Appendix A

Cartan-Kähler Theory

We will give a brief introduction into the theory of exterior differential systems. After a short listing of the basic notions, we will discuss Pfaffian systems and we will outline the construction of regular chains of integral elements. This leads again to the Cartan-Kähler Theorem. Finally we will discuss the relation to the formal theory

A.1 Exterior Systems

We will introduce in this section briefly the basic notions of the calculus of exterior differential forms. As in the following sections we will not give details and only sketch some proofs. For more we refer to the literature, e.g. [17, 26, 27, 42, 64, 84, 93, 109].

Let M be an N-dimensional differentiable manifold, T^*M its cotangent bundle. Over T^*M we can construct the bundle

$$\Lambda T^*M = \bigoplus_{p=0}^{N} \Lambda^p T^*M \,. \tag{A.1}$$

Sections of $\Lambda^p T^*M \to M$ are called *p-forms* $\Lambda^0 T^*M$ contains functions defined on M; $\Lambda^1 T^*M$ is defined as T^*M. To introduce forms of higher degree, we need the *wedge product*. It is associative and distributive but not commutative

$$\omega \wedge \eta = (-1)^{pq} \eta \wedge \omega \,, \qquad \omega \in \Lambda^p T^*M \,,\ \eta \in \Lambda^q T^*M \,. \tag{A.2}$$

Because of the skew-symmetry there are no forms of degree higher than N. This product endows ΛT^*M with the structure of a Grassmann algebra. We will always assume that we deal with homogeneous forms, i.e. each form is element of one $\Lambda^p T^*M$. Since T^*M is the dual of TM, we can consider p-forms as multi-linear, skew-symmetric maps from $(TM)^p$ to $\mathbb{R}$.

The *exterior derivative* maps a p-form into a $(p+1)$-form and satisfies ($\omega \in \Lambda^p T^*M$)

$$\begin{aligned} \mathrm{d}(\omega + \eta) &= \mathrm{d}\omega + \mathrm{d}\eta \,, \\ \mathrm{d}(\omega \wedge \eta) &= \mathrm{d}\omega \wedge \eta + (-1)^p \omega \wedge \mathrm{d}\eta \,, \\ \mathrm{d}^2\omega &= 0 \,, \end{aligned} \tag{A.3}$$

In a local coordinate system $\{z^1, \dots, z^N\}$ on M, we define for a 0-form f

$$\mathrm{d}f = \sum_{i=1}^{N} \frac{\partial f}{\partial z^i} \, \mathrm{d}z^i \, . \tag{A.4}$$

This suffices together with the other properties of the exterior derivative to calculate $\mathrm{d}\omega$ for forms ω of arbitrary degree.

Two further operations which we will need are the *interior derivative* and the *Lie derivative*. Let v be a vector field on M. The interior derivative i_v maps a p-form into a $(p-1)$-form and is defined by:

$$\begin{aligned} i_v \omega &= \omega(v) \, , \qquad \omega \in \Lambda^1 T^* M \, , \\ i_v(\omega \wedge \eta) &= (i_v \omega) \wedge \eta + (-1)^p \omega \wedge (i_v \eta) \, , \qquad \omega \in \Lambda^p T^* M \, . \end{aligned} \tag{A.5}$$

Applied to a 0-form the interior derivative yields 0.

The Lie derivative $\mathcal{L}_v$ can be seen as an intrinsic version of a directional derivative. One way to define it is to take the limit of difference quotients along the flow generated by v. For forms one can introduce it simpler via

$$\mathcal{L}_v = i_v \mathrm{d} + \mathrm{d} i_v \, . \tag{A.6}$$

Thus the Lie derivative of a p-form is again a p-form. But it can also be applied to any tensor field over M. Especially, if w is another vector field, then

$$\mathcal{L}_v w = -\mathcal{L}_w v = [v, w] \tag{A.7}$$

where $[v, w]$ denotes the usual commutator of vector fields. We also note the following useful formulae

$$i_v \mathrm{d}f = \mathcal{L}_v f = v(f) \, , \tag{A.8}$$

$$i_{[v,w]} = [\mathcal{L}_v, i_w] \, , \tag{A.9}$$

$$[\mathcal{L}_v, \mathcal{L}_w] = \mathcal{L}_{[v,w]} \, . \tag{A.10}$$

An *exterior system* Ω consists of a finite set of equations

$$\omega_k = 0 \, , \qquad \omega_k \in \Lambda T^* M \, . \tag{A.11}$$

We can always assume without loss of generality that Ω does not contain 0-forms. Otherwise, we take as base manifold the submanifold defined by them. Restricting all forms in Ω to this manifold yields an equivalent exterior system without 0-forms. *Integral manifolds* are those submanifolds $\mathcal{S} \subset M$ that

$$\omega_k\big|_{\mathcal{S}} = 0 \, , \qquad \forall \omega_k \in \Omega \, . \tag{A.12}$$

An *integral element* $E_{z_0} \subset T_{z_0} M$ is a subset of the tangent space at a point $z_0 \in M$ which is annihilated by the forms in Ω, i.e. $i_{v_{j_1}} \cdots i_{v_{j_k}} \omega = 0$ for every k-form $\omega \in \Omega$ and for all $v_{j_\ell} \in E_{z_0}$. Integral elements can be considered as "linearized solutions": if $\mathcal{S}$ is a k-dimensional integral manifold, then $T_{z_0}\mathcal{S}$ is a k-dimensional integral element for all $z_0 \in \mathcal{S}$. Note, however, that not every integral element is necessarily the tangent space of an integral manifold.

Really important is not the set Ω itself but the ideal generated by it. Furthermore, since $(\mathrm{d}\omega)|_{\mathcal{S}} = \mathrm{d}(\omega|_{\mathcal{S}})$, the integral manifolds do not change, if we add to Ω all forms $\mathrm{d}\omega_k$. Thus we introduce

$$I(\Omega) = \left\{\omega \in \Lambda T^*M \mid \omega = \sum_{k=1}^{|\Omega|}(\alpha_k \wedge \omega_k + \beta_k \wedge \mathrm{d}\omega_k)\right\}, \tag{A.13}$$

the *closed* ideal generated by Ω, i.e. $\mathrm{d}I(\Omega) \subset I(\Omega)$. α_k, β_k are here arbitrary forms.

Many objects in the geometric theory of differential equations can be very easily understood using exterior systems. We start with a brief introduction of Cauchy characteristic vector fields, then we discuss symmetries.

Definition A.1 *A vector field v is called* Cauchy characteristic *for the exterior system Ω, if*

$$i_v I(\Omega) \subset I(\Omega). \tag{A.14}$$

Proposition A.2 *The Cauchy characteristic vector fields form a Lie algebra.*

Proof. This proposition is an immediate consequence of (A.9). We have to show that if v, w are Cauchy characteristic, then $[v, w]$ is it, too. For this purpose, we note that (A.6) implies that for a Cauchy characteristic vector field $\mathcal{L}_v I(\Omega) \subset I(\Omega)$. Thus it follows from (A.9) that $i_{[v,w]}I(\Omega) \subset I(\Omega)$. •

Theorem A.3 *Let the Lie algebra of the Cauchy characteristic vector fields of the exterior system Ω be r-dimensional. Then there exists a coordinate system $(x^1, \dots, x^r, y^1, \dots, y^{n-r})$ such that one can find a basis of $I(\Omega)$ independent of the x^k.*

Proof. We choose a coordinate system such that the algebra of the Cauchy characteristic vector fields are spanned by $\frac{\partial}{\partial x^1}, \dots, \frac{\partial}{\partial x^r}$. We will show how one can construct a basis of $I(\Omega)$ independent of one x^k. By iteration one obtains then a basis independent of all x^k. To abbreviate the notation we write i_k for the interior derivative in direction $\frac{\partial}{\partial x^k}$. A form ω is independent of x^k if and only if $i_k\omega = 0$ (i.e. ω does not contain $\mathrm{d}x^k$) and $i_k\mathrm{d}\omega = 0$ (i.e. no coefficient depends on x^k).

We will use an induction on the degree of the forms in Ω assuming as always that Ω does not contain 0-forms. Since the one-forms are handled exactly in the same way as the forms of higher degree, we detail only the step from forms of degree $d-1$ to degree d. Thus we assume that we have already found a basis for all forms in $I(\Omega)$ with degree less than d independent of x^k.

Let a basis for the forms of degree d in $I(\Omega)$ be ω_j for $j = 1, \dots, p$. Since $\frac{\partial}{\partial x^k}$ is a Cauchy characteristic vector field, we have $i_k\omega_j = \rho_j$ where the $\rho_j \in I(\Omega)$ are forms of degree $d-1$. Thus by assumption $i_k\rho_j = 0$. Consider now the forms

$$\omega_j' = \omega_j - \mathrm{d}x^k \wedge \rho_j. \tag{A.15}$$

They generate the same ideal as the ω_j and satisfy $i_k\omega_j' = 0$, i.e. they do not contain $\mathrm{d}x^k$.

Now we must consider the exterior derivatives of these forms. They can be written

$$\mathrm{d}\omega_j' = \sum_\ell a_j^\ell \omega_\ell' + \sum_i \tau_i \wedge \alpha_j^i \tag{A.16}$$

with $\tau_i \in I(\Omega)$ and $\deg \tau_i < d$. Let b_i^ℓ be the entries of an invertible matrix satisfying

$$\frac{\partial b_i^\ell}{\partial x^k} = \sum_j b_i^j a_j^\ell \tag{A.17}$$

and let β_j^i be forms such that $\partial \beta_j^i / \partial x^k = \alpha_j^i$ (the differentiation is to be understood componentwise). Then we can introduce the d-forms

$$\omega_j'' = \sum_\ell b_j^\ell \omega_\ell' - \sum_i \tau_i \wedge \beta_j^i \tag{A.18}$$

which satisfy $i_k \omega_j'' = i_k \mathrm{d}\omega_j'' = 0$ and also generate all d-forms in $I(\Omega)$. •

Definition A.4 *A vector field v generates a (local)* symmetry *of the exterior system Ω, if*

$$\mathcal{L}_v I(\Omega) \subset I(\Omega)\,. \tag{A.19}$$

v is then called an isovector *of Ω.*

This definition forms the base of the exterior version of Lie's symmetry theory [66] as studied in Chapter 5 (There exists a computer algebra package for symmetry analysis based on this exterior system approach [23]). We have seen above that any Cauchy characteristic vector field is automatically an isovector. The converse does not hold however. The basic idea of symmetry reduction is to impose an isovector as Cauchy characteristic of an augmented ideal.

Proposition A.5 *If v is an isovector of the exterior system Ω, then it is Cauchy characteristic for the augmented system $\Omega \cup i_v\Omega$.*

Proof. Let us define $I_v = I(\Omega \cup i_v\Omega)$ and let $\Omega = \{\omega_k\}$. We have defined $I(\Omega)$ to be the closure of the ideal generated by Ω, thus I_v consists of forms spanned by $\omega_k, \mathrm{d}\omega_k, i_v\omega_k$ and $\mathrm{d}i_v\omega_k$. Since $i_v i_v \omega$ vanishes trivially due to the skew-symmetry of forms, there remains only to show that the forms $i_v \mathrm{d}\omega_k$ and $i_v \mathrm{d} i_v \omega_k$ are contained in I_v. Now using (A.6) $i_v \mathrm{d}\omega_k = \mathcal{L}_v \omega_k - \mathrm{d} i_v \omega_k$. By assumption v is an isovector, hence every term on the right hand side is in I_v. Similarly we obtain $i_v \mathrm{d} i_v \omega_k = i_v(\mathcal{L}_v - i_v \mathrm{d})\omega_k = i_v \mathcal{L}_v \omega_k \in I_v$. •

Now we can invoke Theorem A.3 to find a basis of the augmented system with differentials of one variable less. This reduced system corresponds to the reduced system obtained in Chapter 5 using invariants. The additional generators $i_v\Omega$ can be seen as exterior analogue of the invariant surface condition (5.6). Indeed every integral manifold of $\Omega \cup i_v\Omega$ is also an integral manifold of Ω and furthermore invariant under the transformations generated by v. It is also easy to understand the non-classical method of Bluman and Cole in this context. Looking carefully at the proof of Proposition A.5, we note that we do not have to require that v is an isovector of Ω but that it suffices, if v is an isovector of the augmented system $\Omega \cup i_v\Omega$.

A.2 Pfaffian Systems

An important special case of exterior systems are Pfaffian systems. They can be considered as duals to Lie algebras of vector fields. We will also use them to describe integral manifolds.

Definition A.6 *A* Pfaffian system $\mathcal{P}$ *on an* N*-dimensional manifold* M *is an exterior system containing only one-forms. The number of linearly independent forms in* $\mathcal{P}$ *is called the* rank *of* $\mathcal{P}$*. We will always assume that* $\mathcal{P}$ *is* regular, *i.e. that its rank is constant on* M*. A Pfaffian system* $\mathcal{P}$ *of rank* r *is* completely integrable, *if it has an* $(N-r)$*-dimensional integral manifold.*

If $\mathcal{P} = \{\omega_1, \dots, \omega_r\}$, then it generates algebraically (i.e. without taking exterior derivatives) the ideal

$$I_{\mathcal{P}} = \{\alpha \in \Lambda T^*M \mid \alpha \wedge \omega_1 \wedge \ldots \wedge \omega_r = 0\}. \tag{A.20}$$

This implies that a Pfaffian system is closed, if and only if $\mathrm{d}\omega_k \wedge \omega_1 \wedge \ldots \wedge \omega_r = 0$ for $k = 1, \dots, r$, as then $I_{\mathcal{P}} = I(\mathcal{P})$. With every Pfaffian system we can also associate its annihilator

$$\mathcal{P}^{\perp} = \{v \in TM \mid \omega(v) = 0\,,\ \forall \omega \in I_{\mathcal{P}}\}. \tag{A.21}$$

Theorem A.7 (Frobenius) *Let* $\mathcal{P}$ *be a Pfaffian system of rank* r*. Then the following statements are equivalent:*

(i) $\mathcal{P}$ *is closed.*

(ii) $\mathcal{P}^{\perp}$ *is a* $(N-r)$*-dimensional Lie algebra.*

(iii) $\mathcal{P}$ *is completely integrable.*

(iv) Locally there exist coordinates $z^1, \dots, z^N$ *on* M*, such that* $\mathcal{P}$ *is equivalent to the system* $\mathrm{d}z^1 = \cdots = \mathrm{d}z^r = 0$.

PROOF. The equivalence of (i) and (ii) follows straight-forward. Since $\omega(v)$ is just another way to write $i_v\omega$, we have for any two vectors $v, w \in \mathcal{P}^{\perp}$ using (A.6,A.9)

$$i_{[v,w]}\omega = [\mathcal{L}_v, i_w]\omega = i_w\mathcal{L}_v\omega = i_w i_v \mathrm{d}\omega\,. \tag{A.22}$$

If now $\mathcal{P}$ is closed, then the expression at the right end is obviously zero and we can conclude that $[v,w] \in \mathcal{P}^{\perp}$. If however $\mathcal{P}^{\perp}$ is a Lie algebra, then the expression at the left end is zero and we can conclude that $\mathrm{d}\omega$ vanishes on any two-dimensional integral element of $\mathcal{P}$ and is thus contained in $I_{\mathcal{P}}$. Since TM is N-dimensional and elements of $\mathcal{P}^{\perp}$ have to satisfy r linearly independent conditions, $\dim \mathcal{P}^{\perp} = N - r$.

The equivalence of (iii) and (iv) is also trivial. If $\mathcal{S}$ is an integral manifold of dimension $N - r$, choose coordinates in which it is locally described by the equations $z^{r+1} = \cdots = z^N = 0$. Conversely, this represents an $(N-r)$-dimensional integral manifold of $\mathrm{d}z^1 = \cdots = \mathrm{d}z^r = 0$.

(ii) is an obvious consequence of (iv), since $\frac{\partial}{\partial z^{r+1}}, \dots, \frac{\partial}{\partial z^N}$ yields a basis of $\mathcal{P}^{\perp}$. The hardest part of the proof is to show the converse. This is the classical Frobenius Theorem as it can be found in nearly every textbook on differential geometry. Let us assume that $Y_{r+1}, \dots, Y_N$ form a basis of $\mathcal{P}^{\perp}$. Then there exists locally a coordinate system $y^1, \dots, y^N$ on M such that $\frac{\partial}{\partial y^1}, \dots, \frac{\partial}{\partial y^r}, Y_{r+1}, \dots, Y_N$ yields a basis of TM. In these coordinates we have

$$Y_i = \sum_{j=1}^{r} a_i^j \frac{\partial}{\partial y^j} + \sum_{j=r+1}^{N} b_i^j \frac{\partial}{\partial y^j}\,, \tag{A.23}$$

where the matrix b_i^j is invertible. Hence we can introduce the vector fields

$$Z_j = \sum_{i=r+1}^{N} (b^{-1})_j^i Y_i = \frac{\partial}{\partial y^j} + \sum_{i=1}^{r} \sum_{k=r+1}^{N} (b^{-1})_j^k a_k^i \frac{\partial}{\partial y^i}\,, \quad j = r+1, \dots, N\,, \tag{A.24}$$

which also form a basis of $\mathcal{P}^\perp$. Since by assumption $\mathcal{P}^\perp$ is a Lie algebra, the commutator $[Z_i, Z_j]$ must be a linear combination of the Z_k. But because of the special form (A.24) of these vector fields, this can only be the case, if

$$[Z_i, Z_j] = 0\,, \tag{A.25}$$

as the commutator does not contain any component in direction of $\frac{\partial}{\partial y^j}$ for $j = r+1, \dots, N$. But now by a well-known theorem of differential geometry (cf. [160]) there exist coordinates $z^1, \dots, z^N$ such that $Z_j = \frac{\partial}{\partial z^j}$. In these coordinates, the annihilator of $\mathcal{P}^\perp$ has obviously the simple basis $dz^1, \dots, dz^r$ as claimed. •

We can describe locally a submanifold by giving a basis of either its tangent space or its co-tangent space. A basis of the co-tangent space can of course be considered as a Pfaffian system. Furthermore it must be a closed one by the preceding theorem: since we start with a submanifold, the basis of the tangent space is a Lie algebra, hence its annihilator is closed. We call this now a *Pfaffian representation* of the submanifold. In Section 2.6 we used the term co-normal variety.

A.3 Cartan-Kähler Theorem

The main goal of this section is to describe an algorithm to construct regular chains of integral elements and to obtain this way a Pfaffian representation of integral manifolds. This will yield an exterior version of Theorem 4.9. Again at intermediate steps we will have to apply the Cauchy-Kowalevsky Theorem; thus we restrict again to analytic system. For simplicity we further assume that the exterior system Ω does not contain 0-forms.

As in the proof of the Cartan-Kähler Theorem in Section 4.3 we will build the solution step by step. The main idea is to extend k-dimensional integral elements to $(k+1)$-dimensional ones and to prove the existence of an integral manifold through them using the Cauchy-Kowalevsky Theorem. To achieve this goal we will have to restrict to a "generic" situation.

Let M be an N-dimensional manifold with coordinates z^i. We assume, that we are given a k-dimensional integral manifold $\mathcal{S}_k$. Its tangent space $T_{z_0}\mathcal{S}_k$ at a point $z_0 \in \mathcal{S}_k$ is a k-dimensional integral element E_k spanned by some vectors $Z_1, \dots, Z_k$. We are looking now for a further vector $Z \in T_{z_0}M$ such that $Z_1, \dots, Z_k, Z$ generates a $(k+1)$-dimensional integral element E_{k+1}. Obviously the components of Z have to satisfy the following linear system of equations

$$\begin{aligned} i_Z\omega^{(1)} &= 0\,, \\ i_Z i_{Z_j}\omega^{(2)} &= 0\,, \qquad\qquad j = 1, \dots, k \\ &\vdots \\ i_Z i_{Z_1}\cdots i_{Z_k}\omega^{(k+1)} &= 0 \end{aligned} \tag{A.26}$$

for all j-forms $\omega^{(j)} \in \Omega$. (A.26) is usually called the *polar system* of E_k and its solution space is denoted by $H(E_k)$ (as for the symbol, we will also denote the polar system itself often $H(E_k)$).

Let the rank of (A.26) be r_k. In general, r_k will depend on z_0 and E_k. We call E_k a *regular integral element*, if r_k is constant in a neighborhood of (z_0, E_k). Obviously choosing $Z = Z_j$ yields a solution. Hence we need $\dim H(E_k) = N - r_k > k$ to get a solution linearly independent of the Z_j. Adding such a solution Z to E_k gives a $(k+1)$-dimensional integral element.

If E_k is again regular, we can repeat the process. This way one can compute a chain $\{E_k\}$ of integral elements for k from 0 to some value n. If all E_k but the last one are regular, then one speaks of a *regular chain.* E_n is then called an *ordinary integral element.* The maximal dimension of such an element is the *Cartan genus* g of the exterior system Ω. A set of vector fields $\{Z_i\}$, $i = 1, \dots, g$ such that E_k is generated by $Z_1, \dots, Z_k$ is called a *regular sequence.* We will only consider regular chains, this is the above mentioned "generic" situation. Little is known about singular solutions whose tangent space is not an ordinary integral element.

Theorem A.8 (Cartan-Kähler) *Let $\mathcal{S}_k$ be an analytic k-dimensional integral manifold of the analytic exterior system Ω and $z_0 \in \mathcal{S}_k$. Assume $\{E_i\}_{i=1,\dots,k+1}$ is a regular chain of integral elements with $E_k = T_{z_0}\mathcal{S}_k$. Then there exists a $(k+1)$-dimensional analytic integral manifold $\mathcal{S}_{k+1}$ parameterized by $N-(k+1)-r_k$ functions of $k+1$ variables such that $\mathcal{S}_k \subset \mathcal{S}_{k+1}$.*

Proof. We choose a coordinate system $(x^1, \dots, x^{k+1}, z^1, \dots, z^{N-k-1})$ such that $\mathcal{S}_k$ is given by the equations $x^{k+1} = 0$, $z^j = \phi^j(x^1, \dots, x^k)$ and such that there exists a regular sequence of the form

$$Z_i = \frac{\partial}{\partial x^i} + \sum_j a_i^j \frac{\partial}{\partial z^j}\,. \tag{A.27}$$

We will construct an integral manifold $\mathcal{S}_{k+1}$ of the form $z^j = \Phi^j(x^1, \dots, x^{k+1})$. Thus we make the following ansatz to solve the polar system $H(E_k)$

$$Z = \frac{\partial}{\partial x^{k+1}} + \sum_j a_{k+1}^j \frac{\partial}{\partial z^j}\,. \tag{A.28}$$

Since by assumption $H(E_k)$ has rank r_k, the polar system can be put into the form

$$a_{k+1}^l = f^l(a_i^j, a_{k+1}^m)\,, \qquad \begin{cases} i &= 1,\dots,N-k-1\,, \\ l &= 1,\dots,r_k\,, \\ m &= r_k+1,\dots,N-k-1\,. \end{cases} \tag{A.29}$$

We can assume that in a suitable chosen neighborhood the functions f^l are analytic. Now we must prove that there exists an integral manifold such that $Z_1, \dots, Z_k, Z$ span its tangent space, i.e. we want to substitute

$$a_i^j = \frac{\partial \Phi^j}{\partial x^i}\,. \tag{A.30}$$

This turns (A.29) into a system of partial differential equations. The condition $\mathcal{S}_k \subset \mathcal{S}_{k+1}$ yields the initial values

$$\Phi^j(x^1, \dots, x^k, 0) = \phi^j(x^1, \dots, x^k)\,. \tag{A.31}$$

If we set $\Phi^j = F^j$ for $j > r_k$ where $F^j(x^1, \dots, x^{k+1})$ are some arbitrary analytic functions satisfying the initial conditions (A.31), then this system of partial differential equations becomes normal and we can apply the Cauchy-Kowalevsky Theorem. It yields a unique solution. Thus we have obtained a manifold $\mathcal{S}_{k+1}$ parameterized by $N-(k+1)-r_k$ functions of $k+1$ variables as claimed.

We must, however, still prove that this manifold is indeed an integral manifold. By construction all forms of degree $k+1$ in $I(\Omega)$ vanish when restricted to $\mathcal{S}_{k+1}$. But of the form of lower degree, we know only that they vanish on $\mathcal{S}_k$. The idea is to foliate $\mathcal{S}_{k+1}$ by introducing $\mathcal{S}_k^{(t)} = \mathcal{S}_{k+1} \cap \{x^{k+1} = t\}$. Thus $\mathcal{S}_k^{(0)} = \mathcal{S}_k$. Since we are only interested in local solutions, we must show that the relations

$$i_{Z_{\ell_1}} \cdots i_{Z_{\ell_j}} \omega^{(j)} = 0\,, \quad j = 1, \dots, k\,, \tag{A.32}$$

which hold on $\mathcal{S}_k$, remain valid for small values of t.

To this end we note that it follows from (A.27,A.28) that $[Z_i, Z_j] = [Z_i, Z] = 0$. Thus we can write

$$\begin{aligned} \mathcal{L}_Z i_{Z_{\ell_1}} \cdots i_{Z_{\ell_j}} \omega^{(j)} &= i_Z \mathrm{d} i_{Z_{\ell_1}} \cdots i_{Z_{\ell_j}} \omega^{(j)} + \mathrm{d} i_Z i_{Z_{\ell_1}} \cdots i_{Z_{\ell_j}} \omega^{(j)} \\ &= i_Z \mathrm{d} i_{Z_{\ell_1}} \cdots i_{Z_{\ell_j}} \omega^{(j)} \\ &= i_Z i_{Z_{\ell_1}} \mathrm{d} i_{Z_{\ell_2}} \cdots i_{Z_{\ell_j}} \omega^{(j)} + \mathcal{L}_{Z_1} i_Z i_{Z_{\ell_2}} \cdots i_{Z_{\ell_j}} \omega^{(j)} \\ &= \ldots = i_Z i_{Z_{\ell_1}} \cdots i_{Z_{\ell_j}} \mathrm{d}\omega^{(j)} = 0\,. \end{aligned} \tag{A.33}$$

The first equality is just the definition of the Lie derivative. The second one uses that Z is a solution of the polar system. The third uses that the vector fields commute. Within $\mathcal{S}_k$ nothing happens thus $\mathcal{L}_{Z_{\ell_j}}$ maintains the relations of the polar system. For the last equality we use that $I(\Omega)$ is closed. •

In order to compute the number of arbitrary functions in the general solution of the exterior system Ω we must estimate the number of parameters a regular sequence depends on. If $Z_1, \dots, Z_k$ are given and the polar system $H(E_k)$ has rank r_k, then Z_{k+1} depends on $N - r_k$ parameter. Thus the whole sequence contains $Ng - r_0 - \cdots - r_{g-1}$ parameters.

In the proof of the Cartan-Kähler Theorem these parameters become arbitrary functions. But we must still determine their dependencies. A priori they are arbitrary functions defined on the manifold M. But the vector fields contained in the constructed regular sequence must span the tangent space of integral manifolds, hence by the Frobenius Theorem they must form a Lie algebra. This yields conditions on the functions in form of a first-order system of partial differential equations.

In the special coordinate system we have used in our proof of the Cartan-Kähler Theorem these conditions are very easy: the arbitrary functions appearing in step k depend on $x^1, \dots, x^k$. In general they are more complicated. Instead of evaluating the commutators one can also construct a Pfaffian representation by computing the annihilator of the vector fields in the sequence. Then we must check whether the resulting Pfaffian system is closed.

A.4 Relations to Formal Theory

The purpose of this section is to compare the notions of formal theory and of Cartan-Kähler Theory. Guillemin and Sternberg [63] gave together with Serre a rigorous proof of the equivalence of both approaches to involution. We will not go into this but rather show based on an example how their notions are related. Thus we assume now that the exterior system Ω stems from a partial differential equation.

In the jet bundle formalism we distinguish from the very beginning independent and dependent variables. In the exterior calculus there is nothing like this. We are searching for submanifolds of a manifold which has no natural fibered structure. It must be imposed externally by choosing a set of so-called *independence forms*. These are linearly independent one-forms $\chi_1, \ldots, \chi_n$ and we admit integral manifolds $\mathcal{S}$ only, if they satisfy the *transversality condition*

$$(\chi_1 \wedge \cdots \wedge \chi_n)\big|_{\mathcal{S}} \neq 0\,. \tag{A.34}$$

The choice of the independence forms induces a coordinate system z^i on M such that $\chi_i = \mathrm{d}z^i$ for $i = 1, \ldots, n$. An integral manifold $\mathcal{S}$ satisfying the transversality condition (A.34) is projectable into a submanifold given by $z^i = const$ for $i > n$. Thus we can consider $z^1, \ldots, z^n$ as "independent variables", for $\mathcal{S}$ can be described by equations of the form $z^i = f^i(z^1, \ldots, z^n)$ for $i = n+1, \ldots, N$. If we start with a partial differential equation, we will choose as independence forms a basis of T^*X. Then only transverse integral manifolds respect the fibration of $\mathcal{E}$ and correspond to solutions of the partial differential equation.

We say that an integral element E is in *involution* with respect to the independence forms χ_i, if there is no $v \in E$ such that

$$\chi_i(v) = 0\,, \qquad i = 1, \ldots, n,. \tag{A.35}$$

We call the exterior system Ω involutive, if it is possible to construct an n-dimensional, ordinary, involutive integral element E_n. Using the method outlined in the previous section we can ensure involution, if E_n is spanned by a regular sequence of the form

$$Z_k = \frac{\partial}{\partial z^k} + \sum_{i=n+1}^{N} a^i_k \frac{\partial}{\partial z^i}\,, \quad k = 1, \ldots, n \tag{A.36}$$

(assuming again that $\chi_i = \mathrm{d}z^i$).

But it might be that it is not possible to construct an n-dimensional integral element with this ansatz, because the rank of one polar system becomes too big. Then we must generalize (A.36) to

$$Z_k = \sum_{i=1}^{n} A^i_k \frac{\partial}{\partial z^i} + \sum_{i=n+1}^{N} a^i_k \frac{\partial}{\partial z^i}\,, \tag{A.37}$$

where the coefficients A^i_k must be chosen in such a way that we obtain an n-dimensional, involutive integral element. But this corresponds exactly to the linear changes of coordinates discussed in Section 2.7. Thus if we have chosen a non-systatic basis of T^*X as independence forms, we will not need this generalization.

But of course even with the generalized ansatz (A.37) it might happen that it is not possible to construct an n-dimensional integral element with the desired properties. Then Ω is not in involution. There are three possible reasons for this: (i) the system is inconsistent (at least with respect to the chosen independence forms), (ii) integrability conditions occur, i.e. the exterior system will be restricted to a submanifold of M by adding some 0-forms, (iii) conditions must be imposed on some of the occurring arbitrary functions, i.e. the system must be prolonged.

The number of arbitrary parameters in a regular sequence must now be adapted to the additional transversality condition. All vectors in it must be of the form (A.37). The coefficients A^i_k are, however, no longer relevant, since they correspond only to changes of the independent coordinates. Thus we must subtract in each step n parameters from our previous result. A regular sequence depends now only on $g(N-n) - r_0 - \cdots - r_{g-1}$ parameters.

We can use this result to relate the ranks of the polar systems to the Cartan characters. For simplicity we assume that we are given a first-order equation $\mathcal{R}_1$ and that we transform it into an exterior system by taking the pull-back of the *contact system* of $J_1\mathcal{E}$, i.e. the exterior system generated by the forms $\mathrm{d}u^\alpha - \sum_i p_i^\alpha \mathrm{d}x^i$. It encodes the interpretation of the jet variables as derivatives. The dimension N is now the fiber dimension of $\dim J_1\mathcal{E}$. From Corollary 3.10 we obtain

$$\begin{aligned} \alpha_1^{(g)} &= \dim J_1\mathcal{E} - r_{g-1}\,, \\ \alpha_1^{(k)} - \alpha_1^{(k+1)} &= \dim J_1\mathcal{E} - r_{k-1}\,, \end{aligned} \tag{A.38}$$

where g is the Cartan genus of the equations.

One can find an exterior analogue to our criterion (2.19) for an involutive symbol. This requires the completely algebraic introduction of the tableau[1] of an exterior system [17]. Then one defines involution of the tableau similar to our Definition 2.8 of an involutive symbol. This approach stems from Matsushima [104], but seems to be used only for linear systems.

Example A.9 We consider the following over-determined first-order equation for two dependent variables u, v and two independent variables x, y

$$\mathcal{R}_1 \;:\; \begin{cases} u_x &= f(x,y,u,v,v_x)\,, \\ u_y &= g(x,y,u,v,v_x)\,, \\ v_y &= h(x,y,u,v,v_x)\,. \end{cases} \tag{A.39}$$

Obviously the system is already in the solved form discussed in Chapter 4, if we choose y as x^2.

Before we rewrite $\mathcal{R}_1$ as an exterior system, we analyze the conditions on the right hand sides f, g, h for the equation to be in involution. $\mathcal{M}_1$ is involutive, if and only if v_{xx} does not occur in $\mathcal{R}_2$. This requires

$$\frac{\partial f}{\partial v_x}\frac{\partial h}{\partial v_x} = \frac{\partial g}{\partial v_x}\,. \tag{A.40}$$

There arises correspondingly one integrability condition

$$\frac{\partial f}{\partial y} - \frac{\partial g}{\partial x} + \frac{\partial f}{\partial u}g - \frac{\partial g}{\partial u}f + \frac{\partial f}{\partial v}h - \frac{\partial g}{\partial v}v_x + \frac{\partial f}{\partial v_x}\Big(\frac{\partial h}{\partial x} + \frac{\partial h}{\partial u}f + \frac{\partial h}{\partial v}v_x\Big) = 0\,. \tag{A.41}$$

We will assume in the sequel that these conditions hold. Concluding the formal analysis we note that in this case the Cartan characters are

$$\alpha_1^{(2)} = 0\,, \quad \alpha_1^{(1)} = 1\,. \tag{A.42}$$

We transform this partial differential equation into an exterior system by taking the pull-back on $\mathcal{R}_1$ of the contact system. This yields the following two forms

$$\begin{aligned} \omega_1 &= \mathrm{d}u - f\mathrm{d}x - g\mathrm{d}y\,, \\ \omega_2 &= \mathrm{d}v - v_x\mathrm{d}x - h\mathrm{d}y\,. \end{aligned} \tag{A.43}$$

defined on a 5-dimensional manifold with coordinates x, y, u, v, v_x. To close the system we need their exterior derivatives

$$\begin{aligned} \mathrm{d}\omega_1 &= \Big(\frac{\partial g}{\partial x} - \frac{\partial f}{\partial y} + \frac{\partial g}{\partial u}f - \frac{\partial f}{\partial u}g + \frac{\partial g}{\partial v}v_x - \frac{\partial f}{\partial v}h\Big)\,\mathrm{d}x\wedge\mathrm{d}y + \frac{\partial f}{\partial v_x}\,\mathrm{d}v_x\wedge\mathrm{d}x + \frac{\partial g}{\partial v_x}\,\mathrm{d}v_x\wedge\mathrm{d}y\,, \\ \mathrm{d}\omega_2 &= \Big(\frac{\partial h}{\partial x} + \frac{\partial h}{\partial u}f + \frac{\partial h}{\partial v}v_x\Big)\,\mathrm{d}x\wedge\mathrm{d}y + \mathrm{d}v_x\wedge\mathrm{d}x + \frac{\partial h}{\partial v_x}\,\mathrm{d}v_x\wedge\mathrm{d}y\,. \end{aligned} \tag{A.44}$$

[1]Not to be confused with the tableaux introduced in Chapter 2!

(A.40,A.41) imply that involution of $\mathcal{R}_1$ is equivalent to these two forms being linearly dependent. Using the ansatz

$$Z_1 = \frac{\partial}{\partial x} + a_1\frac{\partial}{\partial u} + b_1\frac{\partial}{\partial v} + c_1\frac{\partial}{\partial v_x}\,, \quad Z_2 = \frac{\partial}{\partial y} + a_2\frac{\partial}{\partial u} + b_2\frac{\partial}{\partial v} + c_2\frac{\partial}{\partial v_x} \tag{A.45}$$

one obtains immediately from the polar systems that

$$r_0 = 2\,, \quad r_1 = 3 \tag{A.46}$$

in perfect agreement with (A.38) and that a regular sequence is given by

$$\begin{aligned} Z_1 &= \frac{\partial}{\partial x} + f\frac{\partial}{\partial u} + v_x\frac{\partial}{\partial v} + c_1\frac{\partial}{\partial v_x}\,, \\ Z_2 &= \frac{\partial}{\partial y} + g\frac{\partial}{\partial u} + h\frac{\partial}{\partial v} + \Big(\frac{\partial h}{\partial x} + f\frac{\partial h}{\partial u} + v_x\frac{\partial h}{\partial v} + c_1\frac{\partial h}{\partial v_x}\Big)\frac{\partial}{\partial v_x}\,. \end{aligned} \tag{A.47}$$

If $\mathcal{R}_1$ is not involutive, i.e. the two forms in (A.44) are linearly independent, we obtain one more equation for the polar system $H(E_1)$ and it is no longer consistent for arbitrary values of c_1. Thus it is not possible to construct a regular sequence of length 2.

From the sequence (A.47) one obtains the following Pfaffian representation of the integral manifolds

$$\begin{aligned} \pi_1 &= \mathrm{d}v_x - c_1\mathrm{d}x - \Big(\frac{\partial h}{\partial x} + f\frac{\partial h}{\partial u} + v_x\frac{\partial h}{\partial v} + c_1\frac{\partial h}{\partial v_x}\Big)\mathrm{d}y\,, \\ \pi_2 &= \mathrm{d}u - f\mathrm{d}x - g\mathrm{d}y\,, \\ \pi_3 &= \mathrm{d}v - v_x\mathrm{d}x - h\mathrm{d}y\,. \end{aligned} \tag{A.48}$$

The requirement that these three forms generate a closed ideal leads to the following complicated differential equation for the so far arbitrary function c_1

$$\begin{aligned} &\left(\frac{\partial h}{\partial x} + f\frac{\partial h}{\partial u} + v_x\frac{\partial h}{\partial v}\right)\frac{\partial c_1}{\partial v_x} + \left(h - v_x\frac{\partial h}{\partial v_x}\right)\frac{\partial c_1}{\partial v} + \left(g - f\frac{\partial h}{\partial v_x}\right)\frac{\partial c_1}{\partial u} + \\ &\quad + \frac{\partial c_1}{\partial y} - \frac{\partial h}{\partial v_x}\frac{\partial c_1}{\partial x} - \frac{\partial^2 h}{\partial v_x^2}c_1^2 - \\ &\quad - \left(\frac{\partial h}{\partial v} + 2v_x\frac{\partial^2 h}{\partial v\partial v_x} + 2f\frac{\partial^2 h}{\partial u\partial v_x} + \frac{\partial f}{\partial v_x}\frac{\partial h}{\partial u} + 2\frac{\partial^2 h}{\partial x\partial v_x}\right)c_1 - \\ &\quad - f^2\frac{\partial^2 h}{\partial u^2} - v_x^2\frac{\partial^2 h}{\partial v^2} - 2v_x f\frac{\partial^2 h}{\partial u\partial v} - 2f\frac{\partial^2 h}{\partial x\partial u} - \frac{\partial^2 h}{\partial x^2} - 2v_x\frac{\partial^2 h}{\partial x\partial v} - \\ &\quad - f\frac{\partial f}{\partial u}\frac{\partial h}{\partial u} - v_x\frac{\partial f}{\partial v}\frac{\partial h}{\partial u} - \frac{\partial f}{\partial x}\frac{\partial h}{\partial u} = 0\,. \end{aligned} \tag{A.49}$$

We encounter now one of the main problems of the Cartan-Kähler Theory. This Frobenius condition on c_1 depends on the solution, as we must know u, v, v_x in order to evaluate it. Only then we obtain the “real” equation for c_1 determining the form of its arguments. In the case of simple hyperbolic equations like the Wave Equation this approach leads to the equations for the characteristics [67].

Appendix B

Janet-Riquier Theory

We will outline the main ideas of the Janet-Riquier approach to differential equations and discuss Riquier's Existence Theorem. Special emphasis will be put on the relation between involutive and passive systems.

B.1 Passive Systems

Most of the concepts used in the Janet-Riquier Theory have already been introduced in the main text, as our presentation of the formal theory was guided by this more differential-algebraic approach to differential equations. We will only outline some the central ideas. For details we refer to the literature [80, 124, 127, 153].

It is easy to generate integrability conditions; the problem is to construct them in a systematic way such that one knows when one has found all non-trivial ones. Apparently the first one to formulate an algorithm for this task and to prove its termination was Tresse [156]. Later the theory was mainly developed by Riquier and Janet introducing the concept of passivity. Intuitively passive means that a system does no longer "actively" generate integrability conditions.

Passivity is defined with respect to a *ranking* (cf. Definition 2.11). It does not make sense to call a system passive without referring to a specific one, as the same system may be passive with respect to one ranking but not with respect to another one. There exists some theory about methods to classify all possible rankings which we will, however, ignore.

As soon as a ranking has been chosen, we can define the *leading derivative* of an equation as the highest occurring derivative. *Principal derivatives* are now all leading derivatives of a system and their derivatives. Recalling the basic properties of rankings especially the monotonicity (property (ii) in Definition 2.11) we see that they occur as leading derivatives of prolonged equations.

We call a system S of differential equations *orthonomic*, if (i) all its equations are solved with respect to their leading derivatives; (ii) all leading derivatives are different and (iii) no principal derivatives are occurring on the right hand side, i.e. they have all been eliminated using the equations of S. In the sequel we will always assume that we are dealing with such a system. Since we consider only smooth equations, this represents no restriction because of the Implicit Function Theorem.

Multiplicative variables are slightly differently defined compared with formal theory[1].

[1] Usually this and the following definitions are given for sets of monomials and not for sets of deriva-

Let p^α_μ be a leading derivative of a given orthonomic system S. x^j is called multiplicative for it, if there does not exist another leading derivative p^α_ν such that $\nu_k = \mu_k$ for $k > j$ and $\nu_j > \mu_j$. Let $\mathcal{L}^\alpha_>$ denote the set of all leading derivatives of S with respect to the ranking $>$ of the form p^α_μ, i.e. belonging to the same dependent variable u^α. We introduce furthermore the set $\bar{\mathcal{L}}^\alpha_>$ containing all derivatives p^α_μ which can be obtained by differentiating other derivatives either in $\bar{\mathcal{L}}^\alpha_>$ or $\mathcal{L}^\alpha_>$ with respect to their multiplicative variables. We call $\mathcal{L}^\alpha_>$ a *complete set*, if every derivative of its elements belongs to $\bar{\mathcal{L}}^\alpha_>$.

Definition B.1 *An orthonomic system S is* passive *with respect to the ranking $>$, if all sets $\mathcal{L}^\alpha_>$ are complete and no integrability conditions occur in the prolongation of S.*

This definition resembles Definition 2.8 of an involutive equation. The completeness condition is very similar to our criterion (2.19) for an involutive symbol. The absence of integrability conditions can be checked analogously to Proposition 2.9: prolonging an equation with respect to a non-multiplicative variable must yield an equation which reduces to zero, if all principal derivatives in it are eliminated using S.

It is not difficult to design an algorithm completing a given system S for a given ranking $>$ to a passive system. One possibility is outlined in Fig. B.1. The basic idea is similar to the Buchberger algorithm for the construction of Gröbner bases. Indeed both algorithms can be seen as special instances of the general Knuth-Bendix critical pair completion [18]. A critical pair consists now of two equations whose leading derivatives belong to the same dependent variable, as such equations may generate an integrability condition.

Algorithm `Passive` $(S, >)$

Input: System S, ranking $>$
Output: Orthonomic and passive system S'

[1] Solve every equation in S for its leading derivative
[2] Eliminate all principal derivatives on the right hand side
[3] $S' \leftarrow S$
[4] `until` S' passive `repeat`
[4.1] Compute integrability conditions
[4.2] Reduce integrability conditions modulo S'
[4.3] Solve non-vanishing conditions for their leading derivatives
[4.4] Add resulting equations to S'
[5] `return` S'

Figure B.1: Algorithm for the completion to a passive system.

The algorithms used in the different implementations mentioned in Section 8.1 differ mainly in the steps [4.1] and [4.2]. It is well-known from the theory of Gröbner bases that choosing a good strategy for selecting the next critical pair is decisive for the performance. Reid [124] has thoroughly analyzed this basic algorithm and found a way to minimize the number of integrability conditions which must be checked using an equivalence class approach.

tives, for the theory was motivated by Hilbert's theory of polynomial ideals generated by monomials (cf. Section 2.5). We will nevertheless use directly the derivatives instead of rewriting them using (2.23).

It is tempting to call a passive system Differential Gröbner Basis and indeed there are some authors using this term. Note, however, that we require that we deal throughout with orthonomic systems, i.e. that we can solve every equation for its leading derivative. Even for equations with purely polynomial non-linearity this is a very optimistic thought. As soon as one tries to remove of this restriction, one gets a lot of trouble. All known solutions of this problem make concessions either in the class of equations treated or in the properties of the resulting basis.

Given a passive system it is easy to compute a general formal power series solution. One proceeds completely analogously to Section 2.5. Recall the main point about the construction of such solutions is the absence of integrability conditions. But this is guaranteed by the passivity of the system. As already mentioned above the principal derivatives are selected according to the ranking: they consist of all leading derivatives and their derivatives. A partial answer to the question of convergence of the series is given by the following

Theorem B.2 (Riquier) *Let the system S be passive with respect to an orderly ranking. Then the corresponding formal power series solution converges.*

This existence theorem due to Riquier [127] (see also [151, 152]) explains the importance of orderly rankings which we have already seen in Example 4.3. It is in some sense even more general than Theorem 4.9, for we can obtain different initial value problems using different rankings. Its proof goes along similar lines as the one of the Cauchy-Kowalevsky Theorem majorising the power series solution by convergent series. The necessary estimates are, however, considerably more involved.

B.2 Relations to Formal Theory

There is no need for an extensive discussion of the relations between the Janet-Riquier Theory and the formal approach. It is important to note that passivity is *not* equivalent to involution, although one can sometimes find such statements in the literature. Passivity corresponds to formal integrability. The correspondence can be made explicit for class-respecting rankings.

Proposition B.3 *A local description of a formally integrable differential equation $\mathcal{R}_q$ will form a passive system with respect to any class-respecting ranking, if the coordinate system is δ-regular and if the system is in solved form.*

PROOF. We note that perhaps present equations of order less than q are not important here, as formal integrability implies that *all* their prolongations up to order q are contained in the system. Thus we can restrict our attention to the equations of highest order. Let us first assume that $\mathcal{R}_q$ is not only formally integrable but in addition involutive. Since we consider only class-respecting rankings, we can further assume that the pivots of the symbol have been chosen according to the ranking with respect to which we want to prove passivity.

Let p^α_μ be one of the pivots and let k be its class. Then from the point of formal theory we associate with it the multiplicative variables $x^1, \dots, x^k$. We claim that from the point of view of Janet-Riquier Theory we obtain exactly the same multiplicative variables. Since all pivots are of order q, this is trivial for x^i with $i < k$. Any other pivot p^α_ν with $\nu_j = \mu_j$ for $j > i$ would have to be identical with p^α_μ. To show that x^k is multiplicative, we note that if $\nu_j = \mu_j$ for $j > k$ and $|\nu| = |\mu|$, then $\nu_k \leq \mu_k$. If $\nu \neq \mu$, then ν must have a non-vanishing entry below k and thus be of lower class. In any case, it is not possible that $\nu_k > \mu_k$. Hence x^k is multiplicative.

Since we assume that our coordinate system is δ-regular, we can invoke Proposition 2.9. It implies that all sets $\mathcal{L}^\alpha_>$ are complete and that no integrability conditions occur because of the formal integrability. In principle, we can thus conclude that our system is passive. This is, however, not really true, because we have required in our definition of passive that the system is orthonomic. First this is stricter than the solved form we used in the formal theory. Furthermore, if lower order equations are present, this requires to eliminate their prolongations.

We ignore this point here, for it does not really affect the meaning of passive but only additional "normal form" properties. One can define passive independently of orthonomic. Once a passive system is obtained, it can be made orthonomic by a simple process of solving for leading derivatives and eliminating them in other equations.

Finally, we must consider the case of a non-involutive equation. Here basically the same remark holds. Any formally integrable system becomes involutive after some prolongations. During these no integrability conditions can occur. A subsequent reduction to obtain an orthonomic system eliminates these prolonged equations. Thus without loss of generality we can apply the above proof. •

Like in the case of the Cartan-Kähler theory the easiest way to understand the differences and similarities of the two approaches is to consider a concrete example.

Example B.4 The following simple equation with one dependent and three independent variables occurs in connection with the (2+1)-dimensional Krichever-Novikov Equation

$$\mathcal{R}_2 \, : \, \begin{cases} u_{xy} - u_t = 0 \,, \\ u_{xx} - u_y = 0 \,. \end{cases} \tag{B.1}$$

In [140] it was shown that this equation becomes involutive after one projection leading to the addition of the integrability condition

$$u_{yy} - u_{xt} = 0 \,. \tag{B.2}$$

A formal analysis reveals that the general solution of $\mathcal{R}_2^{(1)}$ contains three arbitrary functions of one argument.

A δ-regular coordinate system is given by $x^1 = t$, $x^2 = y$, $x^3 = x$. One can easily check that (B.1) together with (B.2) is passive with respect to the inverse lexicographic ranking induced by this ordering. If we choose, however, the inverse lexicographic ranking induced by the ordering $t > y > x$ the system is not passive but we must add a further integrability condition

$$u_{yyy} - u_{tt} = 0 \,. \tag{B.3}$$

Note especially that this equation is of third-order, although our involutive equation was of second-order.

In order to explain this effect Fig. B.2 shows the Reid diagram for the second ranking. The arrows indicate the prolongations with respect to the multiplicative variables and thus bound the domain of principal derivatives. One sees at once that the $(y-t)$-plane contains none. Thus if the system was passive in the form (B.1,B.2), then its general solution would depend on a function of two arguments in contradiction to the result of the formal analysis. This discrepancy is removed, as soon as the integrability condition (B.3) is added to the system. Then the initial value problem generated by Reid's algorithm [124] requires three functions of one argument.

This example shows furthermore that δ-regularity makes its hidden appearance even in this algebraic approach to differential equations. We conjecture that if a system is

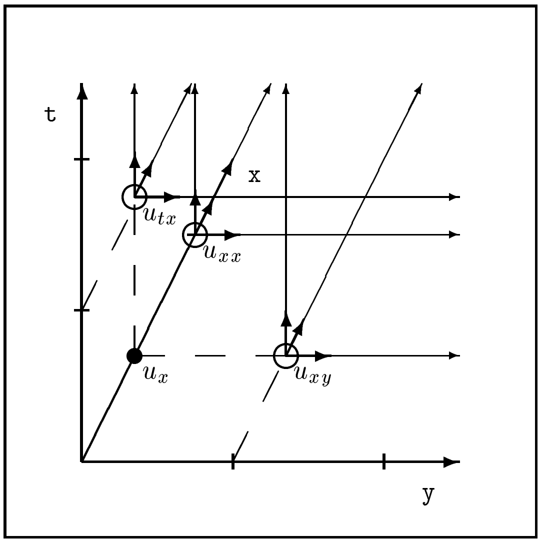

Figure B.2: Reid diagram for equations (B.1,B.2).

passive with respect to a ranking that corresponds to a coordinate system that is not δ-regular, then it contains more equations than a formally integrable system and often these equations will even be of higher order like (B.3). The reason is evident from Fig. B.2. Since the coordinate system is not δ-regular, we do not obtain sufficiently many principal derivatives after a prolongation. Thus we must add equations.

Finally one should remark that the most striking difference between Janet-Riquier Theory and formal theory is the coordinate invariance of the latter one. An involutive equation in the sense of Definition 2.8 represents an intrinsic geometric object. Similarly are the Cartan characters or the Hilbert polynomial intrinsically defined. They are independent of the chosen coordinate system. In contrast the definition of a ranking requires to single out a specific coordinate system, for there does not exist a "natural" ranking after a coordinate transformation.

Appendix C

Miscellaneous

The first section in this appendix contains the definition and some important properties of the modified Stirling numbers used in Chapter 3. The second section shows an intrinsic way to rewrite a higher order equation as a first order one.

C.1 Modified Stirling Numbers

Definition C.1 *The* modified Stirling numbers $s_k^{(n)}(q)$ *are defined for all non-negative integers n, k, q with $n \geq k$ by*

$$s_k^{(n)}(q) = \begin{cases} 1 & \text{for } k = 0\,, \\ \sigma_k^{(n)}(q+1, q+2, \dots, q+n) & \text{for } 0 < k \leq n\,, \end{cases} \tag{C.1}$$

where $\sigma_k^{(n)}$ denotes the elementary symmetric polynomial of degree k in n unknowns.

These numbers arise in the following expansion of binomial coefficients as polynomials which was used in Chapter 3 to obtain an explicit expression for the Hilbert polynomial

$$\binom{q+r+n}{q+r} = \frac{1}{n!} \sum_{i=0}^{n} s_{n-i}^{(n)}(q)\, r^i\,. \tag{C.2}$$

In [138] they were called *symmetric q-products*. But their close relationship to the Stirling numbers of the first kind $S_n^{(i)}$ [1] has lead to the new name. The latter ones can be defined through their generating function by

$$\frac{x!}{n!} = \sum_{i=0}^{n} S_n^{(i)}\, x^i\,. \tag{C.3}$$

With $x = r+n$ we obtain an identity similar to (C.2); note, however, the different position of the scripts

$$\binom{r+n}{r} = \frac{1}{r!} \sum_{i=0}^{n} S_n^{(i)}\,(r+n)^i\,. \tag{C.4}$$

From the elementary symmetric polynomials the modified Stirling numbers inherit the recursion relation

$$s_k^{(n)}(q) = s_k^{(n-1)}(q) + (q+n)\, s_{k-1}^{(n-1)}(q)\,. \tag{C.5}$$

One can use this relation to construct explicit expressions for $s_k^{(n)}(q)$. They become, however, for larger k fairly lengthy.

Lemma C.2

$$s_n^{(n)}(q) = (q+n)!/q!\,, \tag{C.6}$$

$$s_1^{(n)}(q) = \tfrac{1}{2}n(n+2q+1)\,, \tag{C.7}$$

$$s_2^{(n)}(q) = \tfrac{1}{24}n(n-1)\Big(3n^2+12qn+5n+12q^2+12q+2\Big)\,. \tag{C.8}$$

PROOF. (C.6) and (C.7) follow immediately from Definition C.1. For (C.8) we apply (C.5)

$$\begin{aligned} s_2^{(n)}(q) &= s_2^{(n-1)}(q) + (q+n)\, s_1^{(n-1)}(q) \\ &= \ldots = \sum_{l=2}^{n-1}(q+l+1)\, s_1^{(l)}(q) + s_2^{(2)}(q) \\ &= \frac{1}{2}\sum_{l=1}^{n-1}(q+l+1)\Big[l^2+(2q+1)l\Big]\,. \end{aligned}$$

Evaluation of this sum yields the claim. •

One can also deduce recursion formulae relating the modified Stirling numbers for different values of q or express $s_k^{(n)}(q)$ directly through the $s_l^{(m)}(0)$ with $m = 0,\ldots,n$. Like most properties of these numbers it stems from a well-known formula for binomial coefficients.

Lemma C.3

$$s_{n-k+1}^{(n)}(q) = n\, s_{n-k}^{(n-1)}(q) + s_{n-k+1}^{(n)}(q-1)\,. \tag{C.9}$$

PROOF. The base for this lemma is the following identity coming directly from the Pascal triangle

$$\binom{q+r+n}{q+r} = \sum_{k=0}^{n}\binom{q+r+k-1}{q+r-1}\,. \tag{C.10}$$

Plugging in (C.2) yields after a few simple index manipulations yet another recursion relation for the modified Stirling numbers

$$s_k^{(n)}(q) = \sum_{j=0}^{k}\frac{n!}{(n-j)!}s_{k-j}^{(n-j)}(q-1)\,. \tag{C.11}$$

Applying this relation in (C.9) proves at once the lemma. •

In order to express $s_k^{(n)}(q)$ through the modified Stirling numbers with $q=0$, we must generalize (C.11). Each binomial $\binom{r+k}{k}$ represents by (C.2) a polynomial of degree k. Thus every polynomial of degree n can be written as a linear combination of $\binom{r}{0}, \binom{r+1}{1}, \ldots \binom{r+n}{n}$. Especially there exist numbers $d_k^{(n)}(q)$ such that

$$\binom{q+r+n}{q+r} = \sum_{k=0}^{n} d_k^{(n)}(q)\binom{r+k}{k}\,. \tag{C.12}$$

Applying repeatedly (C.10) one can express these coefficients as nested sums

$$d_k^{(n)}(q) = \sum_{k_1=k}^{n} \sum_{k_2=k_1}^{n} \cdots \sum_{k_{q-1}=k_{q-2}}^{n} 1\,. \tag{C.13}$$

For lower values of q these sums can easily be evaluated. We omit, however, explicit expressions. Now we obtain similarly as above

$$s_k^{(n)}(q) = \sum_{j=0}^{k} \frac{n!}{(n-j)!} d_k^{(n)}(q)\, s_{k-j}^{(n-j)}(0)\,. \tag{C.14}$$

C.2 Reduction to First Order

It is sometimes of advantage to rewrite a differential equation of higher order as a first-order equation. In this section we will demonstrate two intrinsic ways to do this. The basic idea is that one can identify $J_{q+1}\mathcal{E}$ with a submanifold of $J_1(J_q\mathcal{E})$. For concreteness we will use coordinates. For $J_{q+1}\mathcal{E}$ we take our usual coordinate system $(x^i \mid u^\alpha \mid p^\alpha_\mu)$. For $J_1(J_q\mathcal{E})$ we use $(x^i \mid v^\alpha_\mu \mid q^\alpha_{\mu,i})$ with $0 \le |\mu| \le q$. The v's are the new dependent variables; the q's denote their first order derivatives.

We can now identify $J_{q+1}\mathcal{E}$ with the image of the immersion

$$\iota_{q+1} : \left\{ \begin{array}{ccc} J_{q+1}\mathcal{E} & \longrightarrow & J_1(J_q\mathcal{E}) \\ (x^i \mid u^\alpha \mid p^\alpha_\nu) & \longmapsto & (x^i \mid p^\alpha_\mu \mid p^\alpha_{\mu+1_i})\,, \qquad |\nu| \le q+1,\ 0 \le |\mu| \le q\,. \end{array} \right. \tag{C.15}$$

Locally, this submanifold is described by the equations

$$\begin{aligned} q^\alpha_{\mu,i} &= v^\alpha_{\mu+1_i}\,, & |\mu| &< q\,, \\ q^\alpha_{\mu+1_i,j} &= q^\alpha_{\mu+1_j,i}\,, & |\mu| &= q-1\,. \end{aligned} \tag{C.16}$$

The mapping ι_{q+1} can be defined intrinsically as the prolongation of the identity map on $J_q\mathcal{E}$, i.e. as the unique mapping satisfying $\iota_{q+1} \circ j_{q+1} = j_1 \circ j_q$. We can represent this condition by a commutative diagram:

$$\begin{tikzcd} J_{q+1}\mathcal{E} \arrow[r, "\iota_{q+1}"] & J_1(J_q\mathcal{E}) \\ \mathcal{E} \arrow[u, "j_{q+1}"] \arrow[r, "j_q"] & J_q\mathcal{E} \arrow[u, "j_1"'] \end{tikzcd} \tag{C.17}$$

The classical way to rewrite a differential equation $\mathcal{R}_q$ as an equivalent first-order equation $\tilde{\mathcal{R}}_1$ is to take $\tilde{\mathcal{R}}_1 = \iota_q(\mathcal{R}_q)$. Thus one introduces for every derivative of order less than q a new dependent variable. If $\mathcal{R}_q$ is locally given by $\Phi^\tau(x^i, u^\alpha, p^\alpha_\mu) = 0$, we get for the reduced equation

$$\tilde{\mathcal{R}}_1 : \left\{ \begin{array}{ll} \tilde{\Phi}^\tau(x^i, v^\alpha_\mu, q^\alpha_{\mu,i}) = 0\,, & \\ q^\alpha_{\mu,i} = v^\alpha_{\mu+1_i}\,, & |\mu| < q\,, \\ q^\alpha_{\mu+1_i,j} = q^\alpha_{\mu+1_j,i}\,, & |\mu| = q-1\,. \end{array} \right. \tag{C.18}$$

$\tilde{\Phi}^\tau$ is not uniquely defined, as there are in general several possibilities to express a derivative p^α_μ by one of the new coordinates. One way which allows one later to compute easily the characters is given by

$$p^\alpha_\mu \longmapsto \begin{cases} v^\alpha_\mu & \text{for } |\mu| \leq q-1\,, \\ q^\alpha_{\mu-1_k,k} & \text{for } |\mu| = q,\ \operatorname{class} p^\alpha_\mu = k\,. \end{cases} \tag{C.19}$$

Proposition C.4 *$\tilde{\mathcal{R}}_1$ and $\mathcal{R}_q$ have the same Cartan characters and there exists a one-to-one correspondence between their solutions. $\mathcal{R}_q$ is involutive, if and only if $\tilde{\mathcal{R}}_1$ is involutive.*

Proof. Due to the substitution rule (C.19) the first line of (C.18) yields $\beta^{(k)}_q$ equations of class k, the same value as in $\mathcal{R}_q$. The second line represents $\dim J_{q-2}\mathcal{E}$ equations of class k and a little combinatorial calculation shows that the last line leads to additionally $\dim S_{q-1}T^*X \otimes V\mathcal{E} - \binom{q+n-k-1}{q-1}$ equations of class k. Adding these values we get

$$\tilde{\beta}^{(k)}_1 = \dim J_{q-2}\mathcal{E} + \dim S_{q-1}T^*X \otimes V\mathcal{E} - \alpha^{(k)}_q \tag{C.20}$$

and thus for the Cartan characters

$$\tilde{\alpha}^{(k)}_1 = \dim J_{q-1}\mathcal{E} - \tilde{\beta}^{(k)}_1 = \alpha^{(k)}_q\,. \tag{C.21}$$

We note that if the x^i form a δ-regular coordinate system for $\mathcal{R}_q$, then they do it also for $\tilde{\mathcal{R}}_1$ and vice versa. The additional equations (C.16) are in a form that leads to maximal values for the number of equations of class n and so on. Thus only the equations $\tilde{\Phi}^\tau = 0$ could make trouble. But again due to our substitution rule (C.19) we see that if we get maximal values for $\mathcal{R}_q$ we get them also for $\tilde{\mathcal{R}}_1$ and vice versa.

These considerations imply furthermore that involution of the symbol of one of the systems is equivalent to involution of the symbol of the other one. It is easy to see that (C.16) alone forms an involutive system and that there arise no integrability conditions between it and the equations $\tilde{\Phi}^\tau = 0$. But if the equations $\Phi^\tau = 0$ generate integrability conditions, then the same happens to the equations $\tilde{\Phi}^\tau = 0$ and vice versa.

The relation between the solution spaces of $\mathcal{R}_q$ and $\tilde{\mathcal{R}}_1$, respectively, is obvious. If a section σ is a solution of $\mathcal{R}_q$, then $j_{q-1}(\sigma)$ is a solution of $\tilde{\mathcal{R}}_1$. Conversely, (C.16) ensures that every solution of $\mathcal{R}_1$ is a section of the form $j_{q-1}(\sigma)$. •

Another way to produce a first-order system is to apply this method to $\mathcal{R}_{q+1}$ instead of $\mathcal{R}_q$. Thus we take the equation $\tilde{\mathcal{R}}_1 = \iota_{q+1}(\mathcal{R}_{q+1})$. Using again the substitution rule (C.19) but now for the derivatives of order $q+1$, we are lead to a quasi-linear system. The original equations $\Phi^\tau = 0$ become then even algebraic equations.

For the Cartan characters $\tilde{\alpha}^{(k)}_1$ of the reduced equation, we obtain now from (C.21) and Lemma 3.6

$$\tilde{\alpha}^{(k)}_1 = \alpha^{(k)}_{q+1} = \alpha^{(k)}_q + \cdots + \alpha^{(n)}_q \tag{C.22}$$

and conversely

$$\alpha^{(k)}_q = \tilde{\alpha}^{(k)}_1 - \tilde{\alpha}^{(k+1)}_1 = \tilde{\beta}^{(k+1)}_1 - \tilde{\beta}^{(k)}_1\,. \tag{C.23}$$

Appendix D

Reference Manual

The purpose of this appendix is to provide tables of most procedures currently implemented in our environment for geometric computations with partial differential equations. They also contain brief descriptions of the tasks of the different operations. However, they do not comment on the implementation or the methods used. For this we refer to Chapter 8. The same holds for examples of the usage of the environment.

The descriptions are very similar to the comments given in the source code using `++`. In the next version of the compiler it will be possible to access these comments using the HyperDoc system incorporated in the AXIOM user interface (at least this is announced). Currently this is unfortunately not possible.

The order of the tables corresponds to the hierarchy explained in Section 8.3. We always start with the categories. Tables for domains are only given, if they contain additional operations. Some procedures return objects of a fairly complicated type. These are mostly meant for the use within programs and not for interactive use. Thus these types are not given explicitly. They must be looked up in the source code or with the `)show` command of the interpreter.

In order to make the tables not larger than they already are, we use abbreviations for the types. Those of the newly implemented ones will be introduced in the next paragraphs (cf. also the table in Fig. 8.2); the library types include `Boolean (B)`, `Expression Integer (EI)`, `Integer (I)` with the subtypes `PositiveInteger (PI)` and `NonNegative-Integer (NNI)` and the quotient field (rational numbers) `Fraction Integer (FI)`, `List (L)`, `Matrix (M)`, `Symbol (S)`, `OutputForm (OUT)`, `UnivariatePolynomial (UP)`, and `Vector (V)`. Following the usual convention of AXIOM we denote the domain or category under consideration by `$`.

Jet Bundles: The category `JetBundleCategory` abbreviated by `JBC` was mainly introduced to facilitate the passing of arguments to other domains and categories. Otherwise it would always be necessary to repeat explicitly all parameters like number of independent and dependent variables, respectively, or their names. There exist three instances of this category, namely `JetBundle` abbreviated by `JB`, `IndexedJetBundle (IJB)` and `JetBundleSymAna (JBSA)`.

`JB` takes two lists of symbols as arguments. They represent the names of independent and dependent variables, respectively. For the derivatives repeated index notation is used. `IJB(x,u,p,n,m)` generates a jet bundle with `n` independent and `m` dependent variables. In the output they are denoted by `x` and `u`, respectively. For derivatives `p` is used. Multi-index as well as repeated index notation can be chosen. `JBSA` is essentially identical with `JB`. It only determines its parameters differently: it requires as arguments another jet bundle which provides the new independent variables and symbols for the names of the new dependent variables.

Most operations of JBC can be found in Fig. D.1 on page 172; in addition it is a sub-category of OrderedSet. IJB does not provide additional procedures. JB (and hence also JBSA) implements two additional procedures for the easier input of jet variables, namely a coerce from Symbol and an operation D with two arguments, a Symbol and a List of Symbol, to enter derivatives.

Functions on Jet Bundles: The basic category in this layer is JetBundleFunction-Category abbreviated by JBFC. It specifies the fundamental operations like partial and formal differentiation, computation of Jacobi matrices and simplification. A list of most operations is given in Fig. D.2 on page 173. Additionally there are the operations inherited from PartialDifferentialRing Symbol. For a more convenient input the operations X, U, P, numIndVar, numDepVar, setNotation and getNotation are copied from JBC.

The simplification procedures can be divided into two classes. simplify, simpMod and simpOne use only algebraic operations. One of the main tasks of simplify is to exhibit integrability conditions, if any are present. Otherwise no mixing of equations of different order happens. reduceMod and autoReduce also use differential operations. They correspond to algorithms used in differential algebra.

There are currently three instances of this category implemented, namely the domains JetBundleExpression (JBE), JetBundleLinearFunction (JBLF) and JetLazyFunction (JLF). JetBundleXExpression (JBX) is a kind of specialization of JBE. It belongs to the sub-category BaseFunctionCategory (BFC) for functions only defined on the base space X.

Higher Structures: The core of this layer is the domain DifferentialEquation abbreviated by DE. Its main purpose is to provide an efficient data structure to store intermediate results. In its current form it is especially adapted for computations as they are needed in the completion algorithm 2.3. DE takes two arguments: a jet bundle from JBC and a domain from JBFC to represent the equations. Fig. D.3 on page 174 shows the implemented operations of DE.

The implementation of the differential geometric domains VectorField (abbreviated by VF) and Differential (DIFF) are fairly primitive. They should be considered as a temporary hack, until AXIOM contains a better environment for differential geometric calculations. The arguments for both are like the ones for DE: a jet bundle from JBC and a coefficient domain from JBFC. Most operations of these domains can be found in Fig. D.4 and Fig. D.5, respectively, on page 175. In addition they contain the operations of a Module over the coefficient domain denoted by D.

Applications and Utility Packages: The package CartanKuranishi is abbreviated by CK. It contains basically a procedure to complete a given differential equation to an involutive one. Furthermore there are some procedures for computations with the Cartan characters and the Hilbert polynomial. A full list is shown in Fig. D.6 on page 176. CK takes two arguments: one domain JB of the category JBC and one domain D of the category JBFC(JB). The differential equations are then from the domain DE(JB,D) (abbreviated by DE in Fig. D.6). Note that if lists of characters like $\alpha_q^{(k)}$ or $\beta_q^{(k)}$ are passed as argument, they must always be ordered by increasing k, in contrast to the way they were written in the main part.

The package `SymmetryAnalysis` is abbreviated by `SYMANA`. It is still in an experimental state. Its first argument is a jet bundle; the other two are symbols used for the names of the coefficients in the ansatz for symmetry generators. A preliminary list of its procedures is shown in Fig. D.8 on page 178.

The domain `SparseEchelonMatrix` is abbreviated by `SEM`. It is especially designed for the fast determination of row echelon forms for sparse matrices which are already close to this form. It takes as argument a domain `D` from `Ring` for the entries and a domain `C` from `OrderedSet` whose elements are used to label the columns of the matrices. A complete list of the operations of `SEM` is contained in Fig. D.7 on page 177.

The packages `JetCoordinateTransformation (JCT)` and `LUDecomposition (LUD)` provide only two and three procedures, respectively. `JCT` takes four arguments: two jet bundle `JB1` and `JB2` plus two vectors `Y, W` containing expressions for the old coordinates in terms of the new ones. We will write `E1, E2` for `JBE JB1` and `JBE JB2`, respectively. `LUD` expects a domain of type `Field` as argument. The operations are shown with their signatures in Fig. D.9 and Fig. D.10, respectively, on page 178.

`allRepeated`	`L NNI -> L L PI`	Computes all possible realizations of a given multi-index as repeated index.
`class`	`L NNI -> NNI`	Returns the class of a multi-index.
`class`	`$ -> NNI`	Returns the class of a jet variable.
`coerce`	`$ -> EI`	Coerces a jet variable into an expression.
`coerce`	`$ -> S`	Coerces a jet variable into a symbol.
`derivativeOf?`	`($, $) -> L NNI`	Checks whether the first argument is a derivative of the second one. Returns either the difference of their multi-indices, if positive, or an empty list.
`differentiate`	`($, PI) -> Union($,"0")`	Differentiates a jet variables with respect to the independent variable labeled by the second argument.
`dimJ`	`NNI -> NNI`	Returns the (fiber) dimension of the jet bundle of the given order.
`dimS`	`NNI -> NNI`	Returns the dimension of $S_qT^*X \otimes V\mathcal{E}$ for the given value of q.
`getNotation`	`() -> S`	Shows the currently used notation.
`index`	`$ -> PI`	Returns the upper index of a jet variable.
`integrate`	`($, PI) -> $`	Integrates a jet variable with respect to an independent variable. Yields an error if not possible.
`integrateIfCan`	`($, PI) -> Union($,"failed")`	Integrates a jet variable with respect to the independent variable labeled by the second argument.
`multiIndex`	`$ -> L NNI`	Returns the multi-index of a jet variable.
`m2r`	`L NNI -> L PI`	Transforms a multi-index into a repeated index.
`name`	`$ -> S`	Returns the name of a jet variables.
`numDepVar`	`() -> PI`	Returns the number of dependent variables.
`numIndVar`	`() -> PI`	Returns the number of independent variables.
`one?`	`$ -> B`	Checks whether its argument is the “jet variable” 1.
`order`	`$ -> NNI`	Order of a jet variable (0 for non-derivatives).
`P`	`(PI, L NNI) -> $`	Generates a derivative. The interpretation of the second argument depends on the currently used notation.
`Pm`	`(PI, L NNI) -> $`	Generates a derivative with a given multi-index.
`Pr`	`(PI, L NNI) -> $`	Generates a derivative with a given repeated index.
`repeatedIndex`	`$ -> L PI`	Returns the multi-index of a jet variable in repeated index notation.
`r2m`	`L PI -> L NNI`	Transforms a repeated index into a multi-index.
`setNotation`	`S -> S`	Chooses the notation used for derivatives. Returns the old one. Possible values are `Repeated` and `Multi`.
`type`	`$ -> S`	Yields the type of a jet variable. Possible values are `Const`, `Indep`, `Dep`, `Deriv`.
`U`	`PI -> $`	Generates a dependent variable.
`variables`	`NNI -> L $`	Returns an ordered list of all jet variables up to the given order.
`variables`	`(NNI, PI) -> L $`	Returns all jet variables of the given order whose class is greater than or equal to a given value.
`weight`	`$ -> NNI`	Computes the weight assigned to a jet variable in the used ranking.
`X`	`PI -> $`	Generates an independent variable.
`1`	`() -> $`	Generates the special “jet variable” 1.

Figure D.1: Procedures in `JetBundleCategory`.

`autoReduce`	`L $ -> L $`	Reduces a system with respect to itself.
`class`	`$ -> NNI`	Class of an expression.
`const?`	`$ -> B`	Checks whether an expression depends on jet variables.
`coerce`	`JB -> $`	Transforms a jet variable into a function.
`denominator`	`$ -> $`	Denominator of an expression.
`differentiate`	`($, JB) -> $`	Differentiation with respect to a jet variable.
`dimension`	`(L $, SEM, NNI) -> NNI`	Dimension of a system with given Jacobi matrix in the jet bundle of given order.
`dSubst`	`($, JB, $) -> $`	Like `subst` but takes also derivatives into account.
`extractSymbol`	`SEM -> SEM`	Extracts symbol from Jacobi matrix.
`formalDiff`	`($, PI) -> $`	Formal differentiation with respect to an independent variable given by its label. There exist further mode maps for systems and with more detailed output. In some versions it is also possible to pass the Jacobi matrix as additional argument.
`freeOf?`	`($, JB) -> B`	Checks whether an expression depends on a given jet variable.
`jacobiMatrix`	`L $ -> SEM`	Computes Jacobi matrix of a system.
`jacobiMatrix`	`(L $, L L JB) -> SEM`	Computes Jacobi matrix of a system. For each equation the occurring jet variables are given.
`jetVariables`	`$ -> L JB`	Returns all jet variables effectively occurring in an expression.
`leadingDer`	`$ -> JB`	Leading derivative of an expression.
`numerator`	`$ -> $`	Numerator of an expression.
`order`	`$ -> NNI`	Highest order of the effectively occurring jet variables.
`orderDim`	`(L $, SEM, NNI) -> NNI`	Dimension of a given system taking only the given order into account.
`reduceMod`	`(L $, L $) -> L $`	Reduces a system with respect to another one.
`simplify`	`(L $, SEM) -> SIMPREC`	Simplifies a system with given Jacobi matrix. Returns the simplified system and its Jacobi matrix. Additionally it tries to trace the dependency of the returned equations of the original equations.
`simpOne`	`$ -> $`	Simplifies individual equations (removes unnecessary coefficients and signs etc.).
`simpMod`	`(L $, L $) -> L $`	Like `reduceMod`, but does not use derivatives of equations.
`simpMod`	`(L $, SEM, L $) -> SIMPREC`	Simplifies a system modulo another one. Returns also the changed Jacobi matrix.
`solveFor`	`($, JB) -> Union($,"failed")`	Solves an expression for a given jet variables if possible.
`sortLD`	`L $ -> L $`	Sorts a list of functions according to their leading derivatives.
`subst`	`($, JB, $) -> $`	Substitutes a given expression for a jet variable in another expression.
`symbol`	`L $ -> SEM`	Computes symbol of a system.

Figure D.2: Procedures in `JetBundleFunctionCategory`.

`analyseSymbol`	`SEM -> MVREC`	Computes a row echelon form of a given symbol. Counts the multiplicative variables and determines the rank of the matrix.
`copy`	`$ -> $`	Returns a copy of a differential equation.
`dimension`	`($, NNI) -> NNI`	Computes the dimension of a given differential equation considered as submanifold of a jet bundle of given order.
`display`	`$ -> OUT`	Prints most of the information stored about a differential equation: equations ordered by their order, for each order the Jacobi matrix, whether the system is already simplified and so on.
`extractSymbol`	`($, B) -> SEM`	Extracts the symbol from the Jacobi matrix of the highest order part of the equation. If the second argument is true, its row echelon form is computed.
`generate`	`L D -> $`	Generates a differential equation from a given system.
`insert`	`(L D, $) -> $`	Inserts additional equations into a differential equation.
`jacobiMatrix`	`$ -> L SEM`	Returns the Jacobi matrices separately for each order.
`join`	`($, $) -> $`	Combines two differential equations into one.
`order`	`$ -> NNI`	Returns the order of a differential equation.
`project`	`($, NNI) -> $`	Projects a differential equation to a given order. Assumes the equation was previously prolonged.
`prolong`	`$ -> SREC`	Prolongs all equations in a differential equation which have not already been prolonged, i.e. especially all equations of highest order. Found integrability conditions are returned separately.
`prolong`	`($, NNI) -> SREC`	Like `prolong`. Considers, however, only equations below a given order.
`prolongMV`	`MVREC -> MVREC`	Calculates directly the multiplicative variables for a prolonged symbol. Result only valid, if the symbol is involutive.
`prolongSymbol`	`SEM -> SEM`	Prolongs directly a given symbol.
`printSys`	`L D -> OUT`	Writes a list of functions as a vector of equations (with right hand side 0) and coerces the result to `OUT`.
`retract`	`$ -> L D`	Returns a list of all equations contained in a given differential equation.
`setSimpMode`	`NNI -> NNI`	Sets the simplification mode. If it is greater than zero, then equations are simplified modulo lower order equations.
`simplify`	`$ -> SREC`	Simplifies the equations order by order. Found integrability conditions are returned separately.
`tableau`	`(SEM, DIFF) -> SEM`	Computes the tableau to a given symbol parameterized by a given one-form.
`tableau`	`(SEM, L DIFF) -> SEM`	Computes the k-tableau to a given symbol parameterized by a given list of one-form.

Figure D.3: Procedures in `DifferentialEquation`.

`coefficient`	`($, JB) -> D`	Returns the coefficient in a given direction.
`coefficients`	`$ -> L D`	Returns the coefficients of a vector field.
`commutator`	`($, $) -> $`	Computes the commutator of two given vector fields.
`copy`	`$ -> $`	Returns a copy of a given vector field.
`diff`	`JB -> $`	Generates base vector field with given direction.
`diffP`	`(PI, L NNI) -> $`	Generates base vector field in direction of a derivative.
`diffU`	`PI -> $`	Generates base vector field in direction of a dependent variable.
`diffX`	`PI -> $`	Generates base vector field in direction of an independent variable.
`directions`	`$ -> L JB`	Returns a list of the directions of the base vectors where the vector fields has non-vanishing coefficients.
`eval`	`($, D) -> D`	Applies a vector field to a function.
`lie`	`($, $) -> $`	Computes the Lie derivative of a given vector field in the direction of another vector field.
`prolong`	`($, NNI) -> $`	Prolongs a vector field defined on $\mathcal{E}$ to a field on the jet bundle of the given order.
`table`	`L $ -> M $`	Computes the commutator table to a given list of vector fields.

Figure D.4: Procedures in `VectorField`.

`coefficient`	`($, JB) -> D`	Returns the coefficient of a given differential.
`coefficients`	`$ -> L D`	Returns a list with the coefficients of a given one-form.
`contract`	`(VF, $) -> D`	Contracts a vector field with a one-form (interior derivative).
`d`	`JB -> $`	Generates the differential of a given jet variable.
`d`	`D -> $`	Computes the differential of a function.
`differentials`	`$ -> L JB`	Returns a list of the base differentials with non-vanishing coefficients.
`dP`	`(PI, L NNI) -> $`	Generates the differential to a derivative.
`dU`	`PI -> $`	Generates the differential to a dependent variable.
`dX`	`PI -> $`	Generates the differential to an independent variable.
`eval`	`($, VF) -> D`	Applies a one-form to a vector field.
`lie`	`(VF, $) -> $`	Computes the Lie derivative of a one-form in direction of a given vector field.

Figure D.5: Procedures in `Differential`.

`alpha`	`(NNI, L NNI) -> L NNI`	Computes the Cartan characters for a differential equation of given order from the $\beta_q^{(k)}$.
`alphaHilbert`	`UP("r",FI) -> L NNI`	Compute the Cartan characters for a given Hilbert polynomial.
`arbFunctions`	`(NNI, I, L NNI) -> L I`	Uses the Cartan characters to compute the number of arbitrary functions of a fixed differentiation order for a differential equation of given order.
`bound`	`(NNI, NNI, NNI) -> NNI`	Computes the bound $\hat{q}(n, m, q)$ for the number of prolongations needed to render a symbol involutive using (2.27).
`complete`	`DE -> Void`	Completes a given differential equation to an involutive one. No result is returned, but information on the completion process are displayed. The amount of it can be controlled using `setOutput`.
`complete2`	`DE -> CREC`	Analog to `complete`. Returns a record containing the involutive equation, a generating set of equations, the order of the involutive equation, the number of projections needed, the dimension of the equation and its Cartan characters.
`gauge`	`(NNI, I, L NNI) -> L I`	Computes the gauge corrections to the number of arbitrary functions for given values of γ_l. The other arguments are as for `arbFunctions`.
`gaugeHilbert`	`(NNI, L NNI) -> UP("r",FI)`	Computes analog to `hilbert` the gauge correction to the Hilbert polynomial.
`hilbert`	`L NNI -> UP("r",FI)`	Computes the Hilbert polynomial for given Cartan characters.
`setOutMode`	`NNI -> NNI`	Controls the amount of output generated by `complete`. The following values are possible for the argument: 0 $\rightarrow$ no output 1 $\rightarrow$ result is displayed 2 $\rightarrow$ Cartan characters are displayed 3 $\rightarrow$ integrability conditions are traced 4 $\rightarrow$ intermediate dimensions are traced 5 $\rightarrow$ intermediate systems are traced 6 $\rightarrow$ intermediate symbols are traced Addition of 10 yields output in TEX.
`setRedMode`	`NNI -> NNI`	Sets reduction mode. If value greater than zero, then integrability conditions are reduced.
`setSimpMode`	`NNI -> NNI`	See `DifferentialEquation`.

Figure D.6: Procedures in `CartanKuranishi`.

`*`	`(M D, $) -> $`	Left multiplication of a sparse matrix with a usual matrix.
`*`	`(M F D, $) -> $`	Left multiplication of a sparse matrix with a usual matrix over the quotient field of `D`. Available only if `D` belongs to `IntegralDomain`.
`allIndices`	`$ -> L C`	Yields a list of all indices used to label columns.
`appendRow!`	`($, ROWREC) -> Void`	Adds a new row as last row. Argument is changed destructively.
`elimZeroCols!`	`$ -> Void`	Removes columns containing only zeros. This effects, however, basically only the value of `allIndices`.
`coerce`	`$ -> M D`	Coerces a matrix from `SEM` to the usual matrix type of AXIOM.
`consRow!`	`($, ROWREC) -> Void`	Adds a new row as first row. Argument is changed destructively.
`copy`	`$ -> $`	Yields a copy of a matrix.
`deleteRow!`	`($, I) -> Void`	Deletes the indicated row.
`elt`	`($, I, C) -> D`	Returns the entry in the given row and the column labeled by the given index.
`extract`	`($, I, I) -> $`	Returns a sub-matrix consisting of the indicated range of rows.
`horizJoin`	`($, $) -> $`	Concatenates horizontally two matrices. It is assumed that all indices of the second one are smaller than those of the first one.
`horizSplit`	`($, C) -> Record(Left:$, Right:$)`	Splits a matrix into two at the indicated column.
`join`	`($, $) -> $`	Concatenates vertically two matrices.
`ncols`	`$ -> NNI`	Returns the number of columns of a matrix.
`new`	`(L C, I) -> $`	Generates a new matrix with a given number of rows. The first argument contains the indices used to label the columns.
`nrows`	`$ -> NNI`	Returns the number of rows of a matrix.
`pivot`	`($, I) -> Record(Index:C, Entry:D)`	Returns the pivot of the indicated row (plus label for column).
`pivots`	`$ -> ROWREC`	Returns all pivots (plus labels for columns).
`primitiveRow-Echelon`	`$ -> Record(...)`	Like `rowEchelon`. Uses however a primitive fraction-free method, i.e. each row is divided by its gcd. Hence the returned transformation matrix contains fractions. Only available, if `D` belongs to `GcdDomain`.
`purge!`	`($, C->B) -> Void`	Eliminates all columns whose label satisfy the given criterion function.
`row`	`($, I) -> ROWREC`	Yields a row of a matrix
`rowEchelon`	`$ -> Record(...)`	Computes the row echelon form of a matrix using fraction-free elimination. Returns furthermore the transformation matrix, the used pivots, and the rank of the matrix.
`setElt!`	`($, I, C, D) -> Void`	Sets an entry to a given value. Row and column are indicated as in `elt`.
`setGcdMode`	`S -> S`	Sets flag for method to compute GCDs for lists.
`setRow!`	`($, I, L C, L D) -> Void`	Sets a whole row given as lists of entries and labels, respectively.
`setRow!`	`($, I, ROWREC) -> Void`	Sets a whole row given as a record.
`sortedPurge!`	`($, C->B) -> Void`	Like `purge!` assumes, however, that the criterion respects the ordering of the labels.

Figure D.7: Procedures in `SparseEchelonMatrix`.

`ansatz`	`() -> VF`	Yields the most general ansatz for a symmetry generator.
`detSys`	`(L JBE1, L JB, VF) -> L JBE1`	Computes the determining system for a given system. The second argument contains a list of derivatives for which the equations can be solved. If it is omitted the leading derivatives are used. The third argument contains an ansatz for the generators. It can also be omitted. `ansatz()` is then used.
`linearize`	`L JBE2 -> L JBL2`	Retracts equations to linear ones, if possible.
`ncDetSys`	`(L JBE1, L JB, VF) -> L JBE1`	Computes the determining system for conditional symmetries. The meaning of the arguments is as in `detSys`.
`transform`	`JBE1 -> JBE2`	Transforms between the different jet bundles involved.

Figure D.8: Procedures in `SymmetryAnalysis`.

`transform`	`JB1 -> E2`	Transforms a jet variable in the old coordinates into an expression in the new ones.
`transform`	`E1 -> E2`	Transforms an expression in the old coordinates into the new ones.

Figure D.9: Procedures in `JetCoordinateTransformation`.

`LUDecomp`	`M D -> Record(...)`	Computes the LU decomposition of a given matrix. Returns a record with a matrix containing the two triangular ones, the permutation needed for partial pivoting, and the used pivots.
`LUSolve`	`(M D, V I, V D) -> V D`	Expects a matrix in LU decomposition, the permutation used for pivoting, and a right hand side. Returns the solution vector.
`LUInverse`	`M D -> Record(...)`	Computes the inverse of a matrix using an LU decomposition. Besides the inverse the used pivots are returned.

Figure D.10: Procedures in `LUDecomposition`.