The Maple Package "Janet": II. Linear Partial Differential Equations

Yuri A. Blinkow[1], Carlos F. Cid[2], Vladimir P. Gerdt[3], Wilhelm Plesken[2], Daniel Robertz[2]

[1] Department of Mathematics and Mechanics, Saratov University, 410071 Saratov, Russia

[2] Lehrstuhl B für Mathematik, RWTH Aachen, Templergraben 64, D-52062 Aachen, Germany

[3] Laboratory of Information Technologies, Joint Institute for Nuclear Research, 141980 Dubna, Russia

Abstract

The MAPLE package "Janet" comes in two parts, the first dealing with polynomials and commutative algebra, the second with linear pdes. Here the second part is described. Amongst others it contains the first MAPLE implementation of the Janet algorithm which brings systems of linear pdes to a normal form from which a quantitative analysis of the space of power series solutions becomes possible.

1 Introduction

This is the second of two papers introducing the Maple package "Janet". Whereas the first part, [BCGPR 02], commented about polynomial systems being the special case of linear pdes with constant coeffiecients, cf. [Ple 03] for details, the present part deals with general systems of linear pdes, also based on the Involutive algorithm [GeB 98a], [GeB 98b] in the form presented in [Ge 02]. More precisely, our implementation fixes Janet separation of variables into multiplicative and non-multiplicative ones [Jan 29] as the input involutive division [GeB 98a] for the algorithm that produces a pde Janet basis in the output. By means of this technique the canonical involutive normal forms for systems of linear partial differential equations can be produced.

Below is a list of the commands available in the pde part of "Janet": With the commands in the first group one can create a Janet basis, produce normal forms of differential expressions, print out the Janet basis with some relevant extra information and get quantitative information about the free Taylor coefficients, called parametric derivatives. The usage with definitions and some typical examples and comments on the algorithms will be given in section 2. The section 3 comments on compatibility conditions for right hand sides of the equations or, equivalently in the language of modules, on syzygies and free resolutions for modules over the ring of differential operators. Finally the last section 4 gives further examples. The first one discusses on the theoretical side a LIE algebra technique to construct all polynomial solutions of linear pdes with a big symmetry LIE algebra and on the practical side the interaction of the present package with the Maple package "jets", cf. [Bar 01], [BaH 02]. The second example is on differential elimination, the third on finding autonomous elements (in the context of contol theory).

We have taken big effort to produce detailed online documentation in the form of help pages with examples, so that we can be brief in this account. Some functions were taken over from "jets"; in particular it is possible to use the more convenient jet notation to some extend, as will be demonstrated in this paper. It is also possible to call the pde-solver function of the Maple package "DESOLV", cf. [VuC 00] with the output of JanetBasis as input via the function Jpdesolv. In fact, it was one of the early successes of our JANET-package, that various big systems of linear pdes coming up in the symmetry analysis of various nonlinear pde-systems could only be solved by "pdesolv' after they had been pro-cesses by the JANET algorithm.

Basic commands:	
JanetBasis	InvReduce
PrincDeriv (=TabVar)	HilbertSeries
SolSeries	PolySol
ParamDeriv	ZeroSets
Commands for special applications	
CompCond	Resolution
Autonom (=Torsion)	SyzOp
WeightedHilbertSeries	
Commands for various invariants derivable from HilbertSeries:	
IndexRegularity	CartanCharacter
HilbertPolynomial	HilbertFunction
HP	HF
Auxiliary Commands:	
LeadingDeriv	Jpdesolv
AffEqn	AssertJanetBasis
Diff2Pol	Pol2Diff
CmpOp	JAdjoint
Diff2Op	AppOp
Ind2Diff	Diff2Ind
Pol2Ind	AppOpInd

2 Basics

Let F be a field of meromorphic functions in $x_1, \ldots, x_n$ over a field of constants $K \subseteq \mathbb{C}$ closed under all partial derivatives $\frac{\partial}{\partial x_i}$ for $i = 1, \ldots, n$, e. g. $F = \mathbb{Q}(x_1, \ldots, x_n)$, and let $R := F\langle \frac{\partial}{\partial x_1}, \ldots, \frac{\partial}{\partial x_n} \rangle$ be a ring of operators on F generated by F and the partial derivatives $\frac{\partial}{\partial x_i}$ for $i = 1, \ldots, n$, where F acts on itself by multiplication. Clearly, R is non commutative and any element of R can be written in the form

$$\sum_i a_i \frac{\partial^{|i|}}{\partial x_1^{i_1} \cdots \partial x_n^{i_n}}$$

with $a_i = a_i(x_1, \ldots, x_n) \in F$ and i running through a finite subset of $(\mathbb{Z}_{\geq 0})^n$ with $|i| := \sum_{j=1}^n i_j$. In general K can be any subfield of $\mathbb{C}$, in which one has a normal form for elements and where addition, subtraction, multiplication, and division can constructively be carried out. For our MAPLE implementation it means $K = \mathbb{Q}$ or a finitely generated field extension of $\mathbb{Q}$. Given q-tuples $A_1, \ldots, A_a \in R^{1\times q}$ we are concerned with the linear system of pdes given by

$$A_i u = 0 \qquad (i = 1, \ldots, a) \quad \text{with} \quad u := \begin{pmatrix} u_1 \\ \vdots \\ u_q \end{pmatrix} \tag{1}$$

for the unknown functions $u_1, \ldots, u_q$ in $x_1, \ldots, x_n$ (which one may think of as dependent variables). The aim is to obtain quantitative control over the power series solutions of (1). At this stage a trivial, nevertheless important remark will provide the bridge between the polynomial case in part 1, cf. [BCGPR 02] and the present case.

Remark 2.1

$$R^{1\times q} \to \oplus_{i=1}^q R u_i \quad : \quad Z \mapsto Zu$$

is an isomorphism of left R-modules. This isomorphism maps the R-submodule

$$S := \langle A_1, \ldots, A_a \rangle \leq R^{1\times q}$$

of $R^{1\times q}$ spanned by the A_i onto the submodule S' of $\oplus_{i=1}^q R u_i$ which are consequences of the pde system (1). In particular, the R-factor module

$$M := R^{1\times q}/S$$

becomes identified with the (analytically more familiar) R-module

$$M' := \oplus_{i=1}^q R u_i / S',$$

whose presentation is given by (1).

So, when we write expressions in MAPLE, we seem to write elements in $\oplus_{i=1}^q Ru_i$, but we need to know them modulo S', i. e. as elements of M'. It will soon become clear that the knowledge of M' is really the (quantive, formal) control over the power series solutions of (1), which is our declared aim. It is achieved by the concepts of JANET basis, multiplicative variables, and parametric derivatives. The JANET basis B consists of finitely many elements $B_1, \ldots, B_d \in R^{1\times q}$, whose equations taken together have the same set of power series solutions as the original equations (1). Moreover, each B_i has attached multiplicative variables from among the x_i to it, such that each equation that is a consequence of (1) can be written uniquely as a F-linear combination of the derivatives of the B_i with respect to B_i-multiplicative variables. In more algebraic terms: The R-submodule S of the free R-module $R^{1\times q}$ spanned by $A_1, \ldots, A_a$ is also spanned by $B_1, \ldots, B_d$ in such a way that the

$$\frac{\partial^{|i|}}{\partial x_1^{i_1}\cdots\partial x_n^{i_n}} B_l$$

where i runs through the subset of those elements of $(\mathbb{Z}_{\geq 0})^n$ with $i_s = 0$, whenever x_s is not multiplicative for B_l, form an F-basis (in the sense of vector spaces) for this submodule. This allows a close comparison to the polynomial case, cf. [BCGPR 02]. The so called monomial basis in the polynomial case corresponds to parametric derivatives here. $\frac{\partial^{|i|}u_l}{\partial x_1^{i_1}\cdots\partial x_n^{i_n}}$ is called a parametric derivative, if it does not occur as the leading derivative in any of the equations $Bu = 0$, where B is derivative of any of the B_i. All other derivatives of the u_l are called principal. The concept of involutive division allows to express any principal derivative in terms of the parametric derivatives. The resulting expression is unique. Concerning the ordering of the derivatives of the u_l the same possibilities as in the polynomial case are realized, the degree reverse lexicographic option with the additional option of favouring the component with the highest degree is usually chosen. Here is a rough description of in- and output:

Input: $A_1, \ldots, A_a \in R^q$ generating the submodule S of R^q (and a list of indeterminants, e. g. $x_1, \ldots, x_n$.)
Output: The JANET basis $B_1, \ldots, B_d$ of S by the call JanetBasis.
Subsequent commands:
TabVar or equivalently PrincDeriv reproduces each B_i, the leading term of B_i and the subset $M_i \subseteq \{x_1, \ldots, x_n\}$ of multiplicative variables of B_i with respect to B, i. e. each element s of S has a unique representation as

$$s = \sum_{i=1}^{d} p_i B_i \text{ with } p_i \in F\langle \frac{\partial}{\partial x_i} | x_i \in M_i \rangle.$$

The second name for the command explains itself from the fact that the output allows to read off all principal derivatives, i. e. those derivatives, which do not occur in a reduced differential expression. They are given by the derivatives of the highest terms of the B_i with respect to B_i-multiplicative variables.
ParamDeriv enumerates all parametric derivatives.

HilbertSeries gives the generating function for the numbers of the parametric derivatives according to their order.
With further **input** $v \in \oplus_{i=1}^{q} Ru_i$ the command InvReduce produces the normalized representative v' of the coset $v + S' \in M'$, i. e. an expression involving only parametric derivatives. E. g. if v is some partial higher derivative of some u_i, then v gets expressed in terms of parametric derivatives.
ZeroSets describes the points which cannot be taken as center of a power series expansion for the solutions, i. e. lists the functions by which one has divided in the course of the algorithm.
SolSeries computes the power series solutions of (1) up to an order given as (additional) input.
PolySol computes the polynomial solutions of (1) up to a degree given as (additional) input.

The first example deals with linear pdes with constant coefficients. It is completely parallel to Example (2.2) of the first part [BCGPR 02]), cf also [Ple 03] for details on the connection between linear pdes with constant coefficients and polynomial equations. All the functions above are demonstrated except for ZeroSets, because in the constant coefficients case one need not devide by non constant functions.

Example 2.2 *(cf. Example (2.2) of [BCGPR 02]):*
Specification of independent and dependent variables:

```
> ivar:=[x,y];dvar:=[u,v];
```

$$ivar := [x, y]$$
$$dvar := [u, v]$$

Tuples of polynomials and their translation into linear differential expressions with constant coefficients, which constitute the pde system:

```
> l:=[[x,-y],[y,x],[y^3,0]];
```

$$l := [[x, -y], [y, x], [y^3, 0]]$$

```
> L:= Pol2Diff(l,ivar,dvar);
```

$$L := [(\frac{\partial}{\partial x}\,\mathrm{u}(x, y)) - (\frac{\partial}{\partial y}\,\mathrm{v}(x, y)),\, (\frac{\partial}{\partial y}\,\mathrm{u}(x, y)) + (\frac{\partial}{\partial x}\,\mathrm{v}(x, y)),\, \frac{\partial^3}{\partial y^3}\,\mathrm{u}(x, y)]$$

Computation of the Janet basis:

```
> J:=JanetBasis(L,ivar,dvar);
```

$$J := [[(\frac{\partial}{\partial y}\,\mathrm{u}(x, y)) + (\frac{\partial}{\partial x}\,\mathrm{v}(x, y)),\, (\frac{\partial}{\partial x}\,\mathrm{u}(x, y)) - (\frac{\partial}{\partial y}\,\mathrm{v}(x, y)),\, \frac{\partial^3}{\partial y^3}\,\mathrm{u}(x, y),\, \frac{\partial^4}{\partial y^4}\,\mathrm{v}(x, y)],\, [x, y],\, [u, v]]$$

Details on the Janet basis: first the basis vector (equation), next the multiplicative variables with the numbers referring to ivar and stars indicating non multiplicative variables, and finally the highest term, thus allowing to read off the principal derivatives. There are four elements in the Janet basis.

```
> PrincDeriv();
```

$$[(\frac{\partial}{\partial y}\,\mathrm{u}(x, y)) + (\frac{\partial}{\partial x}\,\mathrm{v}(x, y)),\, [1, 2],\, \frac{\partial}{\partial x}\,\mathrm{v}(x, y)]$$

$$[(\frac{\partial}{\partial x}\,\mathrm{u}(x,\,y)) - (\frac{\partial}{\partial y}\,\mathrm{v}(x,\,y)),\,[1,\,2],\,\frac{\partial}{\partial x}\,\mathrm{u}(x,\,y)]$$

$$[\frac{\partial^3}{\partial y^3}\,\mathrm{u}(x,\,y),\,[*,\,2],\,\frac{\partial^3}{\partial y^3}\,\mathrm{u}(x,\,y)]$$

$$[\frac{\partial^4}{\partial y^4}\,\mathrm{v}(x,\,y),\,[*,\,2],\,\frac{\partial^4}{\partial y^4}\,\mathrm{v}(x,\,y)]$$

The parametric derivatives and their generating function, the Hilbert series:

```
> ParamDeriv(ivar,dvar);
```

$$[\mathrm{v}(x,\,y),\,\frac{\partial}{\partial y}\,\mathrm{v}(x,\,y),\,\frac{\partial^2}{\partial y^2}\,\mathrm{v}(x,\,y),\,\frac{\partial^3}{\partial y^3}\,\mathrm{v}(x,\,y),\,\mathrm{u}(x,\,y),\,\frac{\partial}{\partial y}\,\mathrm{u}(x,\,y),\,\frac{\partial^2}{\partial y^2}\,\mathrm{u}(x,\,y)]$$

```
> HilbertSeries(t);
```

$$2 + 2\,t + 2\,t^2 + t^3$$

Normalizing differential expressions:

```
> InvReduce(diff(u(x,y),x,x)+diff(v(x,y),y,y),J);
```

$$-(\frac{\partial^2}{\partial y^2}\,\mathrm{u}(x,\,y)) + (\frac{\partial^2}{\partial y^2}\,\mathrm{v}(x,\,y))$$

Commands without analogues in the polynomial case: Computing the Taylor expansion of the power series solutions up to a given order (3 in this case) and computing polynomial solutions up to a given degree (also 3 in this case). That there are 7 free parameters for the Taylor expansion was to be expected from the Hilbert series. That all the expansions up to order three are already solutions is new information.

```
> SolSeries(J,3,'SO');
```

$$[\mathrm{u}(x,\,y) = C1_{0,0} + C2_{0,1}\,x + C1_{0,1}\,y - \frac{1}{2}\,C1_{0,2}\,x^2 + C2_{0,2}\,x\,y + \frac{1}{2}\,C1_{0,2}\,y^2 - \frac{1}{6}\,C2_{0,3}\,x^3 + \frac{1}{2}\,C2_{0,3}\,x\,y^2, \mathrm{v}(x,\,y) = C2_{0,0} - C1_{0,1}\,x + C2_{0,1}\,y - \frac{1}{2}\,C2_{0,2}\,x^2 - C1_{0,2}\,x\,y + \frac{1}{2}\,C2_{0,2}\,y^2 - \frac{1}{2}\,C2_{0,3}\,x^2\,y + \frac{1}{6}\,C2_{0,3}\,y^3]$$

```
> SO;
```

$$[[\mathrm{u}(x,\,y) = 1,\,\mathrm{v}(x,\,y) = 0],\,[\mathrm{u}(x,\,y) = x,\,\mathrm{v}(x,\,y) = y],\,[\mathrm{u}(x,\,y) = y,\,\mathrm{v}(x,\,y) = -x],\,[\mathrm{u}(x,\,y) = -\frac{1}{2}\,x^2 + \frac{1}{2}\,y^2,\,\mathrm{v}(x,\,y) = -x\,y],\,[\mathrm{u}(x,\,y) = x\,y,\,\mathrm{v}(x,\,y) = -\frac{1}{2}\,x^2 + \frac{1}{2}\,y^2],\,[\mathrm{u}(x,\,y) = -\frac{1}{6}\,x^3 + \frac{1}{2}\,x\,y^2,\,\mathrm{v}(x,\,y) = -\frac{1}{2}\,x^2\,y + \frac{1}{6}\,y^3],\,[\mathrm{u}(x,\,y) = 0,\,\mathrm{v}(x,\,y) = 1]]$$

```
> PolySol(J,3,'P'):
> evalb(P=SO);
```

true

If in the previous example the linear pde system with constant coefficients would have been really big, one could proceed as follows: Find the *Janet* basis for the polynomial system using InvolutiveBasisFast,cf. [BCGPR 02], rewrite the polynomial *Janet* basis as pde system using Pol2Diff, and tell the system that this is a pde *Janet* basis by invoking AssertJanetBasis.

The next example deals with non constant coefficients, also demonstrating the difficulties arising from division by non constant functions.

Example 2.3 *($n = 3, q = 3$, non constant coefficients) The following system describes the set of all vectorfields in 3-space commuting with the infinitesimal rotations around the z- and the y-axis:*

```
> ivar:=[x,y,z]; dvar:=[s,t,u];
```

$$ivar := [x, y, z]$$

$$dvar := [s, t, u]$$

```
> L := [y*diff(s(x,y,z),x)-x*diff(s(x,y,z),y)-t(x,y,z),
> y*diff(t(x,y,z),x)-x*diff(t(x,y,z),y)+s(x,y,z),
> y*diff(u(x,y,z),x)-x*diff(u(x,y,z),y),
> z*diff(s(x,y,z),x)-x*diff(s(x,y,z),z)-u(x,y,z),
> z*diff(t(x,y,z),x)-x*diff(t(x,y,z),z),
> z*diff(u(x,y,z),x)-x*diff(u(x,y,z),z)+s(x,y,z)];
```

$$L := [y\,(\tfrac{\partial}{\partial x}\,\mathrm{s}(x, y, z)) - x\,(\tfrac{\partial}{\partial y}\,\mathrm{s}(x, y, z)) - \mathrm{t}(x, y, z),$$
$$y\,(\tfrac{\partial}{\partial x}\,\mathrm{t}(x, y, z)) - x\,(\tfrac{\partial}{\partial y}\,\mathrm{t}(x, y, z)) + \mathrm{s}(x, y, z),\ y\,(\tfrac{\partial}{\partial x}\,\mathrm{u}(x, y, z)) - x\,(\tfrac{\partial}{\partial y}\,\mathrm{u}(x, y, z)),$$
$$z\,(\tfrac{\partial}{\partial x}\,\mathrm{s}(x, y, z)) - x\,(\tfrac{\partial}{\partial z}\,\mathrm{s}(x, y, z)) - \mathrm{u}(x, y, z),\ z\,(\tfrac{\partial}{\partial x}\,\mathrm{t}(x, y, z)) - x\,(\tfrac{\partial}{\partial z}\,\mathrm{t}(x, y, z)),$$
$$z\,(\tfrac{\partial}{\partial x}\,\mathrm{u}(x, y, z)) - x\,(\tfrac{\partial}{\partial z}\,\mathrm{u}(x, y, z)) + \mathrm{s}(x, y, z)]$$

```
> J:=JanetBasis(L,ivar,dvar);
```

$$J := [[\frac{z\,\mathrm{t}(x, y, z)}{-z+y} - \frac{y\,\mathrm{u}(x, y, z)}{-z+y},\ -\frac{z\,\mathrm{s}(x, y, z)}{z-x} + \frac{x\,\mathrm{u}(x, y, z)}{z-x},$$
$$y\,\mathrm{u}(x, y, z) + z^2\,(\tfrac{\partial}{\partial y}\,\mathrm{u}(x, y, z)) - y\,z\,(\tfrac{\partial}{\partial z}\,\mathrm{u}(x, y, z)),$$
$$x\,\mathrm{u}(x, y, z) - x\,z\,(\tfrac{\partial}{\partial z}\,\mathrm{u}(x, y, z)) + z^2\,(\tfrac{\partial}{\partial x}\,\mathrm{u}(x, y, z))],\ [x, y, z],\ [s, t, u]]$$

```
> TabVar();
```

$$[\frac{z\,\mathrm{t}(x, y, z)}{-z+y} - \frac{y\,\mathrm{u}(x, y, z)}{-z+y},\ [1, 2, 3],\ \frac{z\,\mathrm{t}(x, y, z)}{-z+y}]$$
$$[-\frac{z\,\mathrm{s}(x, y, z)}{z-x} + \frac{x\,\mathrm{u}(x, y, z)}{z-x},\ [1, 2, 3],\ -\frac{z\,\mathrm{s}(x, y, z)}{z-x}]$$
$$[y\,\mathrm{u}(x, y, z) + z^2\,(\tfrac{\partial}{\partial y}\,\mathrm{u}(x, y, z)) - y\,z\,(\tfrac{\partial}{\partial z}\,\mathrm{u}(x, y, z)),\ [*, 2, 3],\ z^2\,(\tfrac{\partial}{\partial y}\,\mathrm{u}(x, y, z))]$$
$$[x\,\mathrm{u}(x, y, z) - x\,z\,(\tfrac{\partial}{\partial z}\,\mathrm{u}(x, y, z)) + z^2\,(\tfrac{\partial}{\partial x}\,\mathrm{u}(x, y, z)),\ [1, 2, 3],\ z^2\,(\tfrac{\partial}{\partial x}\,\mathrm{u}(x, y, z))]$$

```
> ZeroSets();
```

$$[[y, \{y=0\}], [x\,z, \{z=0, x=0\}], [z, \{z=0\}], [z\,(y-x), \{z=0, x=y\}], [x, \{x=0\}],$$
$$[y\,z, \{z=0, y=0\}], [-y\,z+y^2, \{y=0, y=z\}], [y\,(z-x), \{y=0, x=z\}],$$
$$[z^2, \{z=0\}]]$$

```
> HilbertSeries(lambda);
```

$$1 + \frac{\lambda}{1-\lambda}$$

```
> ParamDeriv(ivar,dvar);
```

$$[0,\ 0,\ \frac{1}{1-z}]$$

Because of the result of ZeroSets one cannot expand in (0,0,0). We choose (1,2,3) instead:

```
> PolySol(J,5,[1,2,3],'d'):
> d;
```

$$[[\mathrm{s}(x,\,y,\,z)=x,\,\mathrm{t}(x,\,y,\,z)=y,\,\mathrm{u}(x,\,y,\,z)=z],[\mathrm{s}(x,\,y,\,z)=-\frac{16}{3}\,x+\frac{1}{6}\,x\,y^2+\frac{1}{6}\,x^3+\frac{1}{6}\,x\,z^2,$$
$$\mathrm{t}(x,\,y,\,z)=-\frac{16}{3}\,y+\frac{1}{6}\,y^3+\frac{1}{6}\,x^2\,y+\frac{1}{6}\,y\,z^2,\,\mathrm{u}(x,\,y,\,z)=-\frac{16}{3}\,z+\frac{1}{6}\,y^2\,z+\frac{1}{6}\,x^2\,z+\frac{1}{6}\,z^3],[$$
$$\mathrm{s}(x,\,y,\,z)=\frac{125}{6}\,x+\frac{1}{120}\,x^5+\frac{1}{60}\,x^3\,z^2+\frac{1}{60}\,x^3\,y^2+\frac{1}{120}\,x\,y^4+\frac{1}{120}\,x\,z^4+\frac{1}{60}\,x\,y^2\,z^2$$
$$-\frac{5}{6}\,x\,y^2-\frac{5}{6}\,x^3-\frac{5}{6}\,x\,z^2,\mathrm{t}(x,\,y,\,z)=\frac{125}{6}\,y+\frac{1}{120}\,y\,z^4+\frac{1}{120}\,x^4\,y+\frac{1}{60}\,x^2\,y\,z^2$$
$$+\frac{1}{60}\,x^2\,y^3+\frac{1}{120}\,y^5-\frac{5}{6}\,y^3-\frac{5}{6}\,x^2\,y-\frac{5}{6}\,y\,z^2+\frac{1}{60}\,y^3\,z^2,\mathrm{u}(x,\,y,\,z)=\frac{125}{6}\,z+\frac{1}{120}\,z^5$$
$$+\frac{1}{120}\,x^4\,z+\frac{1}{60}\,x^2\,y^2\,z+\frac{1}{60}\,x^2\,z^3+\frac{1}{120}\,y^4\,z-\frac{5}{6}\,y^2\,z+\frac{1}{60}\,y^2\,z^3-\frac{5}{6}\,x^2\,z-\frac{5}{6}\,z^3]]$$

Note, because of the Hilbert series, SolSeries would have produced 5 independent expansions of solutions. Hence not all solutions can be expanded by polynomial solutions. But since all the functions, whose Lie derivatives with the two infinitesimal rotations are functions of sqrt(x^2+y^2+z^2) (use the Janet-program to prove this!), the above Hilbert series tells us, that the R[sqrt(x^2+y^2+z^2](xD_x+yD_y+zD_z) is dense in the space of all (outside 0 analytic) vector fields commuting with the two (and hence all) infinitesimal rotations.

Whatever has been said in part 1, [BCGPR 02], about orderings and gradings for the variables applies in the present case as well. The role of the various components in the polynomial case has been taken over by the dependent variables. Again term over position is usually much more effective. Unfortunately there is no C++-implementation for the present case available.

The way differential expressions are witten in MAPLE is rather clumsy for typing input as seen in the last example, whereas the constant coefficient case is dealt with in a satisfactory manner as demonstrated in the first example. Therefore we have taken over the jet notation from the MAPLE package "jets", cf. [Bar 01], [BaH 02], and provide two functions Ind2Diff and Diff2Ind to translate jet expressions into differential expressions and back again. For instance, if u is a dependent variable and x, y, z are the independent variables, then the jet variable u_{xxyz} stands for the derivative in MAPLE notation $\frac{\partial^4 u(x,y,z)}{\partial x^2 \partial y \partial z}$. In the subsequent examples this more convenient way for producing input will be used.

3 Compatibility conditions and Syzygies

In this section we discuss (local) compatibility conditions for right hand sides of linear pdes, a well known example being the characterization of gradients via the start of the POINCARÉ sequence. The way one goes about it, is to introduce a name for each right hand side of an equation and to get a compatibiltiy condition each time the left hand side gets zero in the JANET algorithm. Here is an example, for which JANET used already the corresponding homogeneous system for demonstrating his algorithm:

Example 3.1 *We first define the system to be investigated by using jet notation:*

```
> ivar:=[x,y,z]:dvar:=[u]:
> Lj:=[u[z,z]-y*u[x,x],u[y,y]];
```

$$Lj := [u_{z,z} - y\, u_{x,x},\ u_{y,y}]$$

The aim is to check for which right hand sides the system

```
> Lh:=Jind2eqn(Lj,ivar,dvar);
```

$$Lh := [(\tfrac{\partial^2}{\partial z^2}\,\mathrm{u}(x,\,y,\,z)) - y\,(\tfrac{\partial^2}{\partial x^2}\,\mathrm{u}(x,\,y,\,z)),\ \tfrac{\partial^2}{\partial y^2}\,\mathrm{u}(x,\,y,\,z)]$$

has solutions. Therefore we introduce names a(x,y,z) and b(x,y,z) for the right hand sides as follows and compute a Janet basis for the resulting system in the usual manner (by carrying the new functions along):

```
> L:=AffEqn(Lh,ivar,[a,b]);
```

$$L := [(\tfrac{\partial^2}{\partial z^2}\,\mathrm{u}(x,\,y,\,z)) - y\,(\tfrac{\partial^2}{\partial x^2}\,\mathrm{u}(x,\,y,\,z)) - \mathrm{a}(x,\,y,\,z),\ (\tfrac{\partial^2}{\partial y^2}\,\mathrm{u}(x,\,y,\,z)) - \mathrm{b}(x,\,y,\,z)]$$

```
> JL:=JanetBasis(L,ivar,dvar):
```

Whenever a left hand side becomes zero, one gets a compatibility condition. They are collected in the global variable COMPA. Other compatibility conditions come from expressing the original equations in L in terms of the Janet basis and from reducing prolongations of the elements of the Janet basis by non-multiplicative variables. All these can be obtained with the command CompCond. To save space, the output is turned into jet notation via the command Jeqn2ind:

```
> Jeqn2ind(CompCond(L,ivar,dvar),ivar,[a,b]);
```

$$\begin{aligned}
&[\frac{1}{2}\,b_{x,y,z,z}\,y^2 - \frac{1}{2}\,a_{x,y,y,y}\,y^2 - \frac{1}{2}\,b_{x,x,x,y}\,y^3 - \frac{3}{2}\,b_{x,x,x}\,y^2,\ \frac{1}{2}\,b_{z,z,z,z,z,z}\,y - \frac{1}{2}\,a_{y,y,z,z,z,z}\,y\\
&- \frac{3}{2}\,b_{x,x,z,z,z,z}\,y^2 - a_{x,x,y,z,z}\,y + a_{x,x,y,y,z,z}\,y^2 + \frac{3}{2}\,b_{x,x,x,x,z,z}\,y^3 - a_{x,x,x,x}\,y\\
&+ a_{x,x,x,x,y}\,y^2 - \frac{1}{2}\,a_{x,x,x,x,y,y}\,y^3 - \frac{1}{2}\,b_{x,x,x,x,x,x}\,y^4,\ \frac{3}{2}\,b_{x,x,x,z,z}\,y^2 - \frac{3}{2}\,b_{x,x,x,x,x}\,y^3\\
&- \frac{1}{2}\,b_{x,y,z,z,z,z}\,y^2 + \frac{1}{2}\,a_{x,y,y,y,z,z}\,y^2 + b_{x,x,x,y,z,z}\,y^3 - \frac{1}{2}\,a_{x,x,x,y,y,y}\,y^3\\
&- \frac{1}{2}\,b_{x,x,x,x,x,y}\,y^4,\ \frac{1}{2}\,b_{y,z,z}\,y^2 - \frac{1}{2}\,a_{y,y,y}\,y^2 - \frac{1}{2}\,b_{x,x,y}\,y^3 - \frac{3}{2}\,b_{x,x}\,y^2,\ \frac{3}{2}\,b_{x,x,z,z}\,y^2\\
&- \frac{3}{2}\,b_{x,x,x,x}\,y^3 - \frac{1}{2}\,b_{y,z,z,z,z}\,y^2 + \frac{1}{2}\,a_{y,y,y,z,z}\,y^2 + b_{x,x,y,z,z}\,y^3 - \frac{1}{2}\,a_{x,x,y,y,y}\,y^3\\
&- \frac{1}{2}\,b_{x,x,x,x,y}\,y^4]
\end{aligned}$$

```
> HilbertSeries(t);
```

$$1+3\,t+4\,t^2+3\,t^3+t^4$$

Summerizing, we have a 12-dimensional affine space of solutions, whenever a(x,y,z) and b(x,y,z) satisfy the equations above. The space of solutions of the homogeneous system can easily be computed with JanetBasis and PolySol.

Of course, one can also check for given specific right hand sides, whether the system allows solutions in the same way: The command CompCond should produce zeroes only, resp. the empty list. In this case, the commands SolSeries and PolySol can be used as in the homogeneous case to look at expansions of solutions and polynomial solutions.

It is worthwhile to have a look at the compatibility from the module point of view. Note, $M = R^{1\times q}/S$ can be viewed as the cokernel of the homomorphism

$$\alpha : R^{1\times a} \to R^{1\times q} : z \mapsto zA$$

of left R-modules, where the rows of $A \in R^{a\times q}$ are $A_1, \ldots, A_a$. Another way of viewing the command CompCond is that it computes the kernel of α. More precisely, if CompCond is performed after the command JanetBasis on input with general right hand side as above, the result can be interpreted as a homomorphism

$$\beta : R^{1\times b} \to R^{1\times a} : z \mapsto zB$$

where $B \in R^{b\times a}$ has rows corresponding (in the sense of 2.1) to the elements of the output of CompCond. In particular

$$R^{1\times b} \xrightarrow{\beta} R^{1\times a} \xrightarrow{\alpha} R^{1\times q} \xrightarrow{\nu} M \to 0,$$

where ν is the natural epimorphism, is the beginning of a free resolution of M as left R-module.

Until now it was not necessary to introduce a proper notation for the elements of the ring R, since we got away with describing the result of applying them to a general function. With the matrices now, we need a proper notation, which is again adopted form the jets-package, cf. [Bar 01], [BaH 02], and goes as follows: An element of R is always written in square brackets, which are part of the name. Each summand $a_i \frac{\partial^{|i|}}{\partial x_1^{i_1} \cdots \partial x_n^{i_n}}$ is written inside these square barackets as $[a_i, [\underbrace{x_1, \ldots, x_1}_{i_1}, \ldots, \underbrace{x_n, \ldots, x_n}_{i_n}]]$. These terms are seperated by commas. Things will become clear in the next example. Matrices over R can be constructed inside the package from tuples of differential expressions with the command Diff2Ind, the reverse command being Ind2Diff. ("Ind" stands for index notation for jet expressions.)

Continuing with the discussion of the matrices for α and β above, one can start all over with the matrix B and iterate to construct a free resolution of M. But there is a way to construct the resolution in one go by using the JANET basis as relations for M. This resolution is necessarily finite, more precisely it finishes after k steps, if k is the maximum number of non-multiplicative variables for the members of the JANET basis. An example follows below. There is another version of this resolution producing only with first order differential operators for the higher syzygies, cf. [Pom 94], which can also realized in the package with the command SyzOp, which works only with operator input and will not be demonstrated here.

Example 3.2 *We start again with the following system:*

```
> L:=[exp(y)*diff(u(x,y,z),x)-y^2*diff(u(x,y,z),y,z),
> diff(u(x,y,z),x,z)];
```

$$L := [e^y\,(\frac{\partial}{\partial x}\,\mathrm{u}(x,\,y,\,z)) - y^2\,(\frac{\partial^2}{\partial z\,\partial y}\,\mathrm{u}(x,\,y,\,z)),\ \frac{\partial^2}{\partial z\,\partial x}\,\mathrm{u}(x,\,y,\,z)]$$

and specify the independent and dependent variables:

```
> ivar:=[x,y,z]:dvar:=[u]:
> Resolution(L,ivar,dvar);
```

$$\left[\begin{bmatrix} 0 & [[1,\,[x]]] & [[-1,\,[z]]] \\ [[1,\,[x]]] & [[y^2,\,[y]]] & [[-e^y,\,[]]] \end{bmatrix},\ \begin{bmatrix} [[-y^2,\,[y,\,z]],\,[e^y,\,[x]]] \\ [[1,\,[x,\,z]]] \\ [[1,\,[x,\,x]]] \end{bmatrix}\right]$$

The column on the right is to be compared with the Janet basis of L:

```
> JanetBasis(L,ivar,dvar);
```

$$[[e^y\,(\frac{\partial}{\partial x}\,\mathrm{u}(x,\,y,\,z)) - y^2\,(\frac{\partial^2}{\partial z\,\partial y}\,\mathrm{u}(x,\,y,\,z)),\ \frac{\partial^2}{\partial z\,\partial x}\,\mathrm{u}(x,\,y,\,z),\ \frac{\partial^2}{\partial x^2}\,\mathrm{u}(x,\,y,\,z)],\ [x,\,y,\,z],\ [u]]$$

4 Further Examples

The JANET package can very well be used together with the package “jets”, which can perform all sorts of jet calculations. Here is an example:

Example 4.1 *The aim is to construct recursively all polynomial solution of the heat equation by using its symmetry Lie algebra.*

We use the package "jets" to set up the equation for the Lie algebra. Here we define the independent and dependent variables and write down the equation in "jet" notation:

```
> ivar:=[x,t]; dvar:=[u];
```

$$ivar := [x,\,t]$$

$$dvar := [u]$$

```
> eq:=[u[x,x]+u[t]];
```

$$eq := [u_{x,x} + u_t]$$

We will only be interested in symmetry vector fields of the following form:

```
> defvec("lin",ivar,dvar);
```

$$[[[\xi_x(x,\,t),\,[x]],\,[\xi_t(x,\,t),\,[t]],\,[u\,\eta_{u,u}(x,\,t),\,[u]]],\,[x,\,t],\,[\xi_x(x,\,t),\,\xi_t(x,\,t),\,\eta_{u,u}(x,\,t)]]$$

Here is the "jets"-command to set up the equations

```
> le:=gengen(eq,"lin",ivar,dvar);
```

$$le := [[\frac{\partial}{\partial x}\,\xi_t(x,\,t),\,(\frac{\partial}{\partial t}\,\eta_{u,\,u}(x,\,t)) + (\frac{\partial^2}{\partial x^2}\,\eta_{u,\,u}(x,\,t)),$$
$$-(\frac{\partial^2}{\partial x^2}\,\xi_x(x,\,t)) + 2\,(\frac{\partial}{\partial x}\,\eta_{u,\,u}(x,\,t)) - (\frac{\partial}{\partial t}\,\xi_x(x,\,t)),$$
$$2\,(\frac{\partial}{\partial x}\,\xi_x(x,\,t)) - (\frac{\partial}{\partial t}\,\xi_t(x,\,t)) - (\frac{\partial^2}{\partial x^2}\,\xi_t(x,\,t))],\,[x,\,t],\,[\xi_x(x,\,t),\,\xi_t(x,\,t),\,\eta_{u,\,u}(x,\,t)],$$
$$[[x,\,t,\,u],\,[x,\,t,\,u],\,[],\,[\xi_x(x,\,t),\,\xi_t(x,\,t),\,u\,\eta_{u,\,u}(x,\,t)]]]$$

We get a Janet basis for these equations using the "Janet" package and ask for the number of solutions:

```
> Jle:=JanetBasis(op(le[1..3]));
```

$$Jle := [[-(\frac{\partial}{\partial t}\,\xi_x(x,\,t)) + 2\,(\frac{\partial}{\partial x}\,\eta_{u,\,u}(x,\,t)),\,\frac{\partial}{\partial x}\,\xi_t(x,\,t),\,2\,(\frac{\partial}{\partial x}\,\xi_x(x,\,t)) - (\frac{\partial}{\partial t}\,\xi_t(x,\,t)),$$
$$-(\frac{\partial^2}{\partial t^2}\,\eta_{u,\,u}(x,\,t)),\,(\frac{\partial^2}{\partial t^2}\,\xi_t(x,\,t)) + 4\,(\frac{\partial}{\partial t}\,\eta_{u,\,u}(x,\,t)),\,\frac{\partial^2}{\partial t^2}\,\xi_x(x,\,t)],\,[x,\,t],$$
$$[\xi_x(x,\,t),\,\xi_t(x,\,t),\,\eta_{u,\,u}(x,\,t)]]$$

```
> HilbertSeries(t);
```

$$3 + 3\,t$$

Here we get the polynomial solutions up to degree 2 (for the Lie algebra):

```
> sol:=PolySol(Jle,2);
```

$$sol := [\xi_x(x,\,t) = C1_{0,\,0} + \frac{1}{2}\,C2_{0,\,1}\,x + C1_{0,\,1}\,t - 2\,C3_{0,\,1}\,x\,t,$$
$$\xi_t(x,\,t) = C2_{0,\,0} + C2_{0,\,1}\,t - 2\,C3_{0,\,1}\,t^2,$$
$$\eta_{u,\,u}(x,\,t) = C3_{0,\,0} + \frac{1}{2}\,C1_{0,\,1}\,x + C3_{0,\,1}\,t - \frac{1}{2}\,C3_{0,\,1}\,x^2]$$

```
> Cons := [C1[0,0],C1[0,1],C3[0,1], C2[0,0],C2[0,1],C3[0,0]];
```

$$Cons := [C1_{0,\,0},\,C1_{0,\,1},\,C3_{0,\,1},\,C2_{0,\,0},\,C2_{0,\,1},\,C3_{0,\,0}]$$

We use a "jets"-command to write down the infinitesimal symmetries explicitly:

```
> lvec:=genvec(sol,Cons,le[4]);
```

$$lvec := [[[1,\,[x]]],\,[[t,\,[x]],\,[\frac{1}{2}\,u\,x,\,[u]]],\,[[-2\,x\,t,\,[x]],\,[-2\,t^2,\,[t]],\,[u\,t - \frac{1}{2}\,u\,x^2,\,[u]]],$$
$$[[1,\,[t]]],\,[[\frac{1}{2}\,x,\,[x]],\,[t,\,[t]]],\,[[u,\,[u]]]]$$

We could have used "jets" to analyse the isomorphism type of the Lie algebra, which we skip. What is important for us is the fact that the solutions of the heat equation form a module for this Lie algebra in the following sense: if f(x,t) is a solution of the heat equation, then taking the commutator of a vector field in the Lie algebra with the vector field [f(x,t),[u]] yields a vector field of the form [g(x,t),[u]] where g(x,t) also is a solution of the heat equation. (This observation carries over to all linear pde-systems.) We start with f(x.t)=1 and take the commutators with the basis of the Lie algebra:

```
> l:=[[1, [u]]];
```

$$l := [[1,\,[u]]]$$

```
> map(s->ldvec(s,l,ivar,dvar),lvec);
```

$$[0,\ [[-\frac{1}{2}\,x,\ [u]]],\ [[-t+\frac{1}{2}\,x^2,\ [u]]],\ 0,\ 0,\ [[-1,\ [u]]]]$$

So the second and the third basis vector of the Lie algebra seem to do something useful:

```
> for i from 1 to 4 do
> ldvec(l,lvec[2],ivar,dvar);l:=ldvec(l,lvec[3],ivar,dvar) od;
```

$$[[\frac{1}{2}\,x,\ [u]]]$$

$$l := [[t-\frac{1}{2}\,x^2,\ [u]]]$$

$$[[\frac{3}{2}\,x\,t-\frac{1}{4}\,x^3,\ [u]]]$$

$$l := [[3\,t^2-3\,x^2\,t+\frac{1}{4}\,x^4,\ [u]]]$$

$$[[\frac{15}{2}\,x\,t^2-\frac{5}{2}\,x^3\,t+\frac{1}{8}\,x^5,\ [u]]]$$

$$l := [[15\,t^3-\frac{45}{2}\,x^2\,t^2+\frac{15}{4}\,x^4\,t-\frac{1}{8}\,x^6,\ [u]]]$$

$$[[\frac{105}{2}\,x\,t^3-\frac{105}{4}\,x^3\,t^2+\frac{21}{8}\,x^5\,t-\frac{1}{16}\,x^7,\ [u]]]$$

$$l := [[105\,t^4-210\,x^2\,t^3+\frac{105}{2}\,x^4\,t^2-\frac{7}{2}\,x^6\,t+\frac{1}{16}\,x^8,\ [u]]]$$

It seems that we get always two linearly independent polynomial solutions of lower degree i for i >0. This can be checked by applying the Janet algorithm to the heat equation itself (or by hand of course):

```
> J:=JanetBasis(ind2eqn(eq,ivar,dvar),ivar,dvar):
> HilbertSeries(t);
```

$$1+2\,t+2\,\frac{t^2}{1-t}$$

We leave it to the reader to prove that the above commutation routine generates all polynomial solutions out of the constant solution.

The next example demonstrates differential elimination.

Example 4.2 *Differential elimination is to be demonstrated. Specification of independent and dependent variables:*

```
> ivar:=[x,y,z];dvar:=[u,v];
```

$$ivar := [x,\ y,\ z]$$

$$dvar := [u,\ v]$$

The pde system:

```
> L:=Ind2Diff([u[z,z]-y*v[x,x], u[y,y]+z*v[y,z]],ivar,dvar);
```

$$L := [(\frac{\partial^2}{\partial z^2}\,\mathrm{u}(x,\ y,\ z))-y\,(\frac{\partial^2}{\partial x^2}\,\mathrm{v}(x,\ y,\ z)),\ (\frac{\partial^2}{\partial y^2}\,\mathrm{u}(x,\ y,\ z))+z\,(\frac{\partial^2}{\partial z\,\partial y}\,\mathrm{v}(x,\ y,\ z))]$$

We want to find an equation for v alone. Computation of the Janet basis with the degree reverse lexicographic ordering with position over term :

```
> J:=JanetBasis(L,ivar,dvar,2);
```

$$J := [[(\frac{\partial^2}{\partial z^2}\,\mathrm{u}(x,\,y,\,z)) - y\,(\frac{\partial^2}{\partial x^2}\,\mathrm{v}(x,\,y,\,z)),\ (\frac{\partial^2}{\partial y^2}\,\mathrm{u}(x,\,y,\,z)) + z\,(\frac{\partial^2}{\partial z\,\partial y}\,\mathrm{v}(x,\,y,\,z)),$$
$$(\frac{\partial^3}{\partial z^2\,\partial y}\,\mathrm{u}(x,\,y,\,z)) - (\frac{\partial^2}{\partial x^2}\,\mathrm{v}(x,\,y,\,z)) - y\,(\frac{\partial^3}{\partial y\,\partial x^2}\,\mathrm{v}(x,\,y,\,z)),\ -2\,(\frac{\partial^3}{\partial z^2\,\partial y}\,\mathrm{v}(x,\,y,\,z))$$
$$-\,2\,(\frac{\partial^3}{\partial y\,\partial x^2}\,\mathrm{v}(x,\,y,\,z)) - y\,(\frac{\partial^4}{\partial y^2\,\partial x^2}\,\mathrm{v}(x,\,y,\,z)) - z\,(\frac{\partial^4}{\partial z^3\,\partial y}\,\mathrm{v}(x,\,y,\,z))],\ [x,\,y,\,z],\ [u,\,v]]$$

So the last equation in the Janet basis is an equation just for v. We want to check whether every solution for this equation can be complemented by a function u such that one gets a solution of the original equations. Therefore the parametric derivatives (still in the same ordering) are computed:

```
> ParamDeriv(ivar,dvar);
```

$$[\frac{z}{1-x} + \frac{1}{1-x} + \frac{y\,z}{1-x} + \frac{y}{1-x},$$
$$\frac{x^2\,y}{(1-x)\,(1-z)} + \frac{x^2}{(1-x)\,(1-z)} + \frac{x}{(1-y)\,(1-z)} + \frac{1}{(1-y)\,(1-z)}]$$

Now the Janet basis and the parametric derivatives for the equation for v are computed, also in the degree reverse lexicographical ordering.

```
> J:=JanetBasis([J[1][4]],ivar,[v]);
```

$$J := [[-2\,(\frac{\partial^3}{\partial z^2\,\partial y}\,\mathrm{v}(x,\,y,\,z)) - 2\,(\frac{\partial^3}{\partial y\,\partial x^2}\,\mathrm{v}(x,\,y,\,z)) - y\,(\frac{\partial^4}{\partial y^2\,\partial x^2}\,\mathrm{v}(x,\,y,\,z))$$
$$-\,z\,(\frac{\partial^4}{\partial z^3\,\partial y}\,\mathrm{v}(x,\,y,\,z))],\ [x,\,y,\,z],\ [v]]$$

```
> ParamDeriv(ivar,[v]);
```

$$\frac{x^2\,y}{(1-x)\,(1-z)} + \frac{x^2}{(1-x)\,(1-z)} + \frac{x}{(1-y)\,(1-z)} + \frac{1}{(1-y)\,(1-z)}$$

Comparison with the second component of the corresponding result for the original equations above shows that any holomorphic solution v of the last equation comes up in a solution of the original equations.

We leave it as an exercise to interchange the role of u and v, i. e. to eliminate u and to compare the corresponding generalized Hilbert series. To get an idea on the coupling between u and v it is helpful to compare the Hilbert Series for the original equations and for the seperated system having the two equations, one for u and one for v. (Of course one has to take the same ordering for both cases.)

There is also a different kind of differential elimination, which tries to find a pde system involving differentiation to fewer variable. For this one has to use the rather expensive lexicographical ordering. The final example demonstrates the command Autonom, which is tantamount to Torsion. In the module language it finds the torsion submodule, gives a presentation of it, and the HILBERT series with repsect to the degrevlex ordering. In the pde-language it finds the functions killed by some linear differential operator. Autonomous elements are relevant in control theory, cf. [Pom 01] and [Qua 01], or [Zer 00] for the constant coeficient case.

Example 4.3 *The pde-system:*

```
> ivar := [s,t]; dvar := [u,v,w];
```

$$ivar := [s, t]$$

$$dvar := [u, v, w]$$

```
> R := Ind2Diff([u[s]+v[t]-w, u+v[s,t]+w], ivar, dvar);
```

$$R := [(\frac{\partial}{\partial s}\,\mathrm{u}(s,\,t)) + (\frac{\partial}{\partial t}\,\mathrm{v}(s,\,t)) - \mathrm{w}(s,\,t),\ \mathrm{u}(s,\,t) + (\frac{\partial^2}{\partial t\,\partial s}\,\mathrm{v}(s,\,t)) + \mathrm{w}(s,\,t)]$$

```
> Torsion(R, ivar, dvar);
```

$$[[\_\mathrm{T1}(s,\,t) = \mathrm{u}(s,\,t) + (\frac{\partial}{\partial t}\,\mathrm{v}(s,\,t)),\ \_\mathrm{T2}(s,\,t) = -\mathrm{u}(s,\,t) - (\frac{\partial}{\partial t}\,\mathrm{v}(s,\,t))],$$
$$[-\_\mathrm{T1}(s,\,t) - \_\mathrm{T2}(s,\,t),\ \_\mathrm{T1}(s,\,t) + (\frac{\partial}{\partial s}\,\_\mathrm{T1}(s,\,t))],\ 1 + \frac{s}{1-s}]$$

5 Acknowledgements

The contribution of two authors (Yu.A.B. and V.P.G.) was partially supported by the grants 00-15-96691 and 01-01-00708 from the Russian Foundation for Basis Research. We thank M. Barakat and G. Hartjen to let us take over various functions from their system "jets".

References

[AGW 03] S. Abenda, G. Gaeta, S. Walcher (eds.), Symmetry and Pertubation Theory SPT2002. to appear World Scientific 2003.

[Bar 01] M. Barakat, Jets. A MAPLE-Package for Formal Differential Geometry. 1-12 in [GMV 01].

[BaH 02] M. Barakat, G. Hartjen Jets. A MAPLE-Package for Variational and Jet Calculus. submitted.

[BCGPR 02] Y. A. Blinkov, C. F. Cid, V. P. Gerdt, W. Plesken, D. Robertz, The Maple Package "Janet": I. Polynomial Systems. Preprint.

[BGe 01] Y. A. Blinkov, V. P. Gerdt, D. A. Yanovich, Construction of Janet bases II. Polynomial Bases. 249-263 in [GMV 01].

[GMV 01] V. G. Ganzha, E. W. Mayr, E. V. Vorozhtsov (eds.) Computer Algebra in Scientific Computing CASC 2001. Springer Berlin etc. 2001.

[GeB 98a] V. P. Gerdt, Y. A. Blinkov, Involutive bases of polynomial ideals. Mathem. and Computers in Simulation 45 (1998), 519-541.

[GeB 98b] V. P. Gerdt, Y. A. Blinkov, Minimal involutive bases. Mathem. and Computers in Simulation 45 (1998), 543-560.

[Ge 99] V. P. Gerdt, Completion of Linear Differential Systems to Involution, In: Computer Algebra in Scientific Computing / CASC'99, V.G.Ganzha, E.W.Mayr, E.V.Vorozhtsov (eds.), Springer-Verlag, Berlin (1999) 115-137.

[Ge 02] V. P. Gerdt, Involutive Division Technique: Some Generalizations and Optimizations. Journal of Mathematical Sciences 108(6), 2002, 1034-1051.

[Jan 29] M. Janet, Lecons sur les systèmes des équationes aux dérivées partielles. Cahiers Scientifique IV, Gauthiers-Villars, Paris 1929.

[Ple 03] W. Plesken, JANETs Algorithm. to appear in [AGW 03].

[Pom 94] J.- F. Pommaret, Partial Differential Equations and Group Theory. Kluver Academic Publishers 1994.

[Pom 01] J.- F. Pommaret, Partial Differential Control Theory. Vol. I: Mathematical Tools, Vol II: Control Systems. Kluver Academic Publishers 2001.

[Qua 01] A. Quadrat, Analyse algébrique des systèmes de contrôle linéaires multidimensionnels. Thèse de Doctorat, Ecole nationale des ponts et Chaussees, Paris 1999, cf. http://www-sop.inria.fr/cafe/Alban.Quadrat/

[VuC 00] Khai T. Vu, John Carminati, DESOLV for Maple V Release 5, School of Computing and Mathematics, Deakin University, Geelong, Victoria, Australia, released June 2000

[Zer 00] E. Zerz, Topics in Multidimensional Linear System Theory. Lecture Notes in Control and Inform. Sciences 256, Springer 2000.