OreModules: A symbolic package for the study of multidimensional linear systems

F. Chyzak[1], A. Quadrat[2], and D. Robertz[3]

[1] INRIA Rocquencourt, ALGO project, Domaine de Voluceau BP 105, 78153 Le Chesnay Cedex, France. frederic.chyzak@inria.fr.
[2] INRIA Sophia Antipolis, CAFE project, 2004, Route des Lucioles BP 93, 06902 Sophia Antipolis Cedex, France. Alban.Quadrat@sophia.inria.fr.
[3] Lehrstuhl B für Mathematik, RWTH - Aachen, Templergraben 64, 52056 Aachen, Germany. daniel@momo.math.rwth-aachen.de.

1 Introduction

In the seventies, the study of transfer matrices of time-invariant linear systems of ordinary differential equations (ODEs) led to the development of the *polynomial approach* [20, 22, 44]. In particular, the univariate polynomial matrices play a central role in this approach (e.g., Hermite, Smith and Popov forms, invariant factors, primeness, Bézout/Diophantine equations).

In the middle of the seventies, while generalizing linear systems defined by ODEs to differential time-delay systems, ODEs with parameters, 2-D and 3-D filters…, one had to face the case of systems described by means of matrices with entries in multivariate commutative polynomial rings. All these new systems were called 2-D or 3-D linear systems and, more generally, *n-D systems* or *multidimensional linear systems* with constant coefficients [4, 16]. It was quickly realized that no canonical forms such as Hermite, Smith and Popov forms existed for polynomial matrices with two and three variables (i.e., with entries in $k[x_1, x_2, x_3]$, where k is a field such as $\mathbb{Q}$, $\mathbb{R}$, $\mathbb{C}$). Moreover, more than only one type of primeness was needed in order to classify n-D systems (e.g., factor/minor/zero primeness [48, 49]). Hence, it is not very surprising that, in the eighties, *Gröbner bases* were introduced in the study of multidimensional linear systems with constant coefficients [4, 16]. A Gröbner basis defines normal forms for polynomials with respect to a certain monomial ordering of the variables x_i [2, 17, 23]. Given a Gröbner basis, there is a simple algorithm to effectively compute these normal forms. In many ways, the computation of these normal forms can be seen as an extension of the Gaussian elimination algorithm to commutative polynomial rings [2, 17].

In a pioneering work, R. E. Kalman developed a *module-theoretic* approach to time-invariant ordinary differential linear systems [21]. In his PhD thesis under the supervision of R. E. Kalman, Y. Rouchaleau considered Kalman-type systems where the entries of (A, B, C, D) belong to a commutative ring. In particular, he studied their structural properties using module theory. Such systems are nowadays called *systems over rings* and they have been considerably studied in the literature [43] since. An extension of the *geometric approach* [46] to linear systems over rings has also been recently developed [1, 12, 18, 19]. Using effective algebra methods (Gröbner bases, *characteristic sets*), the computational aspects of the systems over rings (e.g., differential time-delay systems) were firstly studied by L. Habets in [18, 19].

In the nineties, U. Oberst developed a general module-theoretic approach to multidimensional linear systems with constant coefficients [28]. Using B. Malgrange's approach [24], in which a *finitely presented D-module M* is associated with a linear system of equations over a polynomial ring D, he showed how some structural properties of the system corresponded to algebraic properties of the D-module M. He then was able to develop a complete duality between his module-theoretic approach and the *behavioural approach* developed by J. C. Willems [30]. Based on U. Oberst's ideas, the behavioural approach to multidimensional linear systems has been successfully developed in the recent years. See [30, 29, 36, 47, 49] and the references therein.

Within a similar module-theoretic approach, the concepts of *flatness* and *π-freeness* were introduced in [15, 26] for differential time-delay linear systems with constant coefficients. As it is shown in [26, 27] on different concrete examples, the detection of such structural properties is important for the study of the motion planning problem. In the behavioural approach, the concept of flatness corresponds to the existence of an *observable image representation* for the multidimensional system [32].

In the same years as [28], J.-F. Pommaret studied underdetermined systems of partial differential equations (PDEs) coming from mathematical physics and differential geometry (e.g., elasticity, electromagnetism, hydrodynamics, general relativity). See also [3]. In particular, he showed how his mathematical approach was a generalization of U. Oberst's module-theoretic approach for multidimensional (linear) systems with varying coefficients. See [31] for more details and references. In particular, the problem of checking whether or not a multidimensional linear system described by PDEs with varying coefficients could be formally parametrized was solved within the theory of differential operators. Moreover, the work of M. Fliess on linear systems defined by ODEs with variable coefficients also illustrated the need to pass from the commutative polynomial viewpoint to the non-commutative one [14].

Based on B. Malgrange's approach [24], *algebraic analysis* has been developed in mathematics in order to study general linear systems of PDEs with variable coefficients using module theory, algebraic geometry, homological algebra and functional analysis. Algebraic analysis has recently been introduced in control theory in [38] in order to study multidimensional linear systems defined by PDEs with varying coefficients. In particular, using the formal theories of PDEs (Spencer's, Riquier-Janet's theories), it was shown in

[31, 32, 33, 34, 38] how some structural properties of systems could be checked by means of constructive algorithms.

Finally, using the homological algebra approach developed in [38], we have recently shown in [9, 11] how the previous results could be generalized to some classes of multidimensional linear systems with varying coefficients encountered in the literature (e.g., ODEs, PDEs, differential time-delay systems, multidimensional discrete systems, partial differential delay systems). In order to do that, the concept of multidimensional linear systems over *Ore algebras* was introduced in [9, 11]. An Ore algebra is a ring of non-commutative polynomials in functional operators with polynomial or rational coefficients [5, 6, 7]. Characterizations of algebraic structural properties such as, for instance, controllability, parametrizability and flatness were obtained.

The recent progress of Gröbner bases over Ore algebras (i.e., over some classes of non-commutative polynomial rings) [5, 6, 7, 23] allows us to effectively test the algebraic properties of general multidimensional linear systems (e.g., controllability, observability, parametrizability, flatness, π-freeness) and compute different types of parametrizations and to propose feedback laws (motion planning, tracking, Bézout equations, optimal control).

In this paper, we shall develop the following methodology for the study of multidimensional linear systems over Ore algebras (see also [11]):

1. A linear system is defined by means of a $(q \times p)$-matrix R with entries in an Ore algebra D, i.e., it corresponds to a system of linear equations $R\,z = 0$, where z is composed of the system variables (see Section 2).
2. We associate the finitely presented left D-module $M = D^{1\times p}/(D^{1\times q}\,R)$ with the system $R\,z = 0$.
3. We develop a dictionary between the structural properties of the system and the properties of the left D-module M. Using module theory, we can then classify the properties of the left D-module M (see Section 3).
4. Homological algebra permits to check these properties of the left D-module M using *extension* and *torsion functors* (see Section 4).
5. Gröbner bases over Ore algebras allow to develop effective algorithms which check the properties of the left D-module M, and thus, of the system $R\,z = 0$ (see Section 5).
6. Implementations of these algorithms in the package OREMODULES for the computer algebra system Maple (see Section 6).

The purpose of this paper is to give an introduction to the package OREMODULES [8] for Maple which offers symbolic methods to investigate the structural properties of multidimensional linear systems over Ore algebras. The advantage of describing these properties in the language of homological algebra carries over to the implementation of OREMODULES: up to the choice of the domain of operators which occur in a given system, all algorithms are stated and implemented in sufficient generality such that ODEs, PDEs, differential time-delay systems, discrete systems with constant, polynomial or rational coefficients... are covered at the same time.

This paper is an extension of the congress paper [10].

2 Multidimensional linear systems over Ore algebras

The mathematical framework of this paper is built on the concept of *Ore algebras* [5, 6, 7]. Ore algebras are non-commutative polynomial rings that represent linear functional operators in a natural way.

We recall that a ring with a unit 1 is a *domain* if the product of non-zero elements is non-zero. In what follows, we shall denote by A a domain which has a k-algebra structure, where k is a field.

Definition 1. 1. [25] A *skew polynomial ring* $A[\partial; \sigma, \delta]$ is a non-commutative ring consisting of all polynomials in ∂ with coefficients in A obeying the commutation rule

$$\forall\, a \in A, \quad \partial\, a = \sigma(a)\, \partial + \delta(a), \tag{1}$$

where σ is a *k-algebra endomorphism of A*, namely, $\sigma : A \longrightarrow A$ satisfies

$$\sigma(1) = 1, \quad \forall\, a, b \in A, \quad \sigma(a+b) = \sigma(a) + \sigma(b), \quad \sigma(a\, b) = \sigma(a)\, \sigma(b),$$

and δ is a *σ-derivation of A*, namely, $\delta : A \longrightarrow A$ satisfies:

$$\forall\, a, b \in A, \quad \delta(a+b) = \delta(a) + \delta(b), \quad \delta(a\, b) = \sigma(a)\, \delta(b) + \delta(a)\, b.$$

2. [5, 7] Let $A = k[x_1, \ldots, x_n]$ be a commutative polynomial ring over a field k (if $n = 0$ then $A = k$). The skew polynomial ring

$$D = A[\partial_1; \sigma_1, \delta_1] \ldots [\partial_m; \sigma_m, \delta_m]$$

is called *Ore algebra* if the σ_i's and δ_j's commute for $1 \leq i, j \leq m$ and satisfy:

$$\sigma_i(\partial_j) = \partial_j, \quad \delta_i(\partial_j) = 0, \quad j < i.$$

Example 1. In order to model an ordinary differential linear system with polynomial coefficients, we use the *Weyl algebra* $A_1(k) = k[t][\partial; \sigma, \delta]$ which is a non-commutative k-algebra generated by t and ∂. Elements of $A_1(k)$ are non-commutative polynomials in t and ∂ with coefficients in the field k (e.g., $k = \mathbb{Q}, \mathbb{R}, \mathbb{C}$) satisfying the following commutation rule:

$$\forall\, a \in k[t], \quad \partial\,(a\, \cdot) = a\, \partial \cdot + \frac{da}{dt}\ \cdot .$$

Therefore, regarding Definition 1, we have $\sigma = \mathrm{id}_{k[t]}$ and $\delta = \frac{d}{dt}$.

More generally, for the study of partial differential linear systems, we shall use the *Weyl algebra* $A_n(k) = k[x_1, \ldots, x_n][\partial_1; \sigma_1, \delta_1] \ldots [\partial_n; \sigma_n, \delta_n]$, where σ_i and δ_i are the maps on $k[x_1, \ldots, x_n]$ defined by

$$\sigma_i = \mathrm{id}_{k[x_1,\ldots,x_n]}, \quad \delta_i = \frac{\partial}{\partial x_i}, \quad i = 1, \ldots, n,$$

and every other commutation rule is prescribed by Definition 1. We have:

$$\partial_i \, x_j = x_j \, \partial_i + \delta_{ij}, \quad 1 \leq i, j \leq n, \quad \text{where} \quad \delta_{ij} = 1 \;\text{ if } i = j \;\text{ and }\; 0 \;\text{ else.}$$

Example 2. The algebra of *shift operators* with polynomial coefficients is another special case of an Ore algebra. For h in the field k (e.g., $k = \mathbb{Q}$, $\mathbb{R}$), we define $S_h(k) = k[t][\delta_h; \sigma_h, \delta]$ by:

$$\forall \, a \in k[t], \quad \sigma_h(a)(t) = a(t-h), \quad \delta(a) = 0.$$

The commutation rule $\delta_h \, t = (t-h) \, \delta_h$ represents the action of the shift operator on polynomials. Forming equations over S_h, we model time-delay (resp., time-advance) systems if $h > 0$ (resp., $h < 0$).

Example 3. For differential time-delay linear systems, we mix the constructions of the two preceding examples. For $h \in k$ (e.g., $k = \mathbb{Q}$, $\mathbb{R}$), we define the Ore algebra $D_h(k) = k[t][\partial; \sigma_1, \delta_1][\delta_h; \sigma_2, \delta_2]$ where:

$$\sigma_1 = \mathrm{id}_{k[t]}, \quad \delta_1 = \frac{d}{dt}, \quad \forall \, a \in k[t], \quad \sigma_2(a)(t) = a(t-h), \quad \delta_2 = 0.$$

If the considered system also involves an advance operator, then we may work with the algebra defined by

$$H_{(h,l)}(k) = k[t][\partial; \sigma_1, \delta_1][\delta_h; \sigma_2, \delta_2][\tau_l; \sigma_3, \delta_3],$$

where σ_i, δ_i, $i = 1, 2$, are as above and:

$$\forall \, a \in k[t], \quad \sigma_3(a)(t) = a(t+l), \quad \delta_3 = 0, \quad l > 0.$$

Example 4. In order to study multidimensional discrete linear systems, we can define the following Ore algebra $k[z_1, \ldots, z_n][\partial_1; \sigma_1, \delta_1] \ldots [\partial_n; \sigma_n, \delta_n]$, where σ_i and δ_i, $i = 1, \ldots, n$, are the maps on $k[z_1, \ldots, z_n]$ defined by $\delta_i = 0$ and:

$$\forall \, a \in k[z_1, \ldots, z_n], \quad \sigma_i(a)(z_1, \ldots, z_n) = a(z_1, \ldots, z_{i-1}, z_i + 1, z_{i+1}, \ldots, z_n).$$

We refer to [7] for more examples of Ore algebras using for instance the difference, the divided differences or the q-dilation functional operators.

We can "concatenate" different Ore algebras in order to combine different types of functional operators and, by this means, we get Ore algebras for most of the linear systems commonly considered in control theory. Moreover, we can also use different rings of coefficients such as the field of rational functions or the ring of analytic functions. However, as we shall develop computational aspects, we only consider here polynomial or rational coefficients over $\mathbb{Q}$. Finally, we can prove that the algebras defined in Examples 1, 2, 3 and 4 are *left* and *right noetherian rings* (namely, every left/right ideal is generated by means of a finite number of elements). Thus, they have the *left* and *right Ore properties* (namely, for any pair (a_1, a_2) of elements, there exists a non-zero pair (b_1, b_2) (resp., (c_1, c_2)) such that $b_1\, a_1 = b_2\, a_2$ (resp., $a_1\, c_1 = a_2\, c_2$)) [5, 7, 11, 25].

Linear systems studied in control theory are generally defined by means of systems of ordinary or partial differential equations, time-delay equations, recurrence equations... These equations usually come from mathematical models. Hence, we can generally write a system as $R\, z = 0$, where R is a matrix with entries in a certain Ore algebra and z contains the system variables including the inputs, the outputs, the states, the latent variables...

Example 5.

- The linear system $P\left(\frac{d}{dt}\right) y = Q\left(\frac{d}{dt}\right) u$, where P and Q are two polynomial matrices in the differential operator $\frac{d}{dt}$ and with coefficients in $k[t]$, can be rewritten as $R\, z = 0$, where the entries of the matrix

$$R = \left(P\left(\frac{d}{dt}\right), \quad -Q\left(\frac{d}{dt}\right)\right)$$

belong to the Weyl algebra $A_1(k)$ and $z = (y^T, \, u^T)^T$.

- The differential time-delay linear system $\dot{x}(t) = A(t)\, x(t) + B(t)\, u(t-h)$, where A and B are two matrices with entries in $k[t]$ and $h > 0$, can be rewritten as $R\, z = 0$, where the entries of the matrix

$$R = \left(\frac{d}{dt}\, I - A(t), \quad -B(t)\, \delta_h\right)$$

belong to the Ore algebra $D_h(k)$ and $z = (x^T, \, u^T)^T$.

- The partial differential equation (heat equation)

$$\frac{\partial y(t,x)}{\partial t} = \frac{\partial}{\partial x}\left(a(x)\, \frac{\partial y(t,x)}{\partial x}\right) + u(t, \, x),$$

where the conductivity of the bar a is assumed to be polynomial in x, can be rewritten as $R\, z = 0$, where the entries of the matrix

$$R = \left(\frac{\partial}{\partial t} - \frac{\partial}{\partial x}\left(a(x)\, \frac{\partial}{\partial x}\right), \quad -1\right)$$

belong to the Weyl algebra $A_2(k)$ with $x_1 = t$, $x_2 = x$ and $z = (y, \, u)^T$.

Real systems are generally nonlinear ones, meaning that the theory developed in this paper is not directly applicable to these systems. However, using a linearization around a (generic/given) trajectory of the system, then the linearized system has varying coefficients. Therefore, we can examine the structural properties of the linearized system by means of the approach described here and use them to study the ones of the nonlinear system.

3 A module-theoretical approach to linear systems

In what follows, we denote by D an Ore algebra. The main idea of *algebraic analysis* is to study a linear system of the form $R\,z = 0$, where $R \in D^{q \times p}$, by means of the *finitely presented (f.p.) left D-module* $M = D^{1\times p}/(D^{1\times q}\,R)$ [24]. M is associated with $R\,z = 0$ in the sense that, if we denote by z_i the residue class in M of the row vector $e_i \in D^{1\times p}$ defined by 1 in the i^{th} position and 0 elsewhere and $z = (z_1, \ldots, z_p)^T$, then M is defined by all left D-linear combinations of the system equations $R\,z = 0$. See [11, 28, 31, 47] for more details.

The use of the residue class left D-module M is natural as it is a generalization of the construction of the algebras commonly studied in algebraic or analytic geometry and number theory (e.g., $\mathbb{C} = \mathbb{R}[x]/(x^2+1)$, $\mathbb{Z}[i\sqrt{5}] = \mathbb{Z}[x]/(x^2+5)$, $A = \mathbb{C}[x,y]/(x^2+y^2-1,\; x\,y-1))$ [17]. For instance, $A = \mathbb{C}[x,y]/I$, where I is the ideal $I = (x^2+y^2-1,\; x\,y-1)$ of $\mathbb{C}[x,y]$, can be defined by:

$$A = \mathbb{C}[x,y]/(\mathbb{C}[x,y]^{1\times 2}\,R), \quad R = \begin{pmatrix} x^2+y^2-1 \\ x\,y-1 \end{pmatrix} \in \mathbb{C}[x,y]^{2\times 1}.$$

The first main interest regarding the left D-module M instead of the system $R\,z = 0$ is that M is intrinsically well-defined in the sense that it does not depend on the choice of the representation $R\,z = 0$ of the system. Indeed, the same system can be represented in different equivalent forms having different numbers of unknowns and equations (e.g., state-space or input-output representations, Roesser or Fornasini-Marchesini models) [35, 38, 45].

The second main interest of using the finitely presented left D-module M is that we can classify the structural properties of the system by means of the module properties of M. We introduce a few definitions [25, 45].

Definition 2. Let M be a finitely generated left module over a left noetherian domain D. Then, we have the following definitions:

1. M is *free* if there exists $r \in \mathbb{Z}_+$ such that M is isomorphic to $D^{1\times r}$, a fact that we denote by $M \cong D^{1\times r}$.
2. M is *stably free* if there exist $r,\ s \in \mathbb{Z}_+$ such that $M \oplus D^{1\times s} \cong D^{1\times r}$, where $\oplus$ denotes the direct sum.
3. M is *projective* if there exist a left D-module N and $r \in \mathbb{Z}_+$ such that we have $M \oplus N \cong D^{1\times r}$. Then, the left D-module N is also projective.
4. M is *reflexive* if the following canonical D-morphism (i.e., D-linear map)

$$\varepsilon_M : M \longrightarrow \hom_D(\hom_D(M,D),D), \quad \varepsilon_M(m)(f) = f(m),$$

   $-$ where $m \in M$ and f belongs to the right D-module $\hom_D(M,D)$ formed by the left D-morphisms from M to D $-$ is an isomorphism (i.e., ε_M is both injective and surjective).

5. M is *torsion-free* if the left D-submodule

$$t(M) = \{m \in M \mid \exists\, 0 \neq P \in D,\ P\, m = 0\}$$

   of M is reduced to 0. $t(M)$ is called the *torsion left D-submodule* of M and the elements of $t(M)$ are the *torsion elements* of M.
6. M is *torsion* if $t(M) = M$.

We have the following important results [25, 45].

Theorem 1. *1. We have the following implications of module properties:*

$$\textit{free} \Rightarrow \textit{stably free} \Rightarrow \textit{projective} \Rightarrow \textit{reflexive} \Rightarrow \textit{torsion-free.}$$

2. Every torsion-free left module over $A_1(k)$ (resp., $k\left[\frac{d}{dt}\right]$, $k(t)\left[\frac{d}{dt}\right]$) is stably free (resp., free).
3. Every projective module over the commutative polynomial ring $k[x_1, \ldots, x_n]$ over a field k is free (Quillen-Suslin theorem).

In the recent years, a classification of properties of multidimensional linear systems has been established in terms of the properties of the corresponding left D-module M. Let us summarize some of them in Table 1. We refer the reader to [15, 28, 26, 29, 31, 32, 33, 34, 37, 38, 47, 49] for the precise definitions of the properties listed in the second and third column of Table 1.

4 Homological algebra

The main issue of checking effectively the system properties via the properties of modules defined in Section 3 was still open until recently. Only the case of multidimensional systems defined by a *full row rank* matrix R with entries in the commutative polynomial ring $k[x_1, \ldots, x_n]$ was known using the different concepts of primeness [26, 34, 48, 49] developed in the middle of the seventies.

The concepts of *syzygy modules*, *free resolutions*, *extension* and *torsion functors*, *projective* and *homotopic equivalences*, *projective dimensions*... developed in homological algebra [45] form the basis of new algorithms checking the first column of Table 1, and thus, the system properties. These algorithms were obtained in [38] in the case of PDEs (see also [31, 32, 33, 34]).

We have recently shown in [9, 11] how these algorithms could be extended to some classes of Ore algebras including the interesting ones from the control theory point of view (e.g., ODEs, PDEs, recurrence operators, time-delay operators). The main steps of the algorithms developed in [9, 11] are:

1. Computation of *free resolutions* of f.p. left modules over an Ore algebra.
2. *Dualization* of the previous free resolutions using the $\hom_D(\cdot, D)$ *functor*.
3. Use of *involutions* in order to pass from right to left D-modules.
4. Computation of the quotient module of f.p. left D-modules.

Table 1. Classification of structural properties

Module M	**Structural properties**	**Optimal control**
Torsion	Poles/zeros classifications	
With torsion	Existence of autonomous elements	
Torsion-free	No autonomous elements, Controllability, Parametrizability, π-flatness	Variational problem without constraints (Euler-Lagrange equations)
Reflexive	Filter identification	
Projective	Internal stabilizability, Bézout identities, Stabilizing controllers	Computation of the Lagrange parameters without integration
Free	Flatness, Poles placement, Doubly coprime factorization, Youla-Kučera parametrization	Optimal controller

Using the previous four points, we can then compute the *extension modules* $\mathrm{ext}^i_D(M, D)$, $i \in \mathbb{Z}_+$, of any left D-module of the form $M = D^{1\times p}/(D^{1\times q}\, R)$. Let us explain the previous concepts. See [45] for more details.

Definition 3. We have the following definitions:

- A *complex* of left D-modules is a sequence formed by left D-modules P_i and left D-morphisms $d_i : P_i \longrightarrow P_{i-1}$ which satisfy $\mathrm{im}\, d_{i+1} \subseteq \ker d_i$ for all $i \in \mathbb{Z}_+$. Such a complex is denoted by:

$$\ldots \xrightarrow{d_{i+2}} P_{i+1} \xrightarrow{d_{i+1}} P_i \xrightarrow{d_i} P_{i-1} \xrightarrow{d_{i-1}} P_{i-2} \xrightarrow{d_{i-2}} \ldots \tag{2}$$

- The left D-module $H(P_i) = \ker d_i / \operatorname{im} d_{i+1}$ is called the *defect of exactness* of (2) at P_i. The complex (2) is said to be *exact at* P_i if $H(P_i) = 0$, i.e., $\ker d_i = \operatorname{im} d_{i+1}$, and *exact* if $H(P_i) = 0$ for all $i \in \mathbb{Z}_+$.
- Let $M = D^{1\times p}/(D^{1\times q}\, R)$ be a finitely presented left D-module. A *free resolution* of M is an exact sequence of the form

$$\dots \xrightarrow{.R_3} D^{1\times p_2} \xrightarrow{.R_2} D^{1\times p_1} \xrightarrow{.R_1} D^{1\times p_0} \xrightarrow{\pi} M \longrightarrow 0, \tag{3}$$

where $p_0 = p$, $p_1 = q$, $R_1 = R$, $R_i \in D^{p_i \times p_{i-1}}$ and $.R_i : D^{1\times p_i} \longrightarrow D^{1\times p_{i-1}}$ is defined by $(.R_i)(\lambda) = \lambda\, R_i$ for all $\lambda \in D^{1\times p_i}$.
- Let us consider the free resolution (3) of M and the following complex

$$\dots \xleftarrow{R_3.} D^{p_2} \xleftarrow{R_2.} D^{p_1} \xleftarrow{R_1.} D^{p_0} \longleftarrow 0, \tag{4}$$

where $R_i. : D^{p_{i-1}} \longrightarrow D^{p_i}$ is defined by $(R_i.)(\lambda) = R_i\, \lambda$ for all $\lambda \in D^{p_{i-1}}$. Then, the defects of exactness of the complex (4) are denoted by:

$$\begin{cases} \operatorname{ext}^0_D(M, D) = \ker(R_1.), \\ \operatorname{ext}^i_D(M, D) = \ker(R_{i+1}.)/(R_i\, D^{1\times p_{i-1}}), \quad i \geq 1. \end{cases}$$

$\operatorname{ext}^i_D(M, D)$ inherits a right module structure by the right action of D.

Proposition 1. *[45] The right D-module $\operatorname{ext}^i_D(M, D)$ only depends on M, i.e., we can choose any free resolution of M to compute $\operatorname{ext}^i_D(M, D)$, $i \in \mathbb{Z}_+$. Moreover, we have $\operatorname{ext}^0_D(M, D) = \hom_D(M, D)$.*

Coming back to the four main algorithmic steps outlined above, we recall that an *involution* θ of D is a k-linear map $\theta : D \longrightarrow D$ satisfying:

$$\forall\, a_1, a_2 \in D, \quad \theta(a_1 \cdot a_2) = \theta(a_2) \cdot \theta(a_1), \quad \theta \circ \theta = \mathrm{id}_D. \tag{5}$$

Example 6. We have the following examples of involutions:

1. If $D = k[x_1, \dots, x_n]$ is a commutative polynomial algebra, then $\theta = id$ is a trivial involution.
2. If $D = A_n$ is the Weyl algebra and $P \in D$, then we let $\theta(P)$ be the classical *formal adjoint* of P obtained by multiplying a test function on the left of $P\,z$ and by integrating by parts [31, 33, 34]. Equivalently, θ is defined by $\theta(x_i) = x_i$ and $\theta(\partial_i) = -\partial_i$, $i = 1, \dots, n$.
3. Let $S_h(k)$ be the Ore algebra of shift operators defined in Example 2. Then, an involution θ of $S_h(k)$ is defined by $\theta(t) = -t$ and $\theta(\delta_h) = \delta_h$.
4. If $D_h(k)$ is the Ore algebra of differential time-delay operators defined in Example 3, then an involution θ of $D_h(k)$ can be defined by $\theta(t) = -t$, $\theta(\delta_h) = \delta_h$ and $\theta(\partial) = \partial$. This last result shows that a simple involution of $D_h(k)$ exists contrary to what was written in [11] (we thank V. Levandovskyy for pointing out to us this trivial mistake).

Now, if R is a matrix with entries in an Ore algebra having an involution θ (e.g., $A_n(k)$, $S_h(k)$, $D_h(k)$), then we can define $\theta(R) = (\theta(R_{ij}))^T$ and the left D-module $\widetilde{N} = D^{1\times q}/(D^{1\times p}\,\theta(R))$. The main idea developed in [9, 11, 31, 34, 38] is that the module properties in the first column of Table 1 are characterized by the vanishing of certain $\mathrm{ext}^i_D(\widetilde{N}, D)$ as it is shown in Table 2. We refer the reader to [42] for a constructive algorithm which checks freeness and computes bases of free modules.

Table 2. Characterization of module properties

Module M	$\mathbf{ext}^i_D(\widetilde{N}, D)$	$d(\widetilde{N})$	**Primeness**
With torsion	$\mathrm{ext}^1_D(\widetilde{N}, D) \cong t(M)$	$n-1$	$\emptyset$
Torsion-free	$\mathrm{ext}^1_D(\widetilde{N}, D) = 0$	$n-2$	Minor left-prime
Reflexive	$\mathrm{ext}^i_D(\widetilde{N}, D) = 0$, $i = 1, 2$	$n-3$	
Projective	$\mathrm{ext}^i_D(\widetilde{N}, D) = 0$, $1 \leq i \leq n$	-1	Zero left-prime

The last column of Table 2 explains the correspondence between module properties and primeness for a multidimensional system defined by a full row rank matrix R with entries in the commutative polynomial ring $D = k[x_1, \ldots, x_n]$ [34]. The third column generalizes the last column to multidimensional systems defined by a full row rank matrix R with entries in the ring of differential operators with rational coefficients and $d(\widetilde{N})$ denotes the *Krull dimension* of the *characteristic variety* of $\widetilde{N}$ (see [34, 39]).

5 Computation of $\mathrm{ext}^i_D(\widetilde{N}, D)$

The main difficulty in the computation of $\mathrm{ext}^i_D(\widetilde{N}, D)$ is to be able to construct a free resolution for the left D-module $\widetilde{N} = D^{1\times q}/(D^{1\times p}\,\theta(R))$ (see point 1 in the previous section), i.e., an exact sequence of the form

$$\ldots \xrightarrow{.\widetilde{R}_4} D^{1\times q_3} \xrightarrow{.\widetilde{R}_3} D^{1\times q_2} \xrightarrow{.\widetilde{R}_2} D^{1\times q_1} \xrightarrow{.\widetilde{R}_1} D^{1\times q_0} \longrightarrow \widetilde{N} \longrightarrow 0,$$

where $\widetilde{R}_1 = \theta(R)$, $q_0 = q$, $q_1 = p$ and $.\widetilde{R}_i : D^{1\times q_i} \to D^{1\times q_{i-1}}$ is defined by $(.\widetilde{R}_i)(\lambda) = \lambda\,\widetilde{R}_i$. The left D-module

$$S_i(\widetilde{N}) = \ker(.\widetilde{R}_i) = \{\lambda \in D^{1\times q_i} \mid \lambda\,\widetilde{R}_i = 0\}$$

is called the i^{th} *syzygy left D-module* of $\widetilde{N}$. If D is a noetherian ring, which is the case for a large class of algebras (e.g., A_n, S_h, D_h and $H_{(h,l)}$) [9, 11], then free resolutions always exist for finitely generated left D-modules [25, 45].

The computation of the matrix $\widetilde{R}_{i+1}$ is an *elimination problem* [2, 17]. Indeed, multiplying $\lambda \in S_i(\widetilde{N})$ on the left of the inhomogeneous system $\widetilde{R}_i\, y = u$, we then obtain $\lambda\, u = 0$. Hence, finding a family of generators for $S_i(\widetilde{N})$, i.e., $\{\lambda_j\}_{1\leq j\leq q_{i+1}}$, $\lambda_j \in D^{1\times q_i}$ satisfying $S_i(\widetilde{N}) = D\,\lambda_1 + \ldots + D\,\lambda_{q_{i+1}}$ is equivalent to finding a family of generators for the *compatibility conditions* of the inhomogeneous system $\widetilde{R}_i\, y = u$. Then, if we denote by $\widetilde{R}_{i+1} = (\lambda_1^T, \ldots, \lambda_{q_{i+1}}^T)^T$, we finally obtain $S_i(\widetilde{N}) = D^{1\times q_{i+1}}\,\widetilde{R}_{i+1}$.

Such a difficult problem has largely been studied for linear systems of PDEs since the 19^{th} century [31, 38, 39]. But, only recently some computational answers were found based on the concepts of *Janet* and *Gröbner bases* for non-commutative polynomial rings. We recall the definition of Gröbner bases for polynomial ideals. This definition can easily be extended to modules [2, 17]. The algorithmic methods used in the theory of Gröbner bases require that a monomial order is chosen to compare polynomials.

Definition 4. 1. Let D be an Ore algebra. A *monomial order* $<$ on D is defined as a total order on the *set of monomials* $\mathrm{Mon}(D)$ satisfying the following two conditions:
   a) For all monomials $m \in \mathrm{Mon}(D)\backslash\{1\}$, we have $1 < m$.
   b) If $m_1 < m_2$ holds for two monomials $m_1, m_2 \in \mathrm{Mon}(D)$, then, for all $n \in \mathrm{Mon}(D)$, we have $n \cdot m_1 < n \cdot m_2$.
2. Given a polynomial $P \in D\backslash\{0\}$ and a monomial order $<$ on D, we can compare the monomials with a non-zero coefficient in P w.r.t. $<$. The greatest of these monomials is the *leading monomial* $\mathrm{lm}(P)$ of P.

Definition 5. [2, 17] Let D be a polynomial ring and I a (left) ideal of D. A set of non-zero polynomials $G = \{g_1, \ldots, g_t\} \subset I$ is called a *Gröbner basis* for I if for all $0 \neq f \in I$, there exists $1 \leq i \leq t$ such that $\mathrm{lm}(g_i)$ divides $\mathrm{lm}(f)$.

One consequence of the condition that defines Gröbner bases is that every polynomial f in I is *reduced* to 0 modulo G, i.e., by subtraction of suitable left multiples of the $g_i \in G$ from f, we then obtain the zero polynomial.

For the case of commutative polynomial rings, *Buchberger's algorithm* [2, 17] computes Gröbner bases of polynomial ideals. Recently, Buchberger's algorithm was extended to some non-commutative polynomial rings and, in particular, to some classes of Ore algebras [5, 7] that are important for the study of multidimensional linear systems. Hence, manipulations of (one-sided) ideals and modules over many classes of Ore algebras have been turned effective. Moreover, the Maple library *Mgfun* [6] has been developed for the symbolic manipulation of a large class of special functions and combinatorial sequences. It offers implementations of Gröbner bases for some classes of Ore algebras.

6 The Package OreModules

Using the Maple library *Mgfun*, the authors of this paper have recently been developing the package OreModules [8, 10]. OreModules as well as a library of examples are freely available at:

`http://wwwb.math.rwth-aachen.de/OreModules`.

This second release of OreModules focuses on the following problems:

- Compute free resolutions, formal adjoints, extension functors, duals and biduals of f.p. left D-modules over some classes of Ore algebras D.
- Recognize the properties of a finitely presented left D-module M (torsion-free, reflexive, projective, stably free, free).
- Decide the existence of torsion elements in the corresponding system and, if so, compute a family of generators for them.
- Compute left/right/generalized inverses of matrices with entries in D.
- Check whether or not a multidimensional linear system is controllable in the sense of [14, 15, 26, 30, 29, 31, 32, 47, 49] or compute the autonomous elements of the system [30, 31, 32, 47, 49].
- Check whether or not a multidimensional linear system is parametrizable in the sense of [11, 31, 32, 33].
- Check whether or not a multidimensional linear system is flat and, if so, compute an injective parametrization and a flat output [15, 26, 31, 33, 42].
- Check whether or not a multidimensional linear system with constant coefficients is π-free and, if so, compute the ideal of all the π-polynomials [11, 15, 26].

A list of the most important functions of OreModules is given in Table 3 (the suffix "Rat" distinguishes the procedures which deal with polynomial/rational coefficients). Detailed documentation of OreModules is available in form of Maple help pages.

7 Worked examples using OreModules

OreModules comes with a library of examples which demonstrates the above features by means of systems like two pendula mounted on a cart, stirred

Table 3. List of the most important functions of OreModules

Main functions for the treatment of linear systems over Ore algebras D	
`Parametrization(Rat)`	Find parametrization of the system
`MinimalParametrization(s)(Rat)`	Find minimal parametrization(s) of the system
`AutonomousElements(Rat)`	Find a generating set of autonomous elements of the system (i.e., solve the system of equations for the torsion elements) in case of PDEs
`LeftInverse(Rat)`	Compute a left-inverse for a matrix over D
`LocalLeftInverse`	Given a $0 \neq \pi \in k[x_1, \ldots, x_n]$, compute a left inverse for a matrix over $k[x_1, \ldots, x_n, \pi^{-1}]$
`RightInverse(Rat)`	Compute a right-inverse for a matrix over D
`GeneralizedInverse(Rat)`	Compute a generalized inverse matrix over D
`PiPolynomial`	Given a system matrix over a commutative polynomial ring D and a variable $x_i \in D$, compute the ideal of all π-polynomials in x_i for the given system
`FirstIntegral`	In the case of ODEs, find first integrals of motion
`LQEquations`	Compute the Euler-Lagrange equations for a linear quadratic problem and a controllable OD system
Module theory over Ore algebras D	
`TorsionElements(Rat)`	Compute the torsion submodule of a f.p. D-module
`Exti(Rat)`	Given a f.p. left D-module M and j, compute $\mathrm{ext}^j_D(M, D)$
`Extn(Rat)`	Given a f.p. left D-module M and m, compute $\mathrm{ext}^i_D(M, D)$ for $0 \leq i \leq m$
`Quotient(Rat)`	Compute the quotient module of two left D-modules generated by the rows of two matrices
`SyzygyModule(Rat)`	Compute the first syzygy module of a f.p. left D-module
`Resolution(Rat)`	Given i, compute the first i^{th} terms of a free resolution of a f.p. left D-module
`FreeResolution(Rat)`	Compute a free resolution of a f.p. left D-module
`OreRank(Rat)`	Compute the rank of a f.p. left D-module
Some low-level functions of OreModules	
`DefineOreAlgebra`	Set up an Ore algebra D in OreModules
`Involution`	Apply an involution to a matrix over D
`Factorize(Rat)`	Right-divide a matrix over D by another one
`Mult`	Multiply two or more matrices over D
`ApplyMatrix`	Apply (matrices of) operators in D to (vectors of) functions

tank models, electric transmission line, wind tunnel model, Maxwell equations, Einstein equations, equations of linear elasticity, Lie-Poisson structures... We only give here four simple examples but we refer the reader to [8, 9, 11] for more sophisticated examples. All examples were run on a Pentium III, 1 GHz with 1 GB RAM using Maple 8 (OREMODULES is available for Maple V release 5, Maple 6, Maple 8, Maple 9 and Maple 10).

Example 7. We study a linearized bipendulum [31], i.e., a system composed of a bar, where two pendula are fixed, one of length $l1$ and one of length $l2$. The appropriate Ore algebra for this example is the Weyl algebra $Alg = A_1$, i.e., $A_1 = \mathbb{Q}(l1, l2, g)[t][\mathrm{D}]$, where $\mathrm{D} = \frac{d}{dt}$ is the differential operator w.r.t. time t:

```
> with(OreModules):
> Alg:=DefineOreAlgebra(diff=[D,t],polynom=[t],comm=[g,l1,l2]):
```

Note that we have to declare all constants appearing in the system equations (the gravitational constant g and the lengths $l1$ and $l2$) as variables that commute with D and t. Next, we enter the system matrix:

```
> R:=evalm([[D^2+g/l1, 0, -g/l1], [0, D^2+g/l2, -g/l2]]);
```

$$R := \begin{bmatrix} \mathrm{D}^2 + \dfrac{g}{l1} & 0 & -\dfrac{g}{l1} \\ 0 & \mathrm{D}^2 + \dfrac{g}{l2} & -\dfrac{g}{l2} \end{bmatrix}$$

In terms of equations, the linearized bipendulum is described by:

```
> ApplyMatrix(R, [x1(t),x2(t),u(t)], Alg) = evalm([[0],[0]]);
```

$$\begin{bmatrix} \dfrac{g\,\mathrm{x1}(t)}{l1} + (\frac{d^2}{dt^2}\,\mathrm{x1}(t)) - \dfrac{g\,\mathrm{u}(t)}{l1} \\ \dfrac{g\,\mathrm{x2}(t)}{l2} + (\frac{d^2}{dt^2}\,\mathrm{x2}(t)) - \dfrac{g\,\mathrm{u}(t)}{l2} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

We compute the formal adjoint of R:

```
> R_adj:=Involution(R, Alg);
```

$$R\_adj := \begin{bmatrix} \mathrm{D}^2 + \dfrac{g}{l1} & 0 \\ 0 & \mathrm{D}^2 + \dfrac{g}{l2} \\ -\dfrac{g}{l1} & -\dfrac{g}{l2} \end{bmatrix}$$

By computing $\mathrm{ext}^1_{A_1}(A_1^{1\times 2}/(A^{1\times 3}\,R\_adj), A_1)$, we check whether or not the left A_1-module $M = A_1^{1\times 3}/(A_1^{1\times 2}\,R)$ is torsion-free, i.e., whether or not the bipendulum is controllable and parametrizable:

```
> Ext:=Exti(R_adj, Alg, 1);
```

$$Ext := \left[\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} \mathrm{D}^2\, l1 + g & 0 & -g \\ 0 & \mathrm{D}^2\, l2 + g & -g \end{bmatrix}, \begin{bmatrix} l2\, \mathrm{D}^2\, g + g^2 \\ g^2 + \mathrm{D}^2\, l1\, g \\ l2\, \mathrm{D}^2\, g + l2\, l1\, \mathrm{D}^4 + \mathrm{D}^2\, l1\, g + g^2 \end{bmatrix} \right]$$

From the output, we can see that the system is *generically* controllable because $Ext[1]$ is the identity matrix which means that there are no torsion elements in the left A_1-module M associated with the system. The interpretation of this structural fact is that the system has no autonomous elements *in the generic case* (see Section 3). There may be a few configurations of the constants g, $l1$, $l2$, in which the bipendulum is not controllable. We will actually find the only configuration where it is not controllable below. Let us write down the generic parametrization $Ext[3]$ in a more familiar way with a free function ξ_1.

```
> P:=Parametrization(R, Alg);
```

$$P := \begin{bmatrix} g\,(g\,\xi_1(t) + l2\, \frac{d^2}{dt^2}\,\xi_1(t) \\ g\,(g\,\xi_1(t) + l1\, \frac{d^2}{dt^2}\,\xi_1(t) \\ g^2\,\xi_1(t) + g\, l2\, \frac{d^2}{dt^2}\,\xi_1(t) + g\, l1\, \frac{d^2}{dt^2}\,\xi_1(t) + l1\, l2\, (\frac{d^4}{dt^4}\,\xi_1(t)) \end{bmatrix}$$

Therefore, all smooth solutions of the system are parametrized by P, i.e.,

$$R\,(x_1,\, x_2,\, u)^T = 0 \;\Leftrightarrow\; (x_1,\, x_2,\, u)^T = Ext[3]\,\xi_1 = P\,\xi_1.$$

Since the bipendulum is generically a time-invariant controllable system, it is also generically a flat system. A *flat output* of the system can be computed as a left-inverse of the parametrization $Ext[3]$:

```
> S:=LeftInverse(Ext[3], Alg);
```

$$S := \begin{bmatrix} \frac{l1}{g^2\,(l1 - l2)} & -\frac{l2}{g^2\,(l1 - l2)} & 0 \end{bmatrix}$$

i.e., a flat output is defined by $\xi_1 = S\ (x_1,\ \ x_2,\ \ u)^T$, namely:

```
> xi[1](t)=ApplyMatrix(S, [x1(t),x2(t),u(t)], Alg)[1,1];
```

$$\xi_1(t) = \frac{l1\ \mathrm{x1}(t)}{g^2\,(l1 - l2)} - \frac{l2\ \mathrm{x2}(t)}{g^2\,(l1 - l2)}$$

We remark that this flat output is defined only if $l1 - l2 \neq 0$. In this case, let us compute the Brunovský canonical form of the system.

```
> B:=Brunovsky(R, Alg);
```

$$B := \begin{bmatrix} \frac{l1}{g^2\,(l1-l2)} & -\frac{l2}{g^2\,(l1-l2)} & 0 \\ \frac{\mathrm{D}\,l1}{g^2\,(l1-l2)} & -\frac{\mathrm{D}\,l2}{g^2\,(l1-l2)} & 0 \\ -\frac{1}{g\,(l1-l2)} & \frac{1}{g\,(l1-l2)} & 0 \\ -\frac{\mathrm{D}}{g\,(l1-l2)} & \frac{\mathrm{D}}{g\,(l1-l2)} & 0 \\ \frac{1}{(l1-l2)\,l1} & -\frac{1}{(l1-l2)\,l2} & \frac{1}{l1\,l2} \end{bmatrix}$$

In other words, we have the following transformation between the system variables x_1, x_2 and u and the Brunovský variables z_i, $i = 1, \ldots, 4$, and v:

```
> evalm([seq([z[i](t)],i=1..4),[v(t)]])=ApplyMatrix(B,
> [x1(t),x2(t),u(t)], Alg);
```

$$\begin{bmatrix} z_1(t) \\ z_2(t) \\ z_3(t) \\ z_4(t) \\ \mathrm{v}(t) \end{bmatrix} = \begin{bmatrix} \frac{l1\,\mathrm{x1}(t)}{g^2\,(l1-l2)} - \frac{l2\,\mathrm{x2}(t)}{g^2\,(l1-l2)} \\ \frac{l1\,(\frac{d}{dt}\,\mathrm{x1}(t))}{g^2\,(l1-l2)} - \frac{l2\,(\frac{d}{dt}\,\mathrm{x2}(t))}{g^2\,(l1-l2)} \\ -\frac{\mathrm{x1}(t)}{g\,(l1-l2)} + \frac{\mathrm{x2}(t)}{g\,(l1-l2)} \\ -\frac{\frac{d}{dt}\,\mathrm{x1}(t)}{g\,(l1-l2)} + \frac{\frac{d}{dt}\,\mathrm{x2}(t)}{g\,(l1-l2)} \\ \frac{\mathrm{x1}(t)}{(l1-l2)\,l1} - \frac{\mathrm{x2}(t)}{(l1-l2)\,l2} + \frac{\mathrm{u}(t)}{l1\,l2} \end{bmatrix}$$

Let us check that the new variables z_i, $i = 1, \ldots, 4$, and v satisfy a Brunovský canonical form:

```
> F:=Elimination(linalg[stackmatrix](B, R), [x1,x2,u],
> [seq(z[i],i=1..4),v,0,0], Alg):
> ApplyMatrix(F[1], [x1(t),x2(t),u(t)], Alg)=ApplyMatrix(F[2],
> [seq(z[i](t),i=1..4),v(t)], Alg);
```

$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \mathrm{u}(t) \\ \mathrm{x2}(t) \\ \mathrm{x1}(t) \end{bmatrix} = \begin{bmatrix} -(\frac{d}{dt}\,z_4(t)) + \mathrm{v}(t) \\ -(\frac{d}{dt}\,z_3(t)) + z_4(t) \\ -(\frac{d}{dt}\,z_2(t)) + z_3(t) \\ -(\frac{d}{dt}\,z_1(t)) + z_2(t) \\ g^2\,z_1(t) + (g\,l2 + g\,l1)\,z_3(t) + l1\,l2\,\mathrm{v}(t) \\ g^2\,z_1(t) + g\,l1\,z_3(t) \\ g^2\,z_1(t) + g\,l2\,z_3(t) \end{bmatrix}$$

The last three equations give u, x_1 and x_2 in terms of the z_i and v.

$l1 = l2$ describes the only case in which the bipendulum may be uncontrollable. We now turn to the case where the lengths of the pendula are equal:

```
> R_mod:=subs(l2=l1, evalm(R));
```

$$R\_mod := \begin{bmatrix} D^2 + \dfrac{g}{l1} & 0 & -\dfrac{g}{l1} \\ 0 & D^2 + \dfrac{g}{l1} & -\dfrac{g}{l1} \end{bmatrix}$$

```
> Ext_mod:=Exti(Involution(R_mod, Alg), Alg, 1);
```

$$Ext\_mod := \left[\begin{bmatrix} D^2\, l1 + g & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & -1 & 0 \\ 0 & D^2\, l1 + g & -g \end{bmatrix}, \begin{bmatrix} g \\ g \\ D^2\, l1 + g \end{bmatrix} \right]$$

The computation of $\mathrm{ext}^1_{A_1}(A_1^{1\times 2}/(A_1^{1\times 3}\,\theta(R_{mod})), A_1)$ gives the torsion submodule $t(M)$ of M: it is generated by the residue class of the row r of $Ext\_mod[2]$ which corresponds to the row with entry $l1\,D^2 + g$ in $Ext\_mod[1]$. This means that $(l1\,D^2 + g)\,r = 0$ in M, and the difference of the positions of the pendula (relative to the bar) is an autonomous element of the system. We can conclude that the bipendulum is controllable if and only if $l1 \neq l2$.

We can directly obtain the torsion elements of M as follows:

```
> TorsionElements(R_mod, [x1(t),x2(t),u(t)], Alg);
```

$$[\left[g\,\theta_1(t) + l1\,(\tfrac{d^2}{dt^2}\,\theta_1(t)) = 0 \right], \left[\theta_1(t) = \mathrm{x1}(t) - \mathrm{x2}(t) \right]]$$

We can also explicitly integrate this torsion element of M:

```
> AutonomousElements(R_mod, [x1(t),x2(t),u(t)], Alg)[2];
```

$$\left[\theta_1 = \_C1\,\sin(\frac{\sqrt{g}\,t}{\sqrt{l1}}) + \_C2\,\cos(\frac{\sqrt{g}\,t}{\sqrt{l1}}) \right]$$

The fact that there exists an autonomous element in the system is equivalent to the existence of a first integral of motion in the system. Indeed, we recall that we have a one-to-one correspondence between the torsion elements and the first integrals of motion. For more details, see [33]. We can compute this first integral of motion by using the command FirstIntegral:

```
> V:=FirstIntegral(R_mod, [x1(t),x2(t),u(t)], Alg);
```

$$V := -(-(\tfrac{d}{dt}\,\mathrm{x1}(t))\,\_C1\,\sin(\frac{\sqrt{g}\,t}{\sqrt{l1}})\,\sqrt{l1} - (\tfrac{d}{dt}\,\mathrm{x1}(t))\,\_C2\,\cos(\frac{\sqrt{g}\,t}{\sqrt{l1}})\,\sqrt{l1}$$
$$+ \sqrt{g}\,\mathrm{x1}(t)\,\_C1\,\cos(\frac{\sqrt{g}\,t}{\sqrt{l1}}) - \sqrt{g}\,\mathrm{x1}(t)\,\_C2\,\sin(\frac{\sqrt{g}\,t}{\sqrt{l1}})$$
$$+ (\tfrac{d}{dt}\,\mathrm{x2}(t))\,\_C1\,\sin(\frac{\sqrt{g}\,t}{\sqrt{l1}})\,\sqrt{l1} + (\tfrac{d}{dt}\,\mathrm{x2}(t))\,\_C2\,\cos(\frac{\sqrt{g}\,t}{\sqrt{l1}})\,\sqrt{l1}$$
$$- \sqrt{g}\,\mathrm{x2}(t)\,\_C1\,\cos(\frac{\sqrt{g}\,t}{\sqrt{l1}}) + \sqrt{g}\,\mathrm{x2}(t)\,\_C2\,\sin(\frac{\sqrt{g}\,t}{\sqrt{l1}}))\,/\sqrt{l1}$$

We let the reader check by himself that we have $\dot{V}(t) = 0$. For the explicit computations, see [8].

Finally, even if we have some autonomous elements in the system, we can parametrize all solutions of the system in terms of one arbitrary function ξ_1 and two arbitrary constants C_1 and C_2 (these constants can easily be computed in terms of the initial conditions of the system):

```
> P2:=Parametrization(R_mod, Alg);
```

$$P2 := \begin{bmatrix} g\,\xi_1(t) \\ -\_C1\,\sin(\frac{\sqrt{g}\,t}{\sqrt{l1}}) - \_C2\,\cos(\frac{\sqrt{g}\,t}{\sqrt{l1}}) + g\,\xi_1(t) \\ l1\,(\frac{d^2}{dt^2}\,\xi_1(t)) + g\,\xi_1(t) \end{bmatrix}$$

i.e., we have $R\,(x_1,\,x_2,\,u)^T = 0 \Leftrightarrow (x_1,\,x_2,\,u)^T = P_2(\xi_1, \_C1, \_C2)$. We can easily check that P_2 gives a parametrization of some solutions of the system as we have:

```
> simplify(ApplyMatrix(R_mod, P2, Alg));
```

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

We can prove that P_2 parametrizes all smooth solutions of the system [40].

Example 8. This example demonstrates the study of structural properties of a simple linear time-varying ordinary differential system [41, 43]. See [8] for more sophisticated examples.

```
> Alg:=DefineOreAlgebra(diff=[D,t], polynom=[t]):
```

Let us consider the following matrix of ordinary differential operators:

```
> R:=evalm([[D, -t]]);
```

$$R := \begin{bmatrix} \mathrm{D} & -t \end{bmatrix}$$

The matrix R corresponds to the following time-varying linear system:

```
> ApplyMatrix(R, [x(t),u(t)], Alg)[1,1]=0;
```

$$(\frac{d}{dt}\,\mathrm{x}(t)) - t\,\mathrm{u}(t) = 0$$

Let us check whether or not this system is controllable and flat. In order to do that, let us define the formal adjoint R_adj of R.

```
> R_adj:=Involution(R, Alg);
```

$$R\_adj := \begin{bmatrix} -\mathrm{D} \\ -t \end{bmatrix}$$

We compute the first extension module $\mathrm{ext}^1_{A_1(\mathbb{Q})}(A_1(\mathbb{Q})/(A_1(\mathbb{Q})^{1\times 2}\,R\_adj), A_1(\mathbb{Q}))$ of the left *Alg*-module associated with R_adj:

```
> Ext:=Exti(R_adj, Alg, 1);
```

$$Ext := [\begin{bmatrix} 1 \end{bmatrix}, \begin{bmatrix} \mathrm{D} & -t \end{bmatrix}, \begin{bmatrix} -t^2 & -1 + t\,\mathrm{D} \\ -2 - t\,\mathrm{D} & \mathrm{D}^2 \end{bmatrix}]$$

Therefore, we obtain that the left A_1-module $M = A_1(\mathbb{Q})^{1\times 2}/(A_1(\mathbb{Q})\,R)$ associated with R is torsion-free, and thus, stably free as $A_1(\mathbb{Q})$ is a hereditary ring. A parametrization of the system is given by $Ext[3]$. This result can directly be obtained by using the following command:

```
> Parametrization(R, Alg);
```

$$\begin{bmatrix} -t^2\,\xi_1(t) - \xi_2(t) + t\,(\frac{d}{dt}\,\xi_2(t)) \\ -2\,\xi_1(t) - t\,(\frac{d}{dt}\,\xi_1(t)) + (\frac{d^2}{dt^2}\,\xi_2(t)) \end{bmatrix}$$

Let us notice that the previous parametrization depends on two arbitrary functions ξ_1 and ξ_2. However, the system has only 1 input, and thus, the rank of the left $A_1(\mathbb{Q})$-module M is 1. Let us check this result:

```
> OreRank(R, Alg);
```

$$1$$

Hence, we deduce that there exist some minimal parametrizations of the system which depend on 1 arbitrary function. Let us compute some of them.

```
> P:=MinimalParametrizations(R, Alg);
```

$$P := [\begin{bmatrix} -t^2 \\ -2 - t\,\mathrm{D} \end{bmatrix}, \begin{bmatrix} -1 + t\,\mathrm{D} \\ \mathrm{D}^2 \end{bmatrix}]$$

Let us check whether or not the first minimal parametrization $P[1]$ is injective.

```
> LeftInverse(P[1], Alg);
```

$$[]$$

We obtain that $P[1]$ is not an injective parametrization of the system. Let us examine the second minimal parametrization $P[2]$ in a similar way:

```
> LeftInverse(P[2], Alg);
```

$$[]$$

We find that $P[2]$ is not an injective parametrization of the system either. Therefore, we cannot conclude that the left $A_1(\mathbb{Q})$-module M associated with the system is free. In fact, we can prove that M is not a free left $A_1(\mathbb{Q})$-module, and thus, the corresponding time-varying system is not flat. See [39, 42] for more details. However, we already know that M is a stably free left $A_1(\mathbb{Q})$-module as the matrix R has full row rank and R admits a right-inverse defined by:

```
> RightInverse(R, Alg);
```

$$\begin{bmatrix} t \\ \mathrm{D} \end{bmatrix}$$

See [11] for more details. One of the main interests of the non-minimal parametrization $Ext[3]$ is that it admits a generalized inverse, namely, there exists a matrix G with entries in $A_1(\mathbb{Q})$ satisfying $Ext[3]\ G\ Ext[3] = Ext[3]$ (contrary to $P[1]$ and $P[2]$). This last result implies that the non-minimal parametrization parametrizes all the solutions of the corresponding time-varying system which belong to any left $A_1(\mathbb{Q})$-module $\mathcal{F}$ (e.g., $\mathcal{F} = C^\infty(\mathbb{R})$, $\mathbb{R}(t)$, $\mathbb{R}[t]$). Let us compute one generalized inverse of $Ext[3]$:

```
> G:=GeneralizedInverse(Ext[3], Alg);
```

$$G := \begin{bmatrix} 0 & -1 \\ -1 & 0 \end{bmatrix}$$

```
> Mult(Ext[3], G, Ext[3], Alg);
```

$$\begin{bmatrix} -t^2 & -1 + t\,\mathrm{D} \\ -2 - t\,\mathrm{D} & \mathrm{D}^2 \end{bmatrix}$$

Let us determine the obstruction of flatness. In order to do that, we study the system over the ring $\mathbb{Q}(t)\left[\frac{d}{dt}\right]$ of ordinary differential operators with rational coefficients in t. Let us compute a parametrization of the system by allowing to invert non-zero polynomials in t:

```
> Extrat:=ExtiRat(R_adj, Alg, 1);
```

$$Extrat := [\begin{bmatrix} 1 \end{bmatrix},\ \begin{bmatrix} \mathrm{D} & -t \end{bmatrix},\ \begin{bmatrix} -t^2 \\ -2 - t\,\mathrm{D} \end{bmatrix}]$$

We obtain that the left $\mathbb{Q}(t)\left[\frac{d}{dt}\right]$-module $M' = \mathbb{Q}(t)\left[\frac{d}{dt}\right]^{1\times 2}/(\mathbb{Q}(t)\left[\frac{d}{dt}\right] R)$ is torsion-free, and thus, free because $\mathbb{Q}(t)\left[\frac{d}{dt}\right]$ is a left principal ideal domain.

Moreover, a (minimal) parametrization of the system is defined by $Extrat[3]$. This result can directly be obtained by using ParametrizationRat:

```
> ParametrizationRat(R, Alg);
```

$$\begin{bmatrix} -t^2\,\xi_1(t) \\ -2\,\xi_1(t) - t\,(\frac{d}{dt}\,\xi_1(t)) \end{bmatrix}$$

The fact that the left *AlgRat*-module M associated with R is free implies that, away from some singularities that we are going to determine, the system is flat. Let us compute a basis for this module which gives a flat output of the system.

```
> S:=LeftInverseRat(Extrat[3], Alg);
```

$$S := \begin{bmatrix} -\frac{1}{t^2} & 0 \end{bmatrix}$$

Therefore, we obtain that a basis of the left *AlgRat*-module M is defined by $\xi_1 = S\ (x, \quad u)^T$ and satisfies:

$$(x,\, u)^T = Extrat[3]\,\xi_1.$$

In particular, we see that this parametrization is not defined for $t = 0$ as we have a singularity. Therefore, the system is flat except for $t = 0$. Finally, we note that, away from $t = 0$, we have another right-inverse of R defined by:

```
> RightInverseRat(R, Alg);
```

$$\begin{bmatrix} 0 \\ -\frac{1}{t} \end{bmatrix}$$

Let us compute the Brunovský canonical form:

```
> B:=BrunovskyRat(R, Alg);
```

$$B := \begin{bmatrix} -\frac{1}{t^2} & 0 \\ \frac{2}{t^3} & -\frac{1}{t} \end{bmatrix}$$

Let us check that the variables z and v defined by $(z, \quad v)^T = B\ (x, \quad u)^T$ satisfy a Brunovský canonical form:

```
> E:=EliminationRat(linalg[stackmatrix](B, R), [x,u], [z,v,0],
> Alg):
> ApplyMatrix(E[1], [x(t),u(t)], Alg)=ApplyMatrix(E[2],
> [z(t),v(t)], Alg);
```

$$\begin{bmatrix} 0 \\ \mathrm{u}(t) \\ \mathrm{x}(t) \end{bmatrix} = \begin{bmatrix} -(\frac{d}{dt}\,\mathrm{z}(t)) + \mathrm{v}(t) \\ -2\,\mathrm{z}(t) - t\,\mathrm{v}(t) \\ -t^2\,\mathrm{z}(t) \end{bmatrix}$$

The first equation shows that z and v satisfy a Brunovský canonical form. The last two equations give x and u in terms of z and v. We refer to [8] for more difficult examples of time-varying ordinary differential linear systems.

Example 9. Let us consider the example of a two reflector antenna [26]. We first define an Ore algebra with a differential operator Dt w.r.t. time t and a constant time-delay operator δ. Note also that the constants $K1$, $K2$, Te, Kp, Kc have to be declared in the definition of the Ore algebra.

```
> Alg:=DefineOreAlgebra(diff=[Dt,t], dual_shift=[delta,s],
> polynom=[t,s], comm=[K1,K2,Te,Kp,Kc], shift_action=[delta,t]):
```

Enter the matrix R of the differential time-delay linear system:

```
> R:=evalm([[Dt,-K1,0,0,0,0,0,0,0],
> [0,Dt+K2/Te,0,0,0,0,-Kp/Te*delta,-Kc/Te*delta,-Kc/Te*delta],
> [0,0,Dt,-K1,0,0,0,0,0],
> [0,0,0,Dt+K2/Te,0,0,-Kc/Te*delta,-Kp/Te*delta,-Kc/Te*delta],
> [0,0,0,0,Dt,-K1,0,0,0],
> [0,0,0,0,0,Dt+K2/Te,-Kc/Te*delta,-Kc/Te*delta,-Kp/Te*delta]]);
```

$$R := \begin{bmatrix} Dt & -K1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & Dt + \frac{K2}{Te} & 0 & 0 & 0 & 0 & -\frac{Kp\,\delta}{Te} & -\frac{Kc\,\delta}{Te} & -\frac{Kc\,\delta}{Te} \\ 0 & 0 & Dt & -K1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & Dt + \frac{K2}{Te} & 0 & 0 & -\frac{Kc\,\delta}{Te} & -\frac{Kp\,\delta}{Te} & -\frac{Kc\,\delta}{Te} \\ 0 & 0 & 0 & 0 & Dt & -K1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & Dt + \frac{K2}{Te} & -\frac{Kc\,\delta}{Te} & -\frac{Kc\,\delta}{Te} & -\frac{Kp\,\delta}{Te} \end{bmatrix}$$

Then, we use an involution θ of Alg in order to obtain $R\_adj = \theta(R)$:

```
> R_adj:=Involution(R, Alg):
```

By means of the next command, we compute the torsion-free part (if *Ext1* [1] is not the identity matrix, then the torsion submodule is generated by the rows of *Ext1* [2] modulo the module generated by the rows of R) and a parametrization of the torsion-free part in *Ext1* [3]. Equivalently, we check whether or not the two reflector antenna is controllable:

```
> st:=time(): Ext1:=Exti(R_adj, Alg, 1): time() - st;
```

$$0.920$$

```
> Ext1[1];
```

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

We conclude that the first extension module $\mathrm{ext}^1_{Alg}(\widetilde{N}, Alg)$ of the Alg-module $\widetilde{N} = Alg^{1\times 6}/(A^{1\times 9}\,\theta(R))$ associated with $R\_adj = \theta(R)$ is the zero module. Hence, the module defined by R is torsion-free. Equivalently, R is parametrizable and *Ext1* [3] gives a parametrization of R involving three free parameters:

```
> Ext1[3];
```

$$\begin{bmatrix} K1\,\delta\,Kc & K1\,\delta\,Kc & Kp\,K1\,\delta \\ Dt\,\delta\,Kc & Dt\,\delta\,Kc & Kp\,\delta\,Dt \\ K1\,\delta\,Kc & Kp\,K1\,\delta & K1\,\delta\,Kc \\ Dt\,\delta\,Kc & Kp\,\delta\,Dt & Dt\,\delta\,Kc \\ Kp\,K1\,\delta & K1\,\delta\,Kc & K1\,\delta\,Kc \\ Kp\,\delta\,Dt & Dt\,\delta\,Kc & Dt\,\delta\,Kc \\ 0 & 0 & Dt^2\,Te + Dt\,K2 \\ 0 & Dt^2\,Te + Dt\,K2 & 0 \\ Dt^2\,Te + Dt\,K2 & 0 & 0 \end{bmatrix}$$

The same parametrization can be obtained by using PARAMETRIZATION. The result involves three free functions ξ_1, ξ_2, ξ_3:

```
> Parametrization(R, Alg);
```

$$\begin{bmatrix} K1\,Kc\,\xi_1(t-1) + K1\,Kc\,\xi_2(t-1) + Kp\,K1\,\xi_3(t-1) \\ Kc\,\mathrm{D}(\xi_1)(t-1) + Kc\,\mathrm{D}(\xi_2)(t-1) + Kp\,\mathrm{D}(\xi_3)(t-1) \\ K1\,Kc\,\xi_1(t-1) + Kp\,K1\,\xi_2(t-1) + K1\,Kc\,\xi_3(t-1) \\ Kc\,\mathrm{D}(\xi_1)(t-1) + Kp\,\mathrm{D}(\xi_2)(t-1) + Kc\,\mathrm{D}(\xi_3)(t-1) \\ Kp\,K1\,\xi_1(t-1) + K1\,Kc\,\xi_2(t-1) + K1\,Kc\,\xi_3(t-1) \\ Kp\,\mathrm{D}(\xi_1)(t-1) + Kc\,\mathrm{D}(\xi_2)(t-1) + Kc\,\mathrm{D}(\xi_3)(t-1) \\ Te\,(\mathrm{D}^{(2)})(\xi_3)(t) + K2\,\mathrm{D}(\xi_3)(t) \\ Te\,(\mathrm{D}^{(2)})(\xi_2)(t) + K2\,\mathrm{D}(\xi_2)(t) \\ Te\,(\mathrm{D}^{(2)})(\xi_1)(t) + K2\,\mathrm{D}(\xi_1)(t) \end{bmatrix}$$

The two reflector antenna is not a flat system [15, 26] because $\mathrm{ext}^2_{Alg}(\widetilde{N}, Alg)$ of the Alg-module $\widetilde{N}$ is different from zero as it is shown next:

```
> st:=time(): Ext2:=Exti(R_adj, Alg, 2): time() - st;
```

$$0.750$$

```
> Ext2[1];
```

$$\begin{bmatrix} \delta & 0 & 0 \\ Dt^2\,Te + Dt\,K2 & 0 & 0 \\ 0 & \delta & 0 \\ 0 & Dt^2\,Te + Dt\,K2 & 0 \\ 0 & 0 & \delta \\ 0 & 0 & Dt^2\,Te + Dt\,K2 \end{bmatrix}$$

Since the *torsion-free degree* $i(M)$ of $M = Alg^{1\times 9}/(Alg^{1\times 6}\,R)$ is equal to 1 (i.e., M is a torsion-free but not a projective Alg-module [11, 47]), we can find a polynomial π in the variable δ such that the system is π-*free* [15, 26]:

```
> PiPolynomial(R, Alg, [delta]);
```

$$[\delta]$$

We obtain $\pi = \delta$. By definition of the π-polynomial [15, 26], this means that if we can permit the time-advance operator δ^{-1}, then the system of the two reflector antenna becomes flat, i.e., the new $D = Alg[\delta^{-1}]$-module $P = D^{1\times 9}/(D^{1\times 6}\, R)$ associated with the system is free. We shall find a basis for the D-module P below.

We note that the fact that the two reflector antenna is not a flat system (without the advance operator δ^{-1}) is coherent with the fact that the full row-rank matrix R does not admit a right-inverse. Indeed, we can prove that a full row-rank matrix R admits a right-inverse if and only if the Alg-module $M = Alg^{1\times 9}/(Alg^{1\times 6}\, R)$ is projective [11]. By the Quillen-Suslin theorem (see 3 of Theorem 1), projective modules over commutative polynomial rings are free. This remark applies to our situation as we have:

```
> SyzygyModule(R, Alg); RightInverse(R, Alg);
```

$$\mathrm{INJ}(6)$$

$$[]$$

The fact that the system is not flat is also coherent with the fact that its parametrization *Ext1*[3] does not admit a left-inverse. Indeed, a linear system is flat if and only if it admits a left-invertible parametrization [11].

```
> LeftInverse(Ext1[3], Alg);
```

$$[]$$

We finish by computing a basis of the free $D = Alg[\delta^{-1}]$-module P. In the terminology of control, such a basis is called a *flat output*. We apply LocalLeftInverse to the parametrization *Ext1*[3] by allowing to invert δ:

```
> S:=LocalLeftInverse(Ext1[3], [delta], Alg);
```

$$S := \begin{bmatrix} -\frac{Kc}{\delta\, K1\, \%1} & 0 & -\frac{Kc}{\delta\, K1\, \%1} & 0 & \frac{Kp + Kc}{\delta\, K1\, \%1} & 0 & 0 & 0 & 0 \\ -\frac{Kc}{\delta\, K1\, \%1} & 0 & \frac{Kp + Kc}{\delta\, K1\, \%1} & 0 & -\frac{Kc}{\delta\, K1\, \%1} & 0 & 0 & 0 & 0 \\ \frac{Kp + Kc}{\delta\, K1\, \%1} & 0 & -\frac{Kc}{\delta\, K1\, \%1} & 0 & -\frac{Kc}{\delta\, K1\, \%1} & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\%1 := Kp^2 - 2\, Kc^2 + Kp\, Kc$$

By construction, the matrix S is a left-inverse of *Ext1*[3]:

```
> Mult(S, Ext1[3], Alg);
```

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Therefore, $(z_1, \; z_2, \; z_3)^T = S \; (x_1, \ldots, x_6, \; u_1, \; u_2, \; u_3)^T$ is a basis of the D-module P associated with R, and thus, a flat output of the two reflector antenna. Therefore, a flat output $(z_1, \; z_2, \; z_3)^T$ of the system is defined by:

```
> evalm([seq([z[i](t)], i=1..3)])=
> ApplyMatrix(S,[seq(x[i](t),i=1..6),seq(u[i](t),i=1..3)],Alg);
```

$$\begin{bmatrix} z_1(t) \\ z_2(t) \\ z_3(t) \end{bmatrix} = \begin{bmatrix} -\dfrac{Kc\, x_1(t+1)}{K1\, \%1} - \dfrac{Kc\, x_3(t+1)}{K1\, \%1} + \dfrac{(Kc+Kp)\, x_5(t+1)}{K1\, \%1} \\ -\dfrac{Kc\, x_1(t+1)}{K1\, \%1} + \dfrac{(Kc+Kp)\, x_3(t+1)}{K1\, \%1} - \dfrac{Kc\, x_5(t+1)}{K1\, \%1} \\ \dfrac{(Kc+Kp)\, x_1(t+1)}{K1\, \%1} - \dfrac{Kc\, x_3(t+1)}{K1\, \%1} - \dfrac{Kc\, x_5(t+1)}{K1\, \%1} \end{bmatrix}$$

$$\%1 := Kp\, Kc - 2\, Kc^2 + Kp^2$$

Finally, if we substitute $(z_1, \; z_2, \; z_3)^T$ into the parametrization $Ext1[3]$ of the system, we obtain $(x_1, \ldots, x_6, \, u_1, \, u_2, \, u_3)^T = T(x_1, \ldots, x_6, \, u_1, \, u_2, \, u_3)^T$, where the matrix T is defined by:

```
> T:=Mult(Ext1[3], S, Alg);
```

$$T := \begin{bmatrix} 1, 0, 0, 0, 0, 0, 0, 0, 0 \\ \dfrac{Dt}{K1}, 0, 0, 0, 0, 0, 0, 0, 0 \\ 0, 0, 1, 0, 0, 0, 0, 0, 0 \\ 0, 0, \dfrac{Dt}{K1}, 0, 0, 0, 0, 0, 0 \\ 0, 0, 0, 0, 1, 0, 0, 0, 0 \\ 0, 0, 0, 0, \dfrac{Dt}{K1}, 0, 0, 0, 0 \\ \dfrac{Dt\,(Dt\; Te + K2)\,(Kp+Kc)}{\delta\, K1\, \%1}, 0, \%2, 0, \%2, 0, 0, 0, 0 \\ \%2, 0, \dfrac{Dt\,(Dt\; Te + K2)\,(Kp+Kc)}{\delta\, K1\, \%1}, 0, \%2, 0, 0, 0, 0 \\ \%2, 0, \%2, 0, \dfrac{Dt\,(Dt\; Te + K2)\,(Kp+Kc)}{\delta\, K1\, \%1}, 0, 0, 0, 0 \end{bmatrix}$$

$$\%1 := Kp^2 - 2\, Kc^2 + Kp\, Kc$$

$$\%2 := -\frac{Dt\,(Dt\; Te + K2)\, Kc}{\delta\, K1\, \%1}$$

We note that $(x_2, \; x_4, \; x_6, \; u_1, \; u_2, \; u_3)^T$ is expressed in terms of x_1, x_3 and x_5 only. Thus, $(x_1, \, x_3, \, x_5)$ is also a basis of the $D = Alg[\delta^{-1}]$-module P (compare with [26]). More precisely, we have:

```
> evalm([seq([x[i](t)=ApplyMatrix(T,[seq(x[j](t),j=1..6),
> seq(u[j](t),j=1..3)],Alg)[i,1]],i=1..6)]);
```

$$\begin{bmatrix} x_1(t) = x_1(t) \\ x_2(t) = \dfrac{\mathrm{D}(x_1)(t)}{K1} \\ x_3(t) = x_3(t) \\ x_4(t) = \dfrac{\mathrm{D}(x_3)(t)}{K1} \\ x_5(t) = x_5(t) \\ x_6(t) = \dfrac{\mathrm{D}(x_5)(t)}{K1} \end{bmatrix}$$

```
> evalm([seq([u[i](t)=ApplyMatrix(T,[seq(x[j](t),j=1..6),
> seq(u[j](t),j=1..3)],Alg)[6+i,1]],i=1..3)]);
```

$$\Big[u_1(t) = \frac{K2\,(Kc+Kp)\,\mathrm{D}(x_1)(t+1)}{K1\,\%1} + \frac{Te\,(Kc+Kp)\,(\mathrm{D}^{(2)})(x_1)(t+1)}{K1\,\%1}$$
$$- \frac{K2\,Kc\,\mathrm{D}(x_3)(t+1)}{K1\,\%1} - \frac{Te\,Kc\,(\mathrm{D}^{(2)})(x_3)(t+1)}{K1\,\%1} - \frac{K2\,Kc\,\mathrm{D}(x_5)(t+1)}{K1\,\%1}$$
$$- \frac{Te\,Kc\,(\mathrm{D}^{(2)})(x_5)(t+1)}{K1\,\%1}\Big]$$

$$\Big[u_2(t) = -\frac{K2\,Kc\,\mathrm{D}(x_1)(t+1)}{K1\,\%1} - \frac{Te\,Kc\,(\mathrm{D}^{(2)})(x_1)(t+1)}{K1\,\%1} + \frac{K2\,(Kc+Kp)\,\mathrm{D}(x_3)(t+1)}{K1\,\%1}$$
$$+ \frac{Te\,(Kc+Kp)\,(\mathrm{D}^{(2)})(x_3)(t+1)}{K1\,\%1} - \frac{K2\,Kc\,\mathrm{D}(x_5)(t+1)}{K1\,\%1} - \frac{Te\,Kc\,(\mathrm{D}^{(2)})(x_5)(t+1)}{K1\,\%1}\Big]$$

$$\Big[u_3(t) = -\frac{K2\,Kc\,\mathrm{D}(x_1)(t+1)}{K1\,\%1} - \frac{Te\,Kc\,(\mathrm{D}^{(2)})(x_1)(t+1)}{K1\,\%1} - \frac{K2\,Kc\,\mathrm{D}(x_3)(t+1)}{K1\,\%1}$$
$$- \frac{Te\,Kc\,(\mathrm{D}^{(2)})(x_3)(t+1)}{K1\,\%1} + \frac{K2\,(Kc+Kp)\,\mathrm{D}(x_5)(t+1)}{K1\,\%1}$$
$$+ \frac{Te\,(Kc+Kp)\,(\mathrm{D}^{(2)})(x_5)(t+1)}{K1\,\%1}\Big]$$

$$\%1 := Kp\,Kc - 2\,Kc^2 + Kp^2$$

We refer to [26] for applications of the previous results to the motion planning and tracking problems. See also [8] for more details.

Example 10. We consider the differential time-delay system of a vibrating string with an interior mass [27]. We define the Ore algebra Alg, where D is the differential operator w.r.t. t and σ_1 and σ_2 are two non-commensurate time-delay operators. The constant parameters η_1, η_2 (composition of the mass, tensions and densities) of the system must be declared in the algebra Alg:

```
> Alg:=DefineOreAlgebra(diff=[D,t], dual_shift=[sigma1,y1],
> dual_shift=[sigma2,y2], polynom=[t,y1,y2], comm=[eta1,eta2]):
```

We only study the case of position control on both boundaries [27]. For the case of a single control, we refer to [8]. We enter the system matrix R:

```
> R:=evalm([[1,1,-1,-1,0,0],[D+eta1,D-eta1,-eta2,eta2,0,0],
> [sigma1^2,1,0,0,-sigma1,0],[0,0,1,sigma2^2,0,-sigma2]]);
```

$$R := \begin{bmatrix} 1 & 1 & -1 & -1 & 0 & 0 \\ \mathrm{D}+\eta 1 & \mathrm{D}-\eta 1 & -\eta 2 & \eta 2 & 0 & 0 \\ \sigma 1^2 & 1 & 0 & 0 & -\sigma 1 & 0 \\ 0 & 0 & 1 & \sigma 2^2 & 0 & -\sigma 2 \end{bmatrix}$$

We use an involution θ of Alg in order to obtain $R\_adj = \theta(R)$:

```
> R_adj:=Involution(R, Alg):
```

We check controllability of the system by applying Exti to R_adj:

```
> st:=time(): Ext1:=Exti(R_adj, Alg, 1): time()-st; Ext1[1];
```

$$1.191$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Since $Ext1[1]$ is the identity matrix, then we obtain that the Alg-module $M = Alg^{1\times 6}/(Alg^{1\times 4}\, R)$ associated with the system is torsion-free. This means that the vibrating string with interior mass is controllable and, equivalently, parametrizable. A parametrization of the system is then given by $Ext1[3]$:

```
> Ext1[3];
```

$$\begin{bmatrix} 2\,\sigma 2\,\eta 2\,,\ -\sigma 2\,\sigma 1\,\eta 2\,,\ -\eta 2\,\sigma 1 + \sigma 1\,\eta 1 - \sigma 1\,\mathrm{D} \\ 0\,,\ \sigma 2\,\sigma 1\,\eta 2\,,\ \eta 2\,\sigma 1 + \sigma 1\,\mathrm{D} + \sigma 1\,\eta 1 \\ \sigma 2\,\mathrm{D} + \sigma 2\,\eta 2 + \sigma 2\,\eta 1\,,\ -\sigma 2\,\sigma 1\,\eta 1\,,\ 0 \\ -\sigma 2\,\mathrm{D} + \sigma 2\,\eta 2 - \sigma 2\,\eta 1\,,\ \sigma 2\,\sigma 1\,\eta 1\,,\ 2\,\sigma 1\,\eta 1 \\ 2\,\sigma 2\,\sigma 1\,\eta 2\,,\ \sigma 2\,\eta 2 - \sigma 2\,\eta 2\,\sigma 1^2\,,\ -\eta 2\,\sigma 1^2 + \eta 2 + \eta 1\,\sigma 1^2 - \sigma 1^2\,\mathrm{D} + \mathrm{D} + \eta 1 \\ \mathrm{D} - \mathrm{D}\,\sigma 2^2 + \eta 2\,\sigma 2^2 - \eta 1\,\sigma 2^2 + \eta 2 + \eta 1\,,\ -\sigma 1\,\eta 1 + \sigma 1\,\eta 1\,\sigma 2^2\,,\ 2\,\sigma 2\,\sigma 1\,\eta 1 \end{bmatrix}$$

Therefore, the system can be parametrized by means of three free functions. We now want to check whether this parametrization is a *minimal* one [11, 33]. In order to do that, let us compute the rank of the Alg-module M.

```
> OreRank(R, Alg);
```

$$2$$

Hence, we know that there exist some parametrizations of the system with only two arbitrary functions [11, 33]. We find some minimal parametrizations:

```
> P:=MinimalParametrizations(R, Alg);
```

$$P := \left[\left[\begin{array}{cc} 2\,\sigma2\,\eta2 & -\sigma2\,\sigma1\,\eta2 \\ 0 & \sigma2\,\sigma1\,\eta2 \\ \sigma2\,\mathrm{D}+\sigma2\,\eta2+\sigma2\,\eta1 & -\sigma2\,\sigma1\,\eta1 \\ -\sigma2\,\mathrm{D}+\sigma2\,\eta2-\sigma2\,\eta1 & \sigma2\,\sigma1\,\eta1 \\ 2\,\sigma2\,\sigma1\,\eta2 & \sigma2\,\eta2-\sigma2\,\eta2\,\sigma1^2 \\ \mathrm{D}-\mathrm{D}\,\sigma2^2+\eta2\,\sigma2^2-\eta1\,\sigma2^2+\eta2+\eta1 & -\sigma1\,\eta1+\sigma1\,\eta1\,\sigma2^2 \end{array}\right],\right.$$

$$\left[\begin{array}{c} 2\,\sigma2\,\eta2\,,\ -\eta2\,\sigma1+\sigma1\,\eta1-\sigma1\,\mathrm{D} \\ 0\,,\ \eta2\,\sigma1+\sigma1\,\mathrm{D}+\sigma1\,\eta1 \\ \sigma2\,\mathrm{D}+\sigma2\,\eta2+\sigma2\,\eta1\,,\ 0 \\ -\sigma2\,\mathrm{D}+\sigma2\,\eta2-\sigma2\,\eta1\,,\ 2\,\sigma1\,\eta1 \\ 2\,\sigma2\,\sigma1\,\eta2\,,\ -\eta2\,\sigma1^2+\eta2+\eta1\,\sigma1^2-\sigma1^2\,\mathrm{D}+\mathrm{D}+\eta1 \\ \mathrm{D}-\mathrm{D}\,\sigma2^2+\eta2\,\sigma2^2-\eta1\,\sigma2^2+\eta2+\eta1\,,\ 2\,\sigma2\,\sigma1\,\eta1 \end{array}\right],$$

$$\left.\left[\begin{array}{cc} -\sigma2\,\sigma1\,\eta2 & -\eta2\,\sigma1+\sigma1\,\eta1-\sigma1\,\mathrm{D} \\ \sigma2\,\sigma1\,\eta2 & \eta2\,\sigma1+\sigma1\,\mathrm{D}+\sigma1\,\eta1 \\ -\sigma2\,\sigma1\,\eta1 & 0 \\ \sigma2\,\sigma1\,\eta1 & 2\,\sigma1\,\eta1 \\ \sigma2\,\eta2-\sigma2\,\eta2\,\sigma1^2 & -\eta2\,\sigma1^2+\eta2+\eta1\,\sigma1^2-\sigma1^2\,\mathrm{D}+\mathrm{D}+\eta1 \\ -\sigma1\,\eta1+\sigma1\,\eta1\,\sigma2^2 & 2\,\sigma2\,\sigma1\,\eta1 \end{array}\right]\right]$$

Since R has full row rank (this fact can easily be checked by computing SyzygyModule(R, Alg)), we know that M is projective, and thus, free if and only if R admits a right-inverse (see [11, 33] for more details).

```
> RightInverse(R, Alg);
```

$$[]$$

Hence, M is not projective, which implies that M is not free, i.e., the vibrating string with interior mass is not a flat system [27]. Another way to verify this fact is to compute $\mathrm{ext}^2_{Alg}(\widetilde{N}, Alg)$ and $\mathrm{ext}^3_{Alg}(\widetilde{N}, Alg)$ of the Alg-module $\widetilde{N} = Alg^{1\times 4}/(Alg^{1\times 6}\, R\_adj)$:

```
> Exti(R_adj, Alg, 2);
```

$$\left[\left[\begin{array}{ccc} 1&0&0\\0&1&0\\0&0&1 \end{array}\right],\ \left[\begin{array}{ccc} \sigma2\,\eta2 & 0 & \sigma1\,\eta1 \\ \eta2+\eta1+\mathrm{D} & -\sigma1\,\eta1 & 0 \\ 0 & \sigma2\,\eta2 & \eta2+\eta1+\mathrm{D} \end{array}\right],\ \left[\begin{array}{c} -\sigma1\,\eta1 \\ -\mathrm{D}-\eta2-\eta1 \\ \sigma2\,\eta2 \end{array}\right]\right]$$

```
> Exti(R_adj, Alg, 3);
```

$$\left[\left[\begin{array}{c} \sigma2 \\ \sigma1 \\ \eta2+\eta1+\mathrm{D} \end{array}\right],\ \left[\,1\,\right],\ \mathrm{SURJ}(1)\right]$$

We see that $\mathrm{ext}^2_{Alg}(\widetilde{N}, Alg)$ equals zero but $\mathrm{ext}^3_{Alg}(\widetilde{N}, Alg)$ is different from zero. Therefore, M is a reflexive but not a projective Alg-module. Indeed, we recall that M is reflexive (resp., projective) iff $\mathrm{ext}^i_{Alg}(\widetilde{N}, Alg)$ equals zero for $i = 1, 2$ (resp., $i = 1, 2, 3$). Let us find a polynomial π in the variable σ_1 such that the system is π-free [15, 26, 27].

```
> PiPolynomial(R, Alg, [sigma1]);
```

$$[\sigma 1]$$

Let us find a polynomial π in the variable σ_2 such that the system is π-free.

```
> PiPolynomial(R, Alg, [sigma2]);
```

$$[\sigma 2]$$

Hence, if we invert σ_1 or σ_2, i.e., we allow ourselves to have a time-advance operator, then, by definition of the π-polynomial, the system becomes flat. A flat output for this system can be computed from a left-inverse of the minimal parametrization P, where we allow σ_1 or σ_2 to appear in the denominators.

We compute the annihilator of the Alg-module $M_1 = Alg^{1\times 2}/(Alg^{1\times 6}\, P[1])$ of the minimal parametrization $P[1]$.

```
> Ann1:=AnnExti(linalg[transpose](P[1]), Alg, 1);
```

$$Ann1 := [\sigma 2]$$

Let us compute a left-inverse of the minimal parametrization $P[1]$ by allowing σ_2 to appear in the denominators.

```
> L1:=LocalLeftInverse(P[1], Ann1, Alg);
```

$$L1 := \begin{bmatrix} 0 & 0 & \frac{1}{2\,\sigma 2\,\eta 2} & \frac{1}{2\,\sigma 2\,\eta 2} & 0 & 0 \\ 0 & \frac{\sigma 1}{\sigma 2\,\eta 2} & -\frac{\sigma 1}{\sigma 2\,\eta 2} & -\frac{\sigma 1}{\sigma 2\,\eta 2} & \frac{1}{\sigma 2\,\eta 2} & 0 \end{bmatrix}$$

We easily check that $L1$ is a left-inverse of $P[1]$.

```
> Mult(L1, P[1], Alg);
```

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

If we use σ_2^{-1}, then we obtain that a flat output of the system is defined by

$$(\xi_1,\, \xi_2)^T = L1\,(\phi_1,\, \psi_1,\, \phi_2,\, \psi_2,\, u,\, v)^T,$$

where $\phi_1, \psi_1, \phi_2, \psi_2, u, v$ are the system variables [27]. Let us point out that any multiplication of $(\xi_1, \xi_2)^T$ by a unimodular matrix over the ring $\mathbb{Q}(\eta_1, \eta_2)[\frac{d}{dt}, \sigma_1, \sigma_2, \sigma_2^{-1}]^{2\times 2}$ gives a new flat output of the system. For instance, we obtain the following flat output of the system [27]:

$$\xi_1' = 2\,\eta_2\,\sigma_2\,\xi_1 = \phi_2 + \psi_2, \quad \xi_2' = \eta_2\,\sigma_2(\xi_2 + 2\,\sigma_1\,\xi_1) = \sigma_1\,\psi_1 + u.$$

We can repeat the same procedure for $P[2]$ and $P[3]$.

```
> Ann2:=AnnExti(linalg[transpose](P[2]), Alg, 1);
> Ann3:=AnnExti(linalg[transpose](P[3]), Alg, 1);
```

$$Ann2 := [\eta 2 + \eta 1 + \mathrm{D}]$$
$$Ann3 := [\sigma 1]$$

The annihilator of $P[3]$ only contains σ_1. Let us compute a flat output by allowing the time-advance operator σ_1^{-1} to appear in the basis.

```
> L3:=LocalLeftInverse(P[3], Ann3, Alg);
```

$$L3 := \begin{bmatrix} 0 & 0 & 0 & \frac{\sigma 2}{\sigma 1\,\eta 1} & 0 & -\frac{1}{\sigma 1\,\eta 1} \\ 0 & 0 & \frac{1}{2\,\sigma 1\,\eta 1} & \frac{1}{2\,\sigma 1\,\eta 1} & 0 & 0 \end{bmatrix}$$

$L3$ is a left-inverse of $P[3]$ over $\mathbb{Q}(\eta_1, \eta_2)[\frac{d}{dt}, \sigma_1, \sigma_2, \sigma_1^{-1}]$ as we can check:

```
> Mult(L3, P[3], Alg);
```

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Therefore, if we use the time-advance operator σ_1^{-1}, we obtain the flat output of the system $(\xi_1,\, \xi_2)^T = L3\,(\phi_1,\, \psi_1,\, \phi_2,\, \psi_2,\, u,\, v)^T$. Using trivial $\mathbb{Q}(\eta_1, \eta_2)[\frac{d}{dt}, \sigma_1, \sigma_2, \sigma_1^{-1}]$-linear combinations of ξ_1 and ξ_2, we then obtain that $(\xi_1' = \sigma_2\,\psi_2 - v, \quad \xi_2' = \phi_2 + \psi_2)$ is another flat output of the system over $\mathbb{Q}(\eta_1, \eta_2)[\frac{d}{dt}, \sigma_1, \sigma_2, \sigma_1^{-1}]$.

We refer to [27] for applications of the previous results to the motion planning and tracking problems [26]. See also [8] for more details and examples.

8 Conclusion

We hope to have convinced the reader of the main interest of the package OreModules for the study of the structural properties of multidimensional linear systems over Ore algebras. To our knowledge, OreModules is the first implementation of homological methods with regard to applications in control theory. We hope that OreModules will become in the future a platform for the implementation of different algorithms obtained in the literature of multidimensional linear systems (see e.g., [1, 4, 13, 16, 18, 19, 20, 31, 32, 33, 34, 35, 37, 40, 43, 47, 49] and the references therein).

References

1. Assan, J. (1999) Analyse et synthèse de l'approche géométrique pour les systèmes linéaires sur un anneau, PhD thesis. Ecole Centrale de Nantes (France).
2. Becker, T., Weispfenning, V. (1993) Gröbner Bases. A Computational Approach to Commutative Algebra. Springer.

3. Bender, C. M., Dunne, G. V., Mead, L. R. (2000) Underdetermined systems of partial differential equations. J. Mathematical Physics 41:6388–6398.
4. Bose, N. K. (1985) Multidimensional Systems Theory: Progress, Directions, and Open Problems in Multidimensional Systems. D. Reidel Publishing Company.
5. Chyzak, F. (1998) Fonctions holonomes en calcul formel. PhD thesis. Ecole Polytechnique (France).
6. Chyzak, F. *Mgfun* Project. `http://algo.inria.fr/chyzak/mgfun.html`.
7. Chyzak, F., Salvy, B. (1998) Non-commutative elimination in Ore algebras proves multivariate identities. J. Symbolic Computation 26:187–227.
8. Chyzak, F., Quadrat, A., Robertz, D. (2002). OreModules project. `http://wwwb.math.rwth-aachen.de/OreModules`.
9. Chyzak, F., Quadrat, A., Robertz, D. (2003) Linear control systems over Ore algebras: Effective algorithms for the computation of parametrizations. CDRom of the *Workshop on Time-Delay Systems* (TDS03), IFAC Workshop, INRIA Rocquencourt (France) (08-10/09/03).
10. Chyzak, F., Quadrat, A., Robertz, D. (2004) OreModules: A symbolic package for the study of multidimensional linear systems, proceedings of MTNS04, Leuven (Belgium) (05-09/07/04).
11. Chyzak, F., Quadrat, A., Robertz, D. (2005) Effective algorithms for parametrizing linear control systems over Ore algebras. Applicable Algebra in Engineering, Communication and Computing (AAECC) 16:319–376.
12. Conte, G., Perdon, A. M. (2000) Systems over rings: geometric theory and applications, Annual Reviews in Control 24:113–124.
13. Cotroneo, T. (2001) Algorithms in Behavioral Systems Theory. PhD thesis. University of Groningen (The Netherlands).
14. Fliess, M. (1991) Controllability revisited, in A. C. Antoulas (ed.) Mathematical System Theory. The influence of R. E. Kalman, Springer.
15. Fliess, M., Mounier, H. (1998) Controllability and observability of linear delay systems: an algebraic approach. ESAIM COCV 3:301–314.
16. Galkowski, K., Wood, J. (ed.) (2001) Multidimensional Signals, Circuits and Systems, Taylor and Francis.
17. Greuel, G.-M., Pfister, G. (2002) A Singular Introduction to Commutative Algebra, Springer.
18. Habets, L. (1994) Algebraic and computational aspects of time-delay systems. PhD thesis. University of Eindhoven (The Netherlands).
19. Habets, L. (1996) Computational aspects of systems over rings − reactability and stabilizability, CWI Quarterly 9:85–95.
20. Kailath, T. (1980) Linear Systems, Prentice-Hall.
21. Kalman, R. E., Falb, P. L., Arbib, M. A. (1969) Topics in Mathematical Systems Theory, McGraw-Hill.
22. Kučera, V. (1979) Discrete Linear Control: The Polynomial Equation Approach, Wiley.
23. Li, H. (2002) Non-commutative Gröbner Bases and Filtered-Graded Transfer, Lecture Notes in Mathematics 1795, Springer.
24. Malgrange, B. (1963) Systèmes à coefficients constants, Séminaire Bourbaki 1962/63, 246:1–11.
25. McConnell, J. C., Robson, J. C. (2000) Noncommutative Noetherian Rings. American Mathematical Society.
26. Mounier, H. (1995) Propriétés structurelles des systèmes linéaires à retards: aspects théoriques et pratiques. PhD Thesis. University of Orsay (France).

27. Mounier, H., Rudolph, J., Fliess, M., Rouchon, P. (1998) Tracking control of a vibrating string with an interior mass viewed as delay system. ESAIM COCV 3:315–321.
28. Oberst, U. (1990) Multidimensional constant linear systems. Acta Appl. Math. 20:1–175.
29. Pillai, H. K., Shankar, S. (1998) A behavioral approach to control of distributed systems. SIAM J. Control and Optimization 37:388–408.
30. Polderman, J. W., Willems, J. C. (1998) Introduction to Mathematical Systems Theory. A Behavioral Approach. TAM 26, Springer.
31. Pommaret, J.-F. (2001) Partial Differential Control Theory. Kluwer.
32. Pommaret, J.-F., Quadrat, A. (1998) Generalized Bezout Identity, Applicable Algebra in Engineering, Communication and Computing (AECC) 9:91–116.
33. Pommaret, J.-F., Quadrat, A. (1999) Localization and parametrization of linear multidimensional control systems. Systems and Control Letters 37:247–260.
34. Pommaret, J.-F., Quadrat, A. (1999) Algebraic analysis of linear multidimensional control systems. IMA J. Control and Optimization 16:275–297.
35. Pommaret, J.-F., Quadrat, A. (2000) Equivalences of linear control systems, proceedings of MTNS 2000, Perpignan (France), CDRom.
36. Pommaret, J.-F., Quadrat, A. (2003) A functorial approach to the behaviour of multidimensional control systems, Applied Mathematics and Computer Science 13:7–13.
37. Pommaret, J.-F., Quadrat, A. (2004) A differential operator approach to multidimensional optimal control, International Journal of Control 77:821–836.
38. Quadrat, A. (1999) Analyse algébrique des systèmes de contrôle linéaires multidimensionnels. PhD thesis. Ecole Nationale des Ponts et Chaussées (France).
39. Quadrat, A. (2005) An introduction to the algebraic theory of linear systems of partial differential equations, in preparation.
40. Quadrat, A., Robertz, D. (2005) Parametrizing all solutions of uncontrollable multidimensional linear systems. Proceedings of the 16$^{\text{th}}$ IFAC World Congress, Prague (Czech Republic) (04-08/07/05).
41. Quadrat, A., Robertz, D. (2005) On the blowing-up of stably-free behaviours. Proceedings of CDC-ECC05, Seville (Spain) (12-15/12/05).
42. Quadrat, A., Robertz, D. (2005) Constructive computation of bases of free modules over the Weyl algebras, INRIA report 5181 (`www.inria.fr/rrrt/rr-5181.html`), submitted for publication.
43. Sontag, E. (1976) Linear systems over commutative rings: a survey, Ricerche Automat. 7:1–34.
44. Rosenbrock, H. H. (1970) State Space and Multivariable Theory, J. Wiley.
45. Rotman, J. J. (1979) An Introduction to Homological Algebra. Academic Press.
46. Wonham, M. (1985) Linear Multivariable Control: a Geometric Approach, Springer.
47. Wood, J. (2000) Modules and behaviours in nD systems theory. Multidimensional Systems and Signal Processing 11:11–48.
48. Youla, D. C., Gnavi, G. (1979) Notes on n-dimensional system theory, IEEE Trans. Circuits and Systems 26:259–294.
49. Zerz, E. (2000) Topics in Multidimensional Linear Systems Theory. Lecture Notes in Control and Information Sciences 256, Springer, London.

Index