries `Bfunction`, `LocalizeMap`, `VResolution`, `TransferCycles`, `PreCycles`, `OmegaRes` and `CohomologyGroups`. For example, we have

```
i86 : H.CohomologyGroups

                      1
o86 = HashTable{0 => QQ }
                      1
               1 => QQ
                      2
               2 => QQ
                      2
               3 => QQ

o86 : HashTable
```

showing that the dimensions are as claimed above. One can also extract information on the generator of $R_3[f^{-1}]$ used to represent the cohomology classes by

```
i87 : H.LocalizeMap

o87 = {0} | $x_1^6+2$x_1^3$x_2^3+$x_2^6+2$x_1^3$x_3^3+2$x_2^3$x_3^3+$x ···

o87 : Matrix
```

which proves that the generator in question is f^{-2}. The cohomology classes that *Macaulay 2* computes are differential forms:

```
i88 : H.TransferCycles

o88 = HashTable{0 => {0} | -1/12$x_1^4$x_2^3$D_1-1/3$x_1$x_2^6$D_1-1/1 ···
               1 => {0} | 2/3$x_1^5+2/3$x_1^2$x_2^3+2/3$x_1^2$x_3^3   ···
                    {0} | -2/3$x_1^3$x_2^2-2/3$x_2^5-2/3$x_2^2$x_3^3  ···
                    {0} | 2/3$x_1^3$x_3^2+2/3$x_2^3$x_3^2+2/3$x_3^5   ···
               2 => {0} | 48$x_1$x_2$x_3^2 600$x_3^4      |
                    {0} | 48$x_1$x_2^2$x_3 600$x_2$x_3^3 |
                    {0} | 48$x_1^2$x_2$x_3 600$x_1$x_3^3 |
               3 => {0} | -$x_1$x_2$x_3 -$x_3^3 |

o88 : HashTable
```

So, for example, the left column of the three rows that correspond to $H^2_{\mathrm{dR}}(\mathbb{C}^3 \setminus \mathrm{Var}(f), \mathbb{C})$ represent the form $\frac{xyz(48z\,dx\,dy+48y\,dz\,dx+48x\,dy\,dz)}{f^2}$. If we apply the above elements of D_3 to f^{-2} and equip the results with appropriate differentials one arrives (after dropping unnecessary integral factors) at

Group	Dimension	Generators
H^0_{dR}	1	$e := \frac{f^2}{f^2}$
H^1_{dR}	1	$o := \frac{(x^2dx-y^2dy+z^2dz)f}{f^2}$
H^2_{dR}	2	$t_1 := \frac{xyz(zdxdy+ydzdx+xdydz)}{f^2}$ $t_2 := \frac{(zdxdy+ydzdx+xdydz)z^3}{f^2}$
H^3_{dR}	2	$d_1 := \frac{xyz\,dxdydz}{f^2}$ $d_2 := \frac{z^3\,dxdydz}{f^2}$

In terms of de Rham cohomology, the cup product is given by the wedge product of differential forms. So from the table one reads off the cup product relations $o \cup t_1 = d_1$, $o \cup t_2 = d_2$, $o \cup o = 0$, and that e operates as the identity.

For more involved examples, and the algorithms in [48], an actual implementation would be necessary since paper and pen are insufficient tools then.

Remark 30. The reader should be warned: if f^{-a} generates $R_n[f^{-1}]$ over D_n, then it is not necessarily the case that each de Rham cohomology class of $U = \mathbb{C}^n \setminus \mathrm{Var}(f)$ can be written as a form with a pole of order at most a. A counterexample is given by $f = (x^3 + y^3 + xy)xy$, where $H^1_{\mathrm{dR}}(U;\mathbb{C})$ has a class that requires a third order pole, although -2 is the smallest integral Bernstein root of f on R_2.

Hom and Ext. In [35,43,44] algorithms are explained that compute homomorphisms between holonomic systems. In particular, rational and polynomial solutions can be found because, for example, a polynomial solution to the system $\{P_1, \ldots, P_r\} \in D_n$ corresponds to an element of $\mathrm{Hom}_{D_n}(D_n/I, R_n)$ where $I = D_n \cdot (P_1, \ldots, P_r)$.

Example 31. Consider the GKZ system in 2 variables associated to the matrix $(1, 2) \in \mathbb{Z}^{1\times 2}$ and the parameter vector $(5) \in \mathbb{C}^1$. Named after Gelfand-Kapranov-Zelevinski [11], this is the following system of differential equations:

$$(x\partial_x + y\partial_y) \bullet f = 5f,$$
$$(\partial_x^2 - \partial_y) \bullet f = 0.$$

With *Macaulay 2* one can solve systems of this sort as follows:

```
i89 : I = gkz(matrix{{1,2}}, {5})

                2
o89 = ideal (D  - D , x D  + 2x D  - 5)
              1    2   1 1     2 2

o89 : Ideal of QQ [x , x , D , D , WeylAlgebra => {x  => D , x  => D } ···
                    1   2   1   2                  1     1   2     2
```

This is a simple command to set up the GKZ-ideal associated to a matrix and a parameter vector. The polynomial solutions are obtained by

```
i90 : PolySols I

        5       3         2
o90 = {x  + 20x x  + 60x x }
        1      1 2      1 2

o90 : List
```

This means that there is exactly one polynomial solution to the given GKZ-system, and it is

$$x^5 + 20x^3y + 60xy^2.$$

The algorithm for $\mathrm{Hom}_{D_n}(M, N)$ is implemented and can be used to check whether two given D-modules are isomorphic. Moreover, there are algorithms (not implemented yet) to compute the ring structure of $\mathrm{End}_D(M)$ for a given D-module M of finite holonomic rank which can be used to split a given holonomic module into its direct summands. Perhaps an adaptation of these methods can be used to construct Jordan-Hölder sequences for holonomic D-modules.

Finiteness and stratifications. Lyubeznik pointed out in [26] the following curious fact.

Theorem 32. *Let $P(n,d;K)$ denote the set of polynomials of degree at most d in at most n variables over the field K of characteristic zero. Let $B(n,d;K)$ denote the set of Bernstein-Sato polynomials*

$$B(n,d;K) = \{b_f(s) : f \in P(n,d;K)\}.$$

Then $B(n,d;K)$ is finite. $\square$

So $P(n,d;K)$ has a finite decomposition into strata with constant Bernstein-Sato polynomial. A. Leykin proved in [22] that this decomposition is independent of K and computable in the sense that membership in each stratum can be tested by the vanishing of a finite set of algorithmically computable polynomials over $\mathbb{Q}$ in the coefficients of the given polynomial in $P(n,d;K)$. In particular, the stratification is algebraic and for each K induced by base change from $\mathbb{Q}$ to K. It makes thus sense to define $B(n,d)$ which is the finite set of Bernstein polynomials that can occur for $f \in P(n,d;K)$ (where K is in fact irrelevant).

Example 33. Consider $P(2,2;K)$, the quadratic binary forms over K. With *Macaulay 2*, Leykin showed that there are precisely 4 different Bernstein polynomials possible:

- $b_f(s) = 1$ iff $f \in V_1 = V_1' \setminus V_1''$, where $V_1' = V(a_{1,1}, a_{0,1}, a_{0,2}, a_{1,0}, a_{2,0})$, while $V_1'' = V(a_{0,0})$
- $b_f(s) = s+1$ iff $f \in V_2 = (V_2' \setminus V_2'') \cup (V_3' \setminus V_3'')$, where $V_2' = V(0)$, $V_2'' = V(\gamma_1)$, $V_3' = V(\gamma_2, \gamma_3, \gamma_4)$, $V_3'' = V(\gamma_3, \gamma_4, \gamma_5, \gamma_6, \gamma_7, \gamma_8)$,
- $b_f(s) = (s+1)^2$ iff $f \in V_4' \setminus V_4''$, where $V_4' = V(\gamma_1)$, $V_4'' = V(\gamma_2, \gamma_3, \gamma_4)$,
- $b_f(s) = (s+1)(s+\frac{1}{2})$ iff $f \in V_5' \setminus V_5''$, where $V_5' = V(\gamma_3, \gamma_4, \gamma_5, \gamma_6, \gamma_7, \gamma_8)$, $V_5'' = V(a_{1,1}, a_{0,1}, a_{0,2}, a_{1,0}, a_{2,0})$.

Here we have used the abbreviations

- $\gamma_1 = a_{0,2}a_{1,0}^2 - a_{0,1}a_{1,0}a_{1,1} + a_{0,0}a_{1,1}^2 + a_{0,1}^2 a_{2,0} - 4a_{0,0}a_{0,2}a_{2,0}$,
- $\gamma_2 = 2a_{0,2}a_{1,0} - a_{0,1}a_{1,1}$,
- $\gamma_3 = a_{1,0}a_{1,1} - 2a_{0,1}a_{2,0}$,
- $\gamma_4 = a_{1,1}^2 - 4a_{0,2}a_{2,0}$,
- $\gamma_5 = 2a_{0,2}a_{1,0} - a_{0,1}a_{1,1}$,
- $\gamma_6 = a_{0,1}^2 - 4a_{0,0}a_{0,2}$,
- $\gamma_7 = a_{0,1}a_{1,0} - 2a_{0,0}a_{1,1}$,
- $\gamma_8 = a_{1,0}^2 - 4a_{0,0}a_{2,0}$.

Similarly, Leykin shows that there are 9 possible Bernstein polynomials for $f \in B(2,3;K)$:

$$
\begin{aligned}
B(2,3) = \Big\{ & (s+1)^2(s+\frac{2}{3})(s+\frac{4}{3}),\ (s+1)^2(s+\frac{1}{2}),\ (s+1),\ 1, \\
& (s+1)(s+\frac{2}{3})(s+\frac{1}{3}),\ (s+1)^2,\ (s+1)(s+\frac{1}{2}), \\
& (s+1)(s+\frac{7}{6})(s+\frac{5}{6}),\ (s+1)^2(s+\frac{3}{4})(s+\frac{5}{4}) \Big\}.
\end{aligned}
$$

It would be very interesting to study the nature of the stratification in larger cases, and its restriction to hyperplane arrangements.

A generalization of this stratification result is obtained in [50]. There it is shown that there is an algorithm to give $P(n,d;K)$ an algebraic stratification defined over $\mathbb{Q}$ such that the algebraic de Rham cohomology groups of the complement of $\mathrm{Var}(f)$ do not vary on the stratum in a rather strong sense. Again, the study and explicit computation of this stratification should be very interesting.

Hodge numbers. If Y is a projective variety in $\mathbb{P}^n_{\mathbb{C}}$ then algorithms outlined in [48] show how to compute the dimensions not only of the de Rham cohomology groups of $\mathbb{P}^n_{\mathbb{C}} \setminus Y$ but also of Y itself. Suppose now that Y is in fact a smooth projective variety. An interesting set of invariants are the Hodge numbers, defined by $h^{p,q} = \dim H^p(Y, \Omega^q)$, where Ω^q denotes the sheaf of $\mathbb{C}$-linear differential q-forms with coefficients in $\mathcal{O}_Y$. At present we do not know how to compute them. Of course there is a spectral sequence $H^p(Y,\Omega^q) \Rightarrow H^{p+q}_{\mathrm{dR}}(Y,\mathbb{C})$ and we know the abutment (or at least its dimensions), but the technique for computing the abutment does not seem to be usable to compute the E^1 term because on an affine patch $H^p(Y,\Omega^q)$ is either zero or an infinite dimensional vector space.

Hogde structures and Bernstein-Sato polynomials are related as is for example shown in [45].

5.3 Epilogue

In this chapter we have only touched a few highlights of the theory of computations in D-modules, most of them related to homology and topology. Despite this we harldy touched on the topics of integration and restriction, which are the D-module versions of a pushforward and pullback, [20,29,33,47].

A very different aspect of D-modules is discussed in [39] where at the center of investigations is the combinatorics of solutions of hypergeometric differential equations. The combinatorial structure is used to find series solutions for the differential equations which are polynomial in certain logarithmic functions and power series with respect to the variables.

Combinatorial elements can also be found in the work of Assi, Castro and Granger, see [2,3], on Gröbner fans in rings of differential operators. An important (open) question in this direction is the determination of the set of ideals in D_n that are initial ideals under *some* weight.

Algorithmic D-module theory promises to be an active area of research for many years to come, and to have interesting applications to various other parts of mathematics.

References

1. A.G. Aleksandrov and V.L. Kistlerov. Computer method in calculating b-functions of non-isolated singularities. *Contemp. Math.*, 131:319–335, 1992.
2. A. Assi, F.J. Castro-Jiménez, and M. Granger. How to calculate the slopes of a $\mathcal{D}$-module. *Compositio Math.*, 104(2):107–123, 1996.
3. A. Assi, F.J. Castro-Jiménez, and M. Granger. The Gröbner fan of an A_n-module. *J. Pure Appl. Algebra*, 150(1):27–39, 2000.
4. J. Bernstein. Lecture notes on the theory of D-modules. *Unpublished.*
5. J.-E. Björk. *Rings of Differential Operators.* North Holland, New York, 1979.
6. J. Briançon, M. Granger, Ph. Maisonobe, and M. Miniconi. Algorithme de calcul du polynôme de Bernstein: cas nondégénéré. *Ann. Inst. Fourier*, 39:553–610, 1989.
7. M. Brodmann and J. Rung. Local cohomology and the connectedness dimension in algebraic varieties. *Comment. Math. Helvetici*, 61:481–490, 1986.
8. M.P. Brodmann and R.Y. Sharp. *Local Cohomology: an algebraic introduction with geometric applications.* Cambridge Studies in Advanced Mathematics, 60. Cambridge University Press, 1998.
9. S.C. Coutinho. *A primer of algebraic D-modules.* London Mathematical Society Student Texts 33. Cambridge University Press, Cambridge, 1995.
10. R. Garcia and C. Sabbah. Topological computation of local cohomology multiplicities. *Coll. Math.*, XLIX(2-3):317–324, 1998.
11. I. Gelfand, M. Kapranov, and A. Zelevinski. *Discriminants, resultants, and multidimensional determinants.* Mathematics: Theory & Applications. Birkhäuser Boston, Inc., Boston, MA, 1994.
12. P. Griffiths and J. Harris. *Principles of Algebraic Geometry.* A Wiley Interscience Publication. WILEY and Sons, 1978.
13. R. Hartshorne. *Local Cohomology. A seminar given by A. Grothendieck.* LNM 41. Springer Verlag, 1967.
14. R. Hartshorne. Cohomological Dimension of Algebraic Varieties. *Ann. Math.*, 88:403–450, 1968.
15. R. Hartshorne. On the de Rham Cohomology of Algebraic Varieties. *Publ. Math. Inst. Hautes Sci.*, 45:5–99, 1975.
16. R. Hartshorne and R. Speiser. Local cohomological dimension in characteristic p. *Ann. Math.(2)*, 105:45–79, 1977.
17. C. Huneke. Problems on local cohomology. *Res. Notes Math.*, 2:93–108, 1992.
18. C. Huneke and G. Lyubeznik. On the vanishing of local cohomology modules. *Invent. Math.*, 102:73–93, 1990.
19. M. Kashiwara. B-functions and holonomic systems, rationality of B-functions. *Invent. Math.*, 38:33–53, 1976.

20. M. Kashiwara. On the holonomic systems of linear partial differential equations, II. *Invent. Math.*, 49:121–135, 1978.
21. J. Kollár. Singularities of pairs, *in* Algebraic Geometry, Santa Cruz, 1995. *Proc. Symp. Pure Math. Amer. Math. Soc.*, 62:221–287, 1997.
22. A. Leykin. Towards a definitive computation of Bernstein-Sato polynomials. `math.AG/0003155`[1], 2000.
23. A. Leykin, M. Stillman, and H. Tsai. The D-module package for *Macaulay 2*. `http://www.math.umn.edu/~leykin`.
24. G. Lyubeznik. Finiteness Properties of Local Cohomology Modules: an Application of D-modules to Commutative Algebra. *Invent. Math.*, 113:41–55, 1993.
25. G. Lyubeznik. F-modules: applications to local cohomology and D-modules in characteristic $p > 0$. *Journal für die Reine und Angewandte Mathematik*, 491:65–130, 1997.
26. G. Lyubeznik. On Bernstein-Sato polynomials. *Proc. Amer. Math. Soc.*, 125(7):1941–1944, 1997.
27. Ph. Maisonobe. D-modules: an overview towards effectivity, in: E. Tournier, ed., Computer Algebra and Differential Equations. *London Math. Soc. Lecture Note Ser., 193, Cambridge Univ. Press, Cambridge*, pages 21–55, 1994.
28. B. Malgrange. Le polynôme de Bernstein d'une singularité isolée. *Lecture Notes in Mathematics, Springer Verlag*, 459:98–119, 1975.
29. Z. Mebkhout. *Le formalisme des six opérations de Grothendieck pour les D_X-modules cohérents.* Travaux en Cours. Hermann, Paris, 1989.
30. T. Oaku. An algorithm for computing B-functions. *Duke Math. Journal*, 87:115–132, 1997.
31. T. Oaku. Algorithms for b-functions, restrictions, and algebraic local cohomology groups of D-modules. *Advances in Appl. Math.*, 19:61–105, 1997.
32. T. Oaku. Algorithms for the b-function and D-modules associated with a polynomial. *J. Pure Appl. Algebra*, 117-118:495–518, 1997.
33. T. Oaku and N. Takayama. Algorithms for D-modules – restriction, tensor product, localization and local cohomology groups. `math.AG/9805006`, 1998.
34. T. Oaku and N. Takayama. An algorithm for de Rham cohomology groups of the complement of an affine variety via D-module computation. *J. Pure Appl. Algebra*, 139:201–233, 1999.
35. T. Oaku, N. Takayama, and H. Tsai. Polynomial and rational solutions of holonomic systems. `math.AG/0001064`, 1999.
36. T. Oaku, N. Takayama, and U. Walther. A localization algorithm for D-modules. *J. Symb. Comp.*, 29(4/5):721–728, May 2000.
37. A. Ogus. Local cohomological dimension of algebraic varieties. *Ann. Math. (2)*, 98:327–365, 1973.
38. C. Peskine and L. Szpiro. Dimension projective finie et cohomologie locale. *Inst. Hautes Études Sci. Publ. Math.*, 42:47–119, 1973.
39. M. Saito, B. Sturmfels, and N. Takayama. *Gröbner deformations of hypergeometric differential equations.* Algorithms and Computation in Mathematics, 6. Springer Verlag, 1999.
40. M. Sato, M. Kashiwara, T. Kimura, and T. Oshima. Micro-local analysis of prehomogeneous vector spaces. *Inv. Math.*, 62:117–179, 1980.

[1] The webpage `http://xxx.lanl.gov/` is a page designed for the storage of preprints, and allows posting and downloading free of charge.

41. N. Takayama. Kan: A system for computation in algebraic analysis. Source code available at `www.math.kobe-u.ac.jp/KAN`. *Version 1 (1991), Version 2 (1994)*, the latest Version is 2.990914 (1999).
42. H. Tsai. Weyl closure, torsion, and local cohomology of D-modules. *Preprint*, 1999.
43. H. Tsai. Algorithms for algebraic analysis. *Thesis, University of California at Berkeley*, 2000.
44. H. Tsai and U. Walther. Computing homomorphisms between holonomic D-modules. `math.RA/0007139`, 2000.
45. A. Varchenko. Asymptotic Hodge structure in the vanishing cohomology. *Math. USSR Izvestija*, 18(3):469–512, 1982.
46. U. Walther. Algorithmic Computation of Local Cohomology Modules and the Local Cohomological Dimension of Algebraic Varieties. *J. Pure Appl. Algebra*, 139:303–321, 1999.
47. U. Walther. Algorithmic Computation of de Rham Cohomology of Complements of Complex Affine Varieties. *J. Symb. Comp.*, 29(4/5):795–839, May 2000.
48. U. Walther. Cohomology with Rational Coefficients of Complex Varieties. *Preprint*, 2000.
49. U. Walther. The Cup Product Structure for Complements of Complex Affine Varieties. To appear in *J. Pure Appl. Algebra*, 2000.
50. U. Walther. Homotopy Type, Stratifications and Gröbner bases. *In preparation*, 2000.
51. U. Walther. On the Lyubeznik numbers of a local ring. *Proc. Amer. Math. Soc.*, (To appear) 2001.
52. T. Yano. On the thory of b-functions. *Publ. RIMS, Kyoto Univ.*, 14:111–202, 1978.