# Analysis and Application of the Formal Theory of Partial Differential Equations


by

Dipl. Phys.

**Werner Markus Seiler**

School of Physics and Materials
Lancaster University
and
Institut für Algorithmen und Kognitive Systeme
Universität Karlsruhe



Thesis submitted for the degree of Doctor of Philosophy
Lancaster University — September 1994

## Abstract

An introduction to the formal theory of partial differential equations is given emphasizing the properties of involutive symbols and equations. An algorithm to complete any differential equation to an involutive one is presented. For an involutive equation possible values for the number of arbitrary functions in its general solution are determined. The existence and uniqueness of solutions for analytic equations is proven. Applications of these results include an analysis of symmetry and reduction methods and a study of gauge systems. It is shown that the Dirac algorithm for systems with constraints is closely related to the completion of the equation of motion to an involutive equation. Specific examples treated comprise the Yang-Mills Equations, Einstein Equations, complete and Jacobian systems, and some special models in two and three dimensions. To facilitate the involved tedious computations an environment for geometric approaches to differential equations has been developed in the computer algebra system AXIOM. The appendices contain among others brief introductions into Cartan-Kähler Theory and Janet-Riquier Theory.

# Contents

# List of Figures

# Chapter 1

# Introduction

*Though this be madness, yet there is method in't.*

*W. Shakespeare: Hamlet, II.2.205*

Systems of partial differential equations occur in all branches of physics (and many other sciences). Whenever a continuous process must be modeled mathematically, there is a high probability that differential equations appear. Thus it is not surprising that they represent one of the vastest fields in mathematics. Although in the quotation above Polonius was probably thinking more of Hamlet's state of mind than of differential equations, it characterizes nicely the approach we will take.

Systems of partial differential equations display such a variety of phenomena that it may really appear as madness to try to tackle general systems. But the main feature of our approach will be that it provides us with a method, i.e. a *systematic* way, to treat such systems. This has on one side the advantage that we will be able to unify apparently very different computations for which previously individual techniques had to be found. On the other hand it might often be more efficient to employ some special ad hoc tricks.

The main emphasis in this thesis will be put on initial value problems, i.e. on problems where the initial state of a system is given and a system of differential equations describes its dynamics. One of the central questions will be, how much and what data is necessary to specify an initial state such that the subsequent evolution is unique.

Although this represents a really fundamental problem surprisingly little is known. And this knowledge is not very wide spread. It is well-known that an ordinary differential equation of order $q$ possesses $q$ functionally independent solutions and that this implies that $q$ constants can be prescribed as initial data. This generalizes to a *single* partial differential equation where $q$ functions defined on a submanifold of codimension 1 can be given as initial data. In typical physical applications this is a space-like submanifold of space-time described in some local coordinate system by the equation $t = t_0$.

As soon as one proceeds to *systems* of partial differential equations, the situation becomes considerably more complicated. Now only for so-called normal systems, i.e. systems which can be brought into a form satisfying the conditions of the Cauchy-Kowalevsky Theorem, we obtain a direct generalization of the previous cases, namely if the system contains $m$ unknowns or dependent variables, one can impose $mq$ functions defined on a submanifold of codimension 1 as initial conditions.

However, one encounters in physics many non-normal systems; e.g. no gauge systems can be of this form. This implies that if one still tries to prescribe initial data on a submanifold of codimension 1, then this initial data can no longer be chosen arbitrarily. The conventional way to deal with this effect is to say that one has a system with *constraints*. Many papers have been devoted to the problem of solving the constraints for specific systems like Einstein's field equations in general relativity.

One of the main results of this thesis will be to propose another approach. Instead of insisting on prescribing initial data on one submanifold of codimension 1 we will use a flag of nested submanifolds of different dimensions, i.e. some initial conditions are given as usual on a submanifold of codimension 1, others are given on one of codimension 2 and so on. The main advantage of this ansatz is that the initial data can be chosen completely arbitrary, they do not have to satisfy any constraints.

This does not imply that we have implicitly solved the constraints. What happens is that we do not draw the in many respect artificial distinction in evolutionary equations and constraints. All differential equations are treated on the same footing. This approach should also be interesting for numerical calculations, although we will not pursue this point.

Neither will we consider the problem of quantization, but it is worth while pointing out that our approach might be of considerable interest in this context. Since the initial data are not constrained, they can be considered as an explicit parameterization of the state space which could serve as a starting point for a canonical (or geometric) quantization.

It turns out that the notion of an *involutive* differential equation is of central importance for such a study. Although this concept (and related ones) was already introduced early in this century, it is not very well-known. It is very difficult to find a textbook on differential equations mentioning it at all. Only older books often contain a discussion of complete and Jacobian systems. As we will show in Section 7.1, this is equivalent to studying involution restricted to linear, first order equations in one dependent variable.

One of the goals of this thesis is to show that involution is an important and useful concept for mathematical physics. We will for instance see in Section 6.3 that the well-known Dirac algorithm to compute all constraints of a dynamical system described by a Lagrangian makes nothing else than to complete the equations of motion to an involutive system. This has important implications for the generalization to field theories.

In several respects the here presented theory could be considered as "trivial": We will treat only local problems, although every differential geometer knows that the really difficult questions are the global ones; we will almost all the time restrict our attention to analytic functions, although it is well-known that local solvability becomes really challenging only for smooth or worse functions (and we will not mention at all distributions or Sobolev theory!); finally we will not even try to solve differential equations. We hope, however, that we will be able to convince the reader that there still remain many interesting questions which may even be relevant to applications.

Since all analysis will be of a completely local character, we will often be negligent in our notations. Especially, we almost always omit to denote neighborhoods. In many places one should, strictly seen, read "local neighborhood" instead of "manifold". A similar remark holds for the occurring bundles. Their global structure is never considered and we always operate as if they were trivial. Since in many applications all manifolds will be of the form $\mathbb{R}^n$ and the coordinate systems can be used globally, we do not think that this "sloppiness" harms.

In the geometric approach to partial differential equations one does not distinguish between single equations and systems of equations, as one deals with a geometric object, namely a submanifold in a jet bundle. We will try to reflect this definition in our terminology. *Equation* usually refers to such a submanifold, whereas *system* means a local description of it. However, systems usually consist of equations, thus it is not really possible to maintain such a distinction...

The second chapter contains a detailed introduction into the formal approach to partial differential equations. It explains the fundamental notions and theorems centering about the problem of completion to an involutive equation. Comparing with the books of Pommaret [117, 118, 119] we present the theory more from a computational point of view. Thus all our proofs use local coordinates and operate directly on the equations, whereas he prefers the abstract approach of diagram chasing which is much more powerful for proofs but less useful for practical computations. Due to its introductory character most of this chapter repeats well-known results. New ones can be found in the last two sections, especially the $k$-tableaux seem to be a new tool.

Chapter 3 applies the machinery of formal theory to an analysis of "general solutions". This requires first a definition of this term. We follow here ideas of Ampère and Hermann and translate them into our geometric language. This definition is surely of more theoretical than practical interest. The main part of this chapter consists of analyzing the number of arbitrary functions which can occur in a representation of a general solution. These results have already been published in [138].

As the results of the previous chapter have been completely formal, we turn our attention to initial value problems in Chapter 4. As already mentioned our approach here is fairly different from the standard one. We give a new intrinsic description of formally well-posed problems, where exactly the right amount of Cauchy data is prescribed on a flag of submanifolds. In the analytic category we prove the existence and uniqueness of solutions: the Cartan-Kähler Theorem. This theorem is well-known in the framework of exterior systems; a coordinate dependent version can be found in [117] from where we have taken our proof. For linear systems we extend the classical Holmgren Theorem from normal systems to arbitrary involutive ones.

The application of the machinery developed in the previous chapters starts in Chapter 5 with an analysis of symmetries and different reduction methods for partial differential equations. After a brief introduction into the basic notions of this field we consider the loss of generality occurring during a reduction and the task of finding differential constraints. Whereas we can present some definite results for the former point, published in [137], the latter one remains a big problem.

Chapter 6 is the first one which contains applications to physics, namely to field theories. It compares our approach to the arbitrariness of the general solution with the one taken by Einstein [47] using the strength of a system. We show, how one can subtract the effect of a gauge symmetry on the arbitrariness. The theory of systems with constraints is studied extensively from a formal point of view. As specific examples the Yang-Mills and the Einstein Equations are analyzed. Some results have already been published in [137, 138].

Some miscellaneous applications are collected in Chapter 7. It starts with reviewing the theory of complete and Jacobian systems from a formal point of view. These systems appear in many places in physics and mathematics. Some of the results have already been used in Chapter 5. Then we study the effect of adding some topologically motivated terms to Yang-Mills and Einstein Equations, respectively, in 2+1 dimensions. These models have interesting physical properties, quite different from the standard theories. After a discussion of the two-dimensional dilaton-gravity we finally treat an example stemming from variational calculus.

Since formal computations tend to become very soon very tedious, we have developed in the computer algebra system AXIOM an environment for them. It is described in Chapter 8. After a brief introduction to AXIOM which emphasizes the differences between this system and others, more "standard" ones, a detailed overview of the implementation is given which can also serve as a kind of user's manual. The application of the environment is demonstrated on several examples. An earlier version of it was presented in [132].

Two appendices briefly describe alternative approaches to the formal analysis of partial differential equations, namely the Cartan-Kähler Theory and the Janet-Riquier Theory, as both have decisively influenced the development of the formal theory. Especially our computational point of view draws heavily on the Janet-Riquier approach. A third appendix contains some smaller calculations concerning the so-called modified Stirling numbers which are needed in Chapter 3 and the reduction of arbitrary equations to first order ones. We also provide a kind of reference manual for our AXIOM environment in form of an appendix.

Finally, a few words of thank are appropriate. I have worked on this thesis at three different universities with three different supervisors. It is a pleasure to thank them all: Prof. J. Calmet at the Institut für Algorithmen und Kognitive Systeme in Karlsruhe for his continuous support and his patience during the long initial phase without any results; Prof. P. Winternitz at the Centre de recherches mathématiques in Montréal for his hospitality and many interesting discussions during my stay there; and last but not least Prof. R.W. Tucker at the School of Physics and Materials in Lancaster for the surprising invitation to come here and his readiness to supervise the final stage of my thesis on short notice. I learned a lot from all of them.

I would also like to thank the many colleagues I met at the different universities for creating a warm atmosphere and for many fruitful discussions. I cannot name them all, but I want to mention especially Prof. J.F. Pommaret for introducing me into the world of formal theory and for many patient explanations and Prof. G.W. Bluman and Dr. G.J. Reid for inviting me to Vancouver and for giving me an occasion to present some of my results in a seminar series, especially most of Chapter 1 was written on this occasion.

I thank the Studienstiftung des Deutschen Volkes for their long and generous financial support and the School of Physics and Materials in Lancaster for a postgraduate grant.

Last but no way least I must thank my family and especially my wife for patiently bearing the live with a physicist. Only their love and support made it all possible.

# Chapter 2

# Formal Theory

*We give a general introduction into the ideas and the machinery of the formal theory of partial differential equations. We will define involutive equations and show how to construct power series solutions for them. We will study the properties of involutive symbols and show how to complete a given equation to an involutive one. Introducing tableaux we will be able to define characteristics, over- and underdeterminacy and tackle $\delta$-regularity.*

## 2.1 Geometric Theory

The formal theory is based on a differential geometric approach to differential equations using the jet bundle formalism. An exact introduction to this framework can be found in [117, 130]; here we will only give an informal description mainly in local coordinates. It should, however, be noted that all constructions can be performed in an intrinsic, coordinate-free manner.

The basic idea of jet bundles is to consider the derivatives of the dependent variables with respect to the independent ones as additional, *algebraically* independent variables. This gives naturally rise of a bundle structure. The starting point is a bundle $\pi : \mathcal{E} \to X$ with the independent variables $x^i,\ i = 1, \dots, n$, as coordinates of the base space and the dependent variables $u^\alpha,\ \alpha = 1, \dots, m$, as fiber coordinates.

Over this bundle we construct further bundles: The first order jet bundle $J_1\mathcal{E}$ has $X$ as base and as fiber coordinates the dependent variables plus the first order derivatives $p_i^\alpha = \partial u^\alpha / \partial x^i$. Correspondingly, as fiber coordinates of the $q$-th order jet bundle $J_q\mathcal{E}$ all derivatives up to order $q$ are occurring. We can also use a lower order jet bundle as base space instead of $X$; the canonical projection is then denoted by $\pi_q^{q+r} : J_{q+r}\mathcal{E} \to J_q\mathcal{E}$. A simple combinatorial calculation yields the (fiber) dimension of $J_q\mathcal{E}$:

$$\dim J_q\mathcal{E} = m\binom{n+q}{q}. \tag{2.1}$$

We will mainly use the standard multi-index notation of multivariate calculus. A multi-index is an $n$-tuple $\mu = [\mu_1, \dots, \mu_n]$ and its length is defined by $|\mu| = \mu_1 + \cdots + \mu_n$. Then the jet variable $p_\mu^\alpha$ represents the derivative $\partial^{|\mu|} u^\alpha / \partial(x^1)^{\mu_1} \cdots \partial(x^n)^{\mu_n}$. Multi-indices can be added and subtracted like vectors provided all entries remain non-negative. As shorthand we write $\mu \pm a_i$ for the multi-index $[\mu_1, \dots, \mu_i \pm a, \dots, \mu_n]$. In concrete examples, we will often use the notation $p^\alpha_{ij\dots}$ for a derivative with respect to $x^i, x^j, \dots$ This should not lead to confusion, as multi-indices will always be denoted by greek letters and the indices of the independent variables by Latin letters.

Sections of the bundle $\mathcal{E}$ can be considered as natural generalization of functions (or better: of their graphs). They are defined as mappings $\sigma : X \to \mathcal{E}$ satisfying $\pi \circ \sigma = \mathrm{id}_X$; i.e. in local coordinates $\sigma$ has the form $\sigma : x \mapsto (x, f(x))$. The sheaf of all local sections of $\mathcal{E}$ will be denoted by $\Gamma_{loc}(\mathcal{E})$; local meaning that a section $\sigma$ needs to be defined only in a subset $U_\sigma \subset X$. We can *prolong* a section to a section of the $q$-th order jet bundle. Locally this is simply done by adding the derivatives of $f$: $j_q(\sigma) : x \mapsto \left(x, f(x), \partial^{|\mu|} f(x)/\partial^\mu x\right)$. This explains also a different point of view of jets. There a point $(x^i, u^\alpha, p^\alpha_\mu)$ in $J_q\mathcal{E}$ is considered as an equivalence class of functions with the same power series expansion up to order $q$ around the point $x$.

**Definition 2.1** *A (non-linear)* differential equation *of order $q$ is a fibered submanifold $\mathcal{R}_q$ of $J_q\mathcal{E}$. A* solution *is a section $\sigma \in \Gamma_{loc}(\mathcal{E})$ such that its prolongation $j_q(\sigma)$ lies entirely in $\mathcal{R}_q$.*

Fibered means that $\mathcal{R}_q$ can be projected onto the base space $X$. In other words, the differential equation poses no restrictions on the independent variables.This definition, which does not distinguish between a single equation and a system of equations, is motivated by two main ideas: The differential equation is "reduced" to an algebraic equation in $J_q\mathcal{E}$ and we are now treating a geometric object — namely a submanifold — which is independent of a particular set of equations or a particular coordinate system. If $\mathcal{E}$ is a vector bundle, then linear equations possess in addition the structure of a vector subbundle.

Usually we will be given a map with local representation

$$\Phi : \begin{cases} J_q\mathcal{E} & \longrightarrow & \mathcal{E}' \\ (x^i, u^\alpha, p^\alpha_\mu) & \longmapsto & \Phi^\tau(x^i, u^\alpha, p^\alpha_\mu) \end{cases} \tag{2.2}$$

where $\mathcal{E}'$ is another bundle over $X$ such that $\mathcal{R}_q$ is the kernel of this map. For topological reason not every differential equation in the sense of Definition 2.1 has such a local representation [117]. Since, however, all equations treated in the sequel are of this form, we will always assume the existence of such a mapping $\Phi$ and furthermore that this description is non-degenerate, i.e. the functions $\Phi^\tau$ are functionally independent. Typically $\mathcal{E}$ will be of the form $\mathbb{R}^{m+n}$ and $\mathcal{E}'$ of the form $\mathbb{R}^p$ and one can use the functions $\Phi^\tau$ globally.

There are two basic geometric operations that can be performed with a differential equation: We can *prolong* it into a higher-order jet bundle or we can *project* it into a lower-order one. Prolongation corresponds of course to differentiation of the equations. In the jet bundle formalism the *formal derivative* serves this purpose:

$$D_i\Phi^\tau(x^i, u^\alpha, p^\alpha_\mu) = \frac{\partial \Phi^\tau}{\partial x^i} + \sum_\alpha \frac{\partial \Phi^\tau}{\partial u^\alpha} p^\alpha_i + \sum_{\alpha,\mu|} \frac{\partial \Phi^\tau}{\partial p^\alpha_\mu} p^\alpha_{\mu+1_i}\,. \tag{2.3}$$

This definition reflects the interpretation of the jet variables $p^\alpha_\mu$ as derivatives of the dependent variables $u^\alpha$ and respects the chain rule.

The prolongation $\mathcal{R}_{q+1} \subset J_{q+1}\mathcal{E}$ of the differential equation $\mathcal{R}_q$ is defined by the combined set of equations $\Phi^\tau = 0,\ D_i\Phi^\tau = 0$ for $i = 1, \dots, n$. An intrinsic definition is given by $\mathcal{R}_{q+r} = J_r\mathcal{R}_q \cap J_{q+r}\mathcal{E}$. The intersection means essentially that we identify derivatives of the derivatives in $J_r\mathcal{R}_q$ with higher-order derivatives of the original dependent variables $u^\alpha$. This is necessary, as they are otherwise treated as additional dependent variables.

In the sequel, we will assume throughout that $\mathcal{R}_q$ is a *regular equation*, i.e. all prolongations define again fibered submanifolds. This does not hold automatically, as the following example will show, but can always be achieved by using an appropriate restriction of $\mathcal{R}_q$.

**Example 2.2**

$$\mathcal{R}_2 : \begin{cases} u_{yy} - \frac{1}{2}(u_{xx})^2 &= 0\,, \\ u_{xy} - u_{xx} &= 0\,. \end{cases} \tag{2.4}$$

Adding the $x$- and $y$-derivative of the second equation and subtracting the $x$-derivative of the first one yields the equation $(u_{xx}-1)u_{xxx} = 0$. Thus $\mathcal{R}_3$ is not a fibered submanifold, as its dimension changes wherever $u_{xx} = 1$. We must consider two distinct cases, in which the differential equation can show quite different behavior.

At first sight, it may surprise that the projection $\mathcal{R}_q^{(1)} = \pi_q^{q+1}(\mathcal{R}_{q+1}) \subset J_q\mathcal{E}$ of the prolongation of a differential equation does not always recover the original equation $\mathcal{R}_q$ but sometimes only a subset. This signals the existence of integrability conditions. In terms of the local representation $\Phi^\tau = 0$ they arise mostly as linear combinations of prolonged equations (in the simplest case as cross-derivatives) and represent additional equations which are automatically satisfied by any solution of the original differential equation but which are nevertheless algebraically independent of it. More generally, we introduce $\mathcal{R}_{q+r}^{(s)} = \pi_{q+r}^{q+r+s}(\mathcal{R}_{q+r+s}) \subset J_{q+r}\mathcal{E}$.

**Example 2.3** A trivial example may explain the phenomenon. Consider the first-order system

$$\mathcal{R}_1 : \begin{cases} u_z + yu_x &= 0\,, \\ u_y &= 0\,. \end{cases} \tag{2.5}$$

Its prolongation $\mathcal{R}_2$ contains the equation $u_{yz} + yu_{xy} + u_x = 0$ (from the first equation). It follows, however, from the second equation that both second-order derivatives vanish. Hence, we get the integrability condition $u_x = 0$, which must be added to $\mathcal{R}_1$ to get $\mathcal{R}_1^{(1)}$. It is important to note that it is not possible to construct this equation without differentiation.

Whereas this corresponds exactly to what is usually called an integrability condition, in the geometric theory there exists a second reason why the projection may not be surjective, namely if the local representation of $\mathcal{R}_q$ contains equations of different order. The prolongation of these yields equations of order less than or equal to $q$, thus they have to be taken into account when determining $\mathcal{R}_q^{(1)}$.

**Example 2.4** Consider the simple second order equation

$$\mathcal{R}_2 : \begin{cases} u_{xx} &= 0\,, \\ u_y &= 0\,. \end{cases} \tag{2.6}$$

$\mathcal{R}_3$ contains also the equations $u_{xy} = u_{yy} = 0$. Thus the projection is not surjective and we have to add these two equations to $\mathcal{R}_2$ to obtain $\mathcal{R}_2^{(1)}$.

The occurrence of integrability conditions thoroughly disturbs the construction of power series solutions order by order. To build such a solution for an equation $\mathcal{R}_q$ one starts with the coefficients up to order $q$. Some of them can be freely chosen; the other ones can be expressed by these using the differential equation. Analogously one proceeds with the coefficients of the next order; this time applying the prolonged equation $\mathcal{R}_{q+1}$. (We will detail this construction in Section 2.5). If there occur, however, some integrability conditions during the prolongations, we suddenly get additional equations for the coefficients of *lower order*. Thus the construction of a power series solution order by order can be performed only for an equation containing all its integrability conditions.

**Definition 2.5** *A differential equation $\mathcal{R}_q$ is* formally integrable, *if $\mathcal{R}^{(1)}_{q+r} = \mathcal{R}_{q+r}$ for all $r \geq 0$, i.e. if it contains all its integrability conditions.*

## 2.2 Involutive Equations

Formally integrable equations are often called involutive. This is, however, not correct. Involution comprises formal integrability but requires additional properties which are of a more algebraic nature. To give an exact definition we have to introduce the symbol of a differential equation which is closely connected with the construction of power series solutions.

**Definition 2.6** *Let the differential equation $\mathcal{R}_q$ be locally described by $\Phi^\tau(x^i, u^\alpha, p^\alpha_\mu) = 0$ for $\tau = 1, \dots, p$. The solution space $\mathcal{M}_q$ of the following system of linear equations in the unknowns $v^\alpha_\mu$ with $\alpha = 1, \dots, m$, $|\mu| = q$*

$$\mathcal{M}_q \;:\; \left\{ \sum_{\alpha, |\mu| = q} \left( \frac{\partial \Phi^\tau}{\partial p^\alpha_\mu} \right) v^\alpha_\mu = 0\,, \quad \tau = 1, \dots, p \right. \tag{2.7}$$

*is called the* symbol *of $\mathcal{R}_q$. (By abuse of language, we will often also call the linear system symbol instead of its solution space).*

Differential geometrically seen the symbol is a family of vector spaces over $\mathcal{R}_q$. It can be defined intrinsically as $\mathcal{M}_q = V\mathcal{R}_q \cap (S_qT^*X \otimes V\mathcal{E})$ where $S_q$ denotes the $q$-fold symmetric product, $T^*X$ the cotangent bundle of $X$ and $V$ the verticle bundle[1]. Alternatively we can use the local representation (2.2) of $\mathcal{R}_q$ and introduce the symbol map

$$\sigma \;:\; \left\{ \begin{array}{ccc} S_qT^*X \otimes V\mathcal{E} & \longrightarrow & \mathcal{E}' \\ v^\alpha_\mu & \longmapsto & \displaystyle\sum_{\alpha, |\mu| = q} \left( \frac{\partial \Phi^\tau}{\partial p^\alpha_\mu} \right) v^\alpha_\mu \end{array} \right. . \tag{2.8}$$

It is easy to see that this map is intrinsically defined, i.e. independent of the choice of the local representation $\Phi$, and that its kernel is just $\mathcal{M}_q$.

(2.7) represents at each point of $\mathcal{R}_q$ a linear system in a vector space of the dimension

$$\dim S_qT^*X \otimes V\mathcal{E} = m \binom{n+q-1}{n-1} . \tag{2.9}$$

This is of course nothing else than the number of derivatives $p^\alpha_\mu$ of order $q$. We will make the assumption that $\mathcal{M}_q$ is a vector bundle over $\mathcal{R}_q$, i.e. the dimensions of the vector spaces are constant. Again this can always be achieved by taking an appropriate restriction.

The formal derivative applied to a function over $J_q\mathcal{E}$ yields a function on $J_{q+1}\mathcal{E}$ which is *linear* in the derivatives of order $q+1$. This allows to compute directly without further differentiations the equations of the prolonged symbols $\mathcal{M}_{q+r}$. Applying Definition 2.6 to $\mathcal{R}_{q+1}$ yields the equations

$$\mathcal{M}_{q+1} \;:\; \left\{ \sum_{\alpha, |\nu| = q+1} \left( \frac{\partial D_i \Phi^\tau}{\partial p^\alpha_\nu} \right) v^\alpha_\nu = 0\,, \quad \begin{array}{rcl} \tau &=& 1, \dots, p\,, \\ i &=& 1, \dots, n\,. \end{array} \right. \tag{2.10}$$

[1] Here we use the fact that $J_q\mathcal{E}$ is an affine bundle over $J_{q-1}\mathcal{E}$ modeled on the vector bundle $S_qT^*X \otimes V\mathcal{E}$. This follows easily from the transformation and symmetry properties of $q$-th order derivatives [117]. One can thus consider $S_qT^*X \otimes V\mathcal{E}$ as the "vector space of all $q$-th order derivatives".

But by the definition (2.3) of the formal derivative, $\partial D_i\Phi^\tau/\partial p^\alpha_\nu$ can be different from zero only, if there exist multi-indices $\mu$ of order $q$ with $\nu = \mu + 1_i$ for some $i$ such that $\partial\Phi^\tau/\partial p^\alpha_\mu \neq 0$ . The same argument holds of course for higher prolongations, too. Thus we arrive at the simple formula

$$\mathcal{M}_{q+r} \ : \left\{ \sum_{\alpha,|\mu|=q} \left(\frac{\partial\Phi^\tau}{\partial p^\alpha_\mu}\right) v^\alpha_{\mu+\nu} = 0\,, \qquad \begin{array}{rcl} \tau &=& 1,\dots,p\,, \\ |\nu| &=& r\,. \end{array} \right. \tag{2.11}$$

The notation sometimes gives the impression that the matrices of the linear systems $\mathcal{M}_q$ and $\mathcal{M}_{q+r}$ are identical. This is of course not true; the latter one is much larger. It contains more rows and columns, as it is defined by more equations in a larger vector space. But we can see that all rows with the same index $\tau$ contain the same entries, because the coefficients are independent of $\nu$, and these are basically the same entries as in the row $\tau$ of the first symbol $\mathcal{M}_q$. The only thing that happens is that they appear several times and that they are distributed in a different way.

The symbol indicates whether or not during a prolongation integrability conditions may have occurred and allows us to construct them using only linear algebra. To show this explicitly, we outline a proof of the next theorem.

**Theorem 2.7** *If $\mathcal{M}_{q+1}$ is a vector bundle, then*

$$\dim \mathcal{R}^{(1)}_q = \dim \mathcal{R}_{q+1} - \dim \mathcal{M}_{q+1}\,. \tag{2.12}$$

PROOF. We can compute the dimension of a submanifold by counting how many independent functions are needed to describe it locally, i.e. by the rank of a Jacobi matrix. The spaces occurring in (2.12) are defined by the following equations:

$$\mathcal{R}_q \ : \left\{ \ \Phi^\tau(x^i, u^\alpha, p^\alpha_\mu) = 0\,, \right. \tag{2.13}$$

$$\mathcal{R}_{q+1} \ : \left\{ \begin{array}{rcl} \Phi^\tau(x^i, u^\alpha, p^\alpha_\mu) &=& 0\,, \\ D_i\Phi^\tau(x^i, u^\alpha, p^\alpha_\mu) &=& 0\,, \end{array} \right. \tag{2.14}$$

$$\mathcal{M}_{q+1} \ : \left\{ \sum_{\alpha,|\mu|=q} \left(\frac{\partial\Phi^\tau}{\partial p^\alpha_\mu}\right) v^\alpha_{\mu+1_i} = 0\,. \right. \tag{2.15}$$

The Jacobi matrix of $\mathcal{R}_{q+1}$ can be divided into blocks

$$\left(\begin{array}{|c|c|c|} \hline \dfrac{\partial D_i\Phi^\tau}{\partial p^\alpha_\mu}\,,\ |\mu| = q+1 & \dfrac{\partial D_i\Phi^\tau}{\partial p^\alpha_\mu}\,,\ |\mu| \leq q & \dfrac{\partial D_i\Phi^\tau}{\partial u^\alpha} \\ \hline 0 & \dfrac{\partial\Phi^\tau}{\partial p^\alpha_\mu}\,,\ \ |\mu| \leq q & \dfrac{\partial\Phi^\tau}{\partial u^\alpha} \\ \hline \end{array}\right)\,. \tag{2.16}$$

The lower part stems from the original equations. Hence its rank is $\operatorname{codim}\mathcal{R}_q$ . The upper part comes from the prolonged equations. We consider first the left block. Comparison with (2.10) shows that it is just the symbol $\mathcal{M}_{q+1}$. We must now distinguish two cases. If this symbol has maximal rank, then $\dim\mathcal{R}_{q+1} = \dim\mathcal{R}_q + \dim\mathcal{M}_{q+1}$ and thus $\mathcal{R}^{(1)}_q = \mathcal{R}_q$ .

If, however, $\mathcal{M}_{q+1}$ has not maximal rank, then it is possible to construct by linear operations a row with only zeros in the first block. The question is now, how does this row look like in the two blocks to the right. If it is independent of the rows in the lower part, then we have found an integrability condition. It is constructed by performing exactly the same linear operations on the full equations $D_i\Phi^\tau = 0$. Otherwise, we have only obtained a redundant description of $\mathcal{R}_{q+1}$. This will lead either to *identities* or *compatibility conditions*, if we add a right hand side to the differential equation. Comparing ranks yields the claim for one point, but since $\mathcal{M}_{q+1}$ is a vector bundle, it holds everywhere. •

**Example 2.3 (cont.)** We can demonstrate this construction for our previous example. The first prolongation of this equation is given by

$$\mathcal{R}_2 : \begin{cases} u_z + yu_x = 0\,, & u_y = 0\,, \\ u_{xz} + yu_{xx} = 0\,, & u_{xy} = 0\,, \\ u_{yz} + yu_{xy} + u_x = 0\,, & u_{yy} = 0\,, \\ u_{zz} + yu_{xz} = 0\,, & u_{yz} = 0\,. \end{cases} \tag{2.17}$$

Obviously these equations are independent. With $\dim J_2\mathcal{E} = 10$ we get $\dim \mathcal{R}_2 = 10 - 8 = 2$. The equations for the symbol are

$$\mathcal{M}_2 : \begin{cases} v_{xz} + yv_{xx} = 0\,, & v_{xy} = 0\,, \\ v_{yz} + yv_{xy} = 0\,, & v_{yy} = 0\,, \\ v_{zz} + yv_{xz} = 0\,, & v_{yz} = 0\,. \end{cases} \tag{2.18}$$

The third equation is linearly dependent of the second and sixth. With $\dim S_2T^*X \otimes V\mathcal{E} = 6$ we find $\dim \mathcal{M}_2 = 6 - 5 = 1$. Applying Theorem 2.7 yields $\dim \mathcal{R}_1^{(1)} = 2 - 1 = 1$. The dimension of $\mathcal{R}_1$ is however 2. Hence an integrability condition has occurred and we can construct it by subtracting the eighth and $y$ times the fourth equation from the fifth in (2.17) getting $u_x = 0$.

To proceed further, we must introduce an apparently coordinate dependent notion: the *class* of a multi-index. We will study in Section 2.7 under what conditions the results are nevertheless intrinsic. The multi-index $\mu = [\mu_1, \mu_2, \ldots, \mu_n]$ is said to be of class $k$, if the first non-vanishing entry is $\mu_k$. We can order the columns of the symbol by the class of the multi-index of the corresponding $v_\mu^\alpha$ (the ordering within a class is of no importance) and then compute a row echelon form. In this form, the class of each row is given by the class of its first non-vanishing entry, the pivot. We associate with a row of class $k$ the *multiplicative variables* $x^1, x^2, \ldots, x^k$. The symbol is now in *solved form*. We denote by $\beta_q^{(k)}$ the number of rows of class $k$ [2].

The jet variables corresponding to the pivots are called *principal derivatives*, all others are *parametric derivatives*. The meaning of this distinction will become clearer in Section 2.5 where we construct power series solutions. Principal derivatives are obtained using the differential equations; the parametric ones must be prescribed by initial or boundary conditions. The idea is now to find a unique way to compute all derivatives of the principal derivatives. This leads to the notion of an *involutive* symbol.

An intrinsic definition of an involutive symbol can be obtained from the Spencer cohomology [58, 117]. Since this is fairly abstract, we will mention here only a simple criterion for an involutive symbol using the above introduced multiplicative variables. It can also be easily applied in concrete computations.

[2] Note that this number could take different values at different points of $\mathcal{R}_q$. But like we always assume that $\mathcal{M}_q$ is a vector bundle, we will always assume that the values of $\beta_q^{(k)}$ are constant on $\mathcal{R}_q$.

**Definition 2.8** *The symbol $\mathcal{M}_q$ is called* involutive, *if*

$$\operatorname{rank} \mathcal{M}_{q+1} = \sum_{k=1}^{n} k\beta_q^{(k)} . \tag{2.19}$$

*The differential equation $\mathcal{R}_q$ is called* involutive *or* in involution, *if it is formally integrable and if its symbol $\mathcal{M}_q$ is involutive.*

## 2.3 Involutive Symbols and Multiplicative Variables

The definition given at the end of the last section looks at first sight rather strange. It has, however, a straightforward explanation, which is so important that we will give it in form of a proposition for easier reference.

**Proposition 2.9** *If the symbol $\mathcal{M}_q$ of the differential equation $\mathcal{R}_q$ is involutive, then we obtain (in a given local representation) all independent equations of order $q+1$ of the prolongation $\mathcal{R}_{q+1}$ by differentiating each equation of $\mathcal{R}_q$ with respect to its multiplicative variables only.*

Proof. Obviously we need only linear operations to transform a symbol into the above described solved form. Thus we can assume without loss of generality, that our local representation of $\mathcal{R}_q$ is such that we obtain the symbol directly in solved form by applying (2.7). Now it is easy to see that all equations obtained by prolonging each equation with respect to its multiplicative variables only are independent, as they all have distinct pivots. Since there are $\beta_q^{(k)}$ equations of class $k$ in $\mathcal{R}_q$, this yields $\sum k\beta_q^{(k)}$ independent equations of order $q+1$ in the prolongation $\mathcal{R}_{q+1}$. Thus we know a lower bound for the number of such equations. By assumption $\operatorname{rank} \mathcal{M}_{q+1}$ equals this number. Hence there cannot be more independent equations of order $q+1$. Note, however, that there may very well exist further independent equations in $\mathcal{R}_{q+1}$, but they must be of lower order, i.e. they yield integrability conditions (cf. the proof of Theorem 2.7). •

This approach to involution via analyzing pivots is inspired by the Janet-Riquier Theory (cf. Appendix B). In Chapter 3 we will see that involution of a symbol means that we can calculate the ranks of its prolongations $\mathcal{M}_{q+r}$ directly from the numbers $\beta_q^{(k)}$ without having to compute explicitly their equations. This will allow statements about the arbitrariness of the general solution.

**Example 2.10** A simple example of a differential equation with a non-involutive symbol is given by

$$\mathcal{R}_2 : \begin{cases} u_{xx} = 0 , \\ u_{yy} = 0 . \end{cases} \tag{2.20}$$

It does not matter, whether we choose $x$ or $y$ as $x^2$, in each case we have one equation of class 2 and one of class 1, thus 3 multiplicative variables. However, the rank of $\mathcal{M}_3$ is 4, as the prolongation contains 4 independent equations. Thus $\mathcal{M}_2$ is not involutive but $\mathcal{M}_3$. It has 5 multiplicative variables and the rank of $\mathcal{M}_4$ is indeed 5. As (2.10) generates obviously no integrability conditions, we have here an example of a formally integrable equation which is not involutive. More generally, one can show (cf. Section 5.4) that the equation

$$\mathcal{R}_q : \begin{cases} u_{[0,q]} = 0 , \\ u_{[q,0]} = 0 \end{cases} \tag{2.21}$$

is formally integrable for any $q$ but becomes involutive only after $q-1$ prolongations.

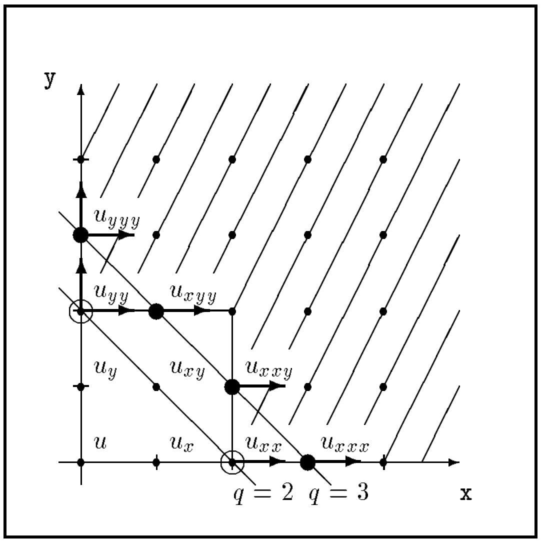


Figure 2.1: Reid diagram for (2.20).

Reid [124] has developed a nice graphical representation of principal and parametric derivatives which can also be used to explain the meaning of involution. Fig. 2.1 shows the Reid diagram for (2.20).

The axes represent the independent variables, the dots the different derivatives of $u$. The circles around $u_{xx}$ and $u_{yy}$ indicate that they are pivots in the symbol. All derivatives in the dashed area can be obtained from them by differentiating the equations, i.e. they are principal derivatives. All other derivatives are parametric ones. The full circles for the third-order derivatives depict the pivots of the prolonged symbol $\mathcal{M}_3$. The arrows indicate the multiplicative variables taking $x^1 = x$, $x^2 = y$.

We see that it is not possible to get all derivatives in the shaded area, if we prolong $\mathcal{M}_2$ only with respect to its multiplicative variables. We miss $u_{xxy}$. $\mathcal{M}_3$ is involutive and indeed every principal derivative of order 4 or higher can be obtained by differentiating one of its equations with respect to its multiplicative variables.

In order to study further properties of involutive symbols, we need a systematic way to order derivatives. In the definition of the solved form of a symbol we used the class, but this provides only a partial ordering. We introduce therefore the concept of a ranking.

**Definition 2.11** *A* ranking *of derivatives is a total ordering defined on the space of all derivatives $p^\alpha_\mu$ for fixed values of $m, n$ which satisfies two conditions:*

*(i) If $|\mu| < |\nu|$, then $p^\alpha_\mu < p^\alpha_\nu$.*

*(ii) If $p^\alpha_{\mu_1} < p^\alpha_{\mu_2}$, then $p^\alpha_{\mu_1+\nu} < p^\alpha_{\mu_2+\nu}$ for any multi-index $\nu$.*

If the first property holds not only for derivatives of the same dependent variable $u^\alpha$ but for all, then the ranking is called a *total-degree* or *orderly* ranking. In such a ranking derivatives are always ordered first by their order and then by other criteria. A total-degree ranking *respects classes*, if $p^\alpha_\mu < p^\beta_\nu$ for $\operatorname{class} p^\alpha_\mu < \operatorname{class} p^\beta_\nu$.

A ranking of derivatives induces of course also a ranking of the unknowns $v^\alpha_\mu$ used as place holders in the symbol. From now on, we will always assume that the columns of the symbol are ordered using a ranking respecting the classes, i.e. if one derivative has a higher class than another one, then it is also higher in the ranking.

An important example for this provides the *inverse lexicographic ranking*. It is a total-degree ranking. If two derivatives have the same order, then we look at the first differing entry in the multi-index. The one with the higher entry is taken as higher in the ranking (hence inverse). Thus it respects classes. Of two derivatives with the same multi-index we take the one as higher which belongs to the dependent variable with the higher index.

Using rankings one can make useful statements about the pivots in the solved form of involutive symbols [117]. They allow one to construct to given values of $m, n, q$ all possible sets of pivots with a given number of equations. Furthermore we will derive several very important properties of involutive symbols (and equations) as corollaries of the following rather technical

**Proposition 2.12** *Let $\mathcal{M}_q$ be an involutive symbol in solved form, where the columns have been ordered according to a class-respecting ranking.*

*(i) If $v^\alpha_\mu$ is a pivot of class $i$, then $v^\alpha_{\mu-1_i+1_j}$ is also a pivot for all $j > i$.*

*(ii) If there are pivots $v^\alpha_\mu$ of class $i$, then the entries $\mu_i$ of their multi-indices take all values between 1 and a maximal value.*

Proof. We start with Part (i). Since $v^\alpha_\mu$ is a pivot of class $i$ and $j > i$, $x^j$ is a non-multiplicative variable for the corresponding equation. We prolong it with respect to $x^j$. Because of the second property of a ranking, $v^\alpha_{\mu+1_j}$ is the pivot of the prolonged equation. For an involutive symbol one obtains, however, all independent equations of the prolonged symbol by prolonging with respect to the multiplicative variables only (Proposition 2.9). Since $v^\alpha_{\mu+1_j}$ is still of class $i$, it must be possible to get it by prolonging a pivot $v^\alpha_\nu$ of higher or equal class with respect to $x^i$. Thus $\nu + 1_i = \mu + 1_j$ or $\nu = \mu - 1_i + 1_j$.

Part (ii) follows easily from Part (i). Let $v^\alpha_\mu$ be the pivot of class $i$ with minimal value of $\mu_i$. But according to (i) $v^\alpha_{\mu-1_i+1_j}$ is also a pivot for $j > i$ and for $\mu_i > 1$ it is also of class $i$. This contradicts the assumption that $\mu_i$ was minimal. Thus $\mu_i = 1$ is the minimal value and one sees at once that with $\mu_i = a$ also $\mu_i = a - 1$ occurs. •

**Corollary 2.13** *Let $\mathcal{M}_1$ be the involutive symbol of a first-order equation. Then*

$$0 \leq \beta_1^{(1)} \leq \beta_1^{(2)} \leq \cdots \leq \beta_1^{(n)} \leq m\,. \tag{2.22}$$

Proof. This is a direct consequence of Part (i) of Proposition 2.12. Let $v^\alpha_\mu$ be a pivot of class $k$, i.e. $\mu_i = \delta_{ik}$. Then $v^\alpha_\nu$ with $\nu = \mu - 1_k + 1_{k+1}$, i.e. $\nu_i = \delta_{i,k+1}$, is a pivot of class $k + 1$ occurring in the symbol $\mathcal{M}_1$. Thus $\beta_1^{(k+1)} \geq \beta_1^{(k)}$. $\beta_q^{(n)}$ can never be greater than $m$ for any value of $q$, as there exist only $m$ different derivatives of class $n$. •

**Theorem 2.14** *Let $\mathcal{M}_q$ be the involutive symbol of the differential equation $\mathcal{R}_q$. Then*

*(i) $\mathcal{M}_{q+1}$ is involutive, too.*

*(ii) $(\mathcal{R}_q^{(1)})_{+1} = \mathcal{R}_{q+1}^{(1)}$.*

Proof. We start with Part (i); Part (ii) will then be a corollary. Without loss of generality we can assume that $\mathcal{M}_q$ is in solved form with the columns ordered according to a class-respecting ranking. Let $v^\alpha_\mu$ be the pivots of $\mathcal{M}_q$. Then by Proposition 2.9 the pivots of $\mathcal{M}_{q+1}$ are $v^\alpha_{\mu+1_\ell}$ where $\ell = 1, \ldots, \operatorname{class}\mu$. The class of such a pivot is $\ell$.

To prove Part (i), we must show that prolonging an equation in $\mathcal{M}_{q+1}$ with respect to a non-multiplicative variable leads to a linearly dependent equation. Thus let $k = \operatorname{class}\mu$ and let $\ell \leq k$. Prolonging the equations of $\mathcal{M}_{q+1}$ with respect to all independent variables leads to pivots of the form $v^\alpha_{\mu+1_\ell+1_j}$ with $j = 1, \ldots, n$. Three cases must be distinguished:

$j \leq \ell$: These are the prolongations with respect to multiplicative variables. Obviously all these equations are linearly independent.

$\ell < j \leq k$: Since $j \leq k$ $x^j$ is a multiplicative variable for the equation in $\mathcal{M}_q$ with the pivot $v^\alpha_\mu$. Thus we can first prolong this equation with respect to $x^j$ and then the result with respect to $x^\ell$. Thus each prolongation is only with respect to multiplicative variables and the resulting equation is already contained in the equations of the previous case.

$j > k$: Let us denote the equation by $\phi = 0$. As in the previous case we interchange the prolongations, thus prolong first with respect to $x^j$ and then with respect to $x^\ell$ to obtain $\phi$. Since $j > k$, $v^\alpha_{\mu-1_k+1_j}$ is by Proposition 2.12 also a pivot in $\mathcal{M}_q$. Its class is greater than or equal $k$; thus $x^k$ is a multiplicative variable for the corresponding equation. By prolonging it with respect to $x^k$ we see that $v^\alpha_{\mu+1_j}$ is among the pivots of $\mathcal{M}_{q+1}$. Since $x^\ell$ is multiplicative for this equation, we have found an equation contained in the first case with pivot $v^\alpha_{\mu+1_\ell+1_j}$.

If we compare the two equations with pivot $v^\alpha_{\mu+1_\ell+1_j}$, then we see that they are both prolongations with respect to $x^\ell$ of equations with the pivot $v^\alpha_{\mu+1_j}$. One of these was obtained by prolonging an equation of $\mathcal{M}_q$ with respect to a non-multiplicative variable. Since $\mathcal{M}_q$ is involutive, it is not contained in the generating set of independent equations we use for $\mathcal{M}_{q+1}$ but is a linear combination of those. Thus $\phi$ can be written as $x^\ell$-prolongation of this linear combination. If $x^\ell$ is a non-multiplicative variable for some of the equations involved, then we can apply the same analysis again. Since the considered pivots will become smaller and smaller, this process will eventually terminate and then we will have expressed $\phi$ as a linear combination of equations contained in the first case.

Thus we have shown that the only independent equations in $\mathcal{M}_{q+2}$ can be obtained by prolonging the equations of $\mathcal{M}_{q+1}$ with respect to their multiplicative variables only. But this means nothing else than that $\mathcal{M}_{q+1}$ is involutive.

Part (ii) may look trivial at first sight, but it is not. At the left hand side only the integrability conditions which arise during the projection from order $q+1$ to order $q$ are taken into account; then one prolongs again to order $q+1$. On the right hand side, however, we project from order $q+2$ to order $q+1$. Hence we have to show that all integrability conditions which occur during this projection are prolongations of integrability conditions of the projection of order $q+1$ to $q$.

In order to achieve this we make basically the same considerations as in the proof of Part (i) but now with the full equations instead of only with their highest-order part. Let $\Phi^\tau$ be an equation of class $k$ in a local description of $\mathcal{R}_q$. We have to analyses all equations of the form $D_\ell D_j \Phi^\tau$. We distinguish two cases assuming always $\ell \leq j$:

$j \leq k$: In this case $D_\ell D_j \Phi^\tau$ contains only prolongations with respect to multiplicative variables. As we have seen in Proposition 2.9, these are the only independent equations of order $q+2$ in our local description of $\mathcal{R}_{q+2}$.

$k < j$: This implies that $D_j \Phi^\tau$ was a prolongation with respect to a non-multiplicative variable. By Proposition 2.9 we can express this equation as a linear combination of prolongations of equations of $\mathcal{R}_q$ with respect to multiplicative variables and perhaps some integrability conditions. Following again the argument of the third case above, we find that $D_\ell D_j \Phi^\tau$ is a linear combination of the equations contained in the first case and of prolonged integrability conditions.

Note that it is not necessary here to distinguish between $\ell \leq k$ and $\ell > k$, as we are allowed to use all prolongations to construct $(\mathcal{R}_q^{(1)})_{+1}$. •

Finally we prove a generalization of an old result by Finzi [53, 111] for systems with the same number of equations and dependent variables. It emphasizes the distinguished position of normal equations (cf. Section 2.6).

**Proposition 2.15** *A differential equation $\mathcal{R}_q$ has either identities or integrability conditions, if and only if $\beta_q^{(n-1)} > 0$.*

PROOF. If $\beta_q^{(n-1)} > 0$, there exist equations with non-multiplicative variables, as the only equations without are those of class $n$. Thus according to Proposition 2.9 the prolongation of these equations with respect to $x^n$ will lead either to identities or to integrability conditions. For the converse, we note that if $\beta_q^{(n-1)} = 0$ all $\beta_q^{(k)}$ with $k < n-1$ vanish, too. This can be shown similarly to Corollary 2.13. Thus $\mathcal{R}_q$ contains only equations of class $n$ and all prolonged equations are independent. •

## 2.4 Cartan-Kuranishi Theorem

Theorem 2.14 has a very important application. It yields an easy criterion for an involutive equation. Recall that Definition 2.5 of a formally integrable equation requires to check infinitely many conditions, namely that at no order of prolongation integrability conditions arise. If we have, however, an involutive symbol, this can be reduced to just one prolongation.

**Theorem 2.16 (Criterion of Involution)** *$\mathcal{R}_q$ is an involutive equation, if and only if its symbol $\mathcal{M}_q$ is involutive and $\mathcal{R}_q^{(1)} = \mathcal{R}_q$.*

PROOF. Since $\mathcal{M}_q$ is involutive, $\mathcal{R}_{q+1}^{(1)} = (\mathcal{R}_q^{(1)})_{+1} = \mathcal{R}_{q+1}$ according to Theorem 2.14 and our assumptions. But all prolonged symbols $\mathcal{M}_{q+r}$ are also involutive and we can apply this argument at any order. Thus $\mathcal{R}_{q+r}^{(1)} = \mathcal{R}_{q+r}$ for all $r \geq 1$ and $\mathcal{R}_q$ is formally integrable. •

Based on this theorem we will now show that any differential equation $\mathcal{R}_q$ can be completed to an equivalent equation, i.e. one with the same solution space, which is involutive. Thus in the sequel we can always assume that we deal with involutive equations. We will later see that this has many advantages, as involution entails many special properties.

**Theorem 2.17 (Cartan-Kuranishi)** [3] *For every differential equation $\mathcal{R}_q$ there exist two integers $r, s$ such that $\mathcal{R}_{q+r}^{(s)}$ is an involutive equation with the same solution space.*

PROOF. We refer for a detailed proof of the theorem to [117]. We here outline a proof by giving a concrete algorithm for the completion and showing its correctness and termination. It consists of two nested loops. The inner one prolongs the equation, until its symbol is involutive. The outer one performs the projections; i.e. it computes the arising integrability conditions. In a more formal language, we get the algorithm shown in Fig. 2.2.

To prove the correctness and termination of this algorithm, we need the following key theorem of the formal theory of partial differential equations which we cite without proof.

[3] The name *Cartan-Kuranishi Theorem* stems from a similar theorem in Cartan-Kähler Theory [92].

**Algorithm** `Completion` $(\mathcal{R}_q)$

Input: Equation $\mathcal{R}_q$
Output: Equation $\mathcal{R}^{(s)}_{q+r}$ in involution

```
[1]      r ← 0; s ← 0
[2]      until R^(s)_{q+r} involutive repeat
[2.1]        compute M^(s)_{q+r}                    {extract symbol}
[2.2]        while not(M^(s)_{q+r} involutive) repeat
[2.2.1]          r ← r+1
[2.2.2]          compute M^(s)_{q+r}                {prolong symbol}
[2.3]        compute R^(s)_{q+r}                    {prolong equation}
[2.4]        if R^(s+1)_{q+r} ⊂ R^(s)_{q+r} then
[2.4.1]          s ← s+1
[2.4.2]          compute R^(s)_{q+r}                {project equation}
[3]      return R^(s)_{q+r}
```

Figure 2.2: Algorithm for the Cartan-Kuranishi Theorem (sketchy version).

**Theorem 2.18 (Prolongation Theorem)** *Let $\mathcal{M}_q$ be the symbol of an equation $\mathcal{R}_q$. Then there exists an integer $\hat{q}(n,m,q) \geq q$ such that $\mathcal{M}_{\hat{q}}$ is involutive.*

The formal proof of this theorem is highly nontrivial and was first given by Sweeny [150]. It makes decisive use of the Spencer cohomology and also yields a bound for $\hat{q}(n,m,q)$. In our algorithm, it guarantees the termination of the inner loop in step [2.2].

An important point for the correctness of the algorithm is the question, whether it really results in an equation of the form $\mathcal{R}^{(s)}_{q+r}$. Due to the structure of the loops, it will generate equations of the form $(((((\mathcal{R}_{q+r_1})^{(1)})_{+r_2})^{(1)})\cdots)_{+r_s}$. Since the proposed algorithm first prolongs to an involutive symbol and then projects, we can always apply Theorem 2.14 and get indeed an equation of the predicted form.

The check whether a symbol or an equation is involutive makes no difficulties with our criterion (2.8) and Theorem 2.16. To decide whether or not $\mathcal{R}_q$ and $\mathcal{R}^{(1)}_q$ are equal, it suffices to compare their dimensions, as the second is a submanifold of the first. The dimension of $\mathcal{R}^{(1)}_q$ can be computed indirectly using Theorem 2.7. The proof of that theorem indicated clearly the construction of the integrability conditions.

Finally, we show by a Noetherian argument that the algorithm will terminate. To simplify the notation, we set $m = 1$. The proof runs along the same lines for higher $m$. Since we know already from Theorem 2.18 that the inner loop terminates, it suffices to consider the equations $\mathcal{R}^{(1)}_{q+r_1}, \mathcal{R}^{(2)}_{q+r_2}, \ldots$ obtained after each repetition of the outer loop and the corresponding symbols $\mathcal{M}^{(1)}_{q+r_1}, \mathcal{M}^{(2)}_{q+r_2}, \ldots$ We will associate with each an ideal in the polynomial ring $\mathbb{R}[\chi_1,\ldots,\chi_n]$.

Let the symbol $\mathcal{M}_q$ of the original equation be described by the system $A^{\tau\mu}(P)v_\mu = 0$ for an arbitrary point $P \in \mathcal{R}_q$. This defines a set of polynomials through the substitution

$$v_\mu = v_{[\mu_1\cdots\mu_n]} \mapsto \chi_\mu = \chi_1^{\mu_1}\cdots\chi_n^{\mu_n}\,. \tag{2.23}$$

Let $I_0$ denote the ideal generated by these polynomials.

Let the basis of the ideal $I_i$ be given by the basis of $I_{i-1}$ plus all polynomials obtained by applying (2.23) to the equations of the symbol $\mathcal{M}^{(i)}_{q+r_i}$. This means essentially that we add one new polynomial for each new integrability condition, since differentiation with respect to $x^i$ corresponds now to multiplication with $\chi_i$. Thus we obtain a new independent generator in the ideal basis only, if an integrability condition has occurred.

This procedure yields an ascending chain of ideals $I_0 \subseteq I_1 \subseteq \cdots \subseteq I_s \subseteq \cdots$. It is well-known from algebra that such a chain must become stationary. Consequently there exists an $s$ after which no more integrability conditions have to be added. But then the equation is involutive and the algorithm terminates.

To finish the proof of the Cartan-Kuranishi Theorem, we note that every solution of $\mathcal{R}_q$ is also a solution of $\mathcal{R}^{(s)}_{q+r}$, as by the definition of an integrability condition every solution of the original equation also satisfies all integrability conditions. Prolongations have of course no effect on the solution space. (Recall that we consider only smooth solutions!) •

With the help of these considerations, we can design a more detailed algorithm using essentially only linear algebra and partial derivatives. It is shown in Fig. 2.3. There remains essentially only one *algebraic* problem, namely determining the dimensions of the submanifolds $\mathcal{R}^{(s)}_{q+r}$.

**Algorithm** `Completion`$(\mathcal{R}_q)$

Input: Equation $\mathcal{R}_q$
Output: Equation $\mathcal{R}^{(s)}_{q+r}$ in involution

| | | |
|---|---|---|
| [1] | r $\leftarrow$ 0; s$\leftarrow$ 0 | |
| [2] | compute $\mathcal{R}_{q+1}$ | {*prolong*} |
| [3] | compute $\mathcal{M}_q, \mathcal{M}_{q+1}$ | {*extract symbols*} |
| [4] | `until` $\mathcal{R}^{(s)}_{q+r}$ involutive `repeat` | |
| [4.1] | `while` $\#multVar(\mathcal{M}^{(s)}_{q+r}) \neq \operatorname{rank} \mathcal{M}^{(s)}_{q+r+1}$ `repeat` | |
| [4.1.1] | r $\leftarrow$ r+1 | {*counter for prolongations*} |
| [4.1.2] | compute $\mathcal{R}^{(s)}_{q+r+1}$ | {*prolong*} |
| [4.1.3] | compute $\mathcal{M}^{(s)}_{q+r+1}$ | {*extract symbol*} |
| [4.2] | `if` $\dim \mathcal{R}^{(s)}_{q+r+1} - \dim \mathcal{M}^{(s)}_{q+r+1} < \dim \mathcal{R}^{(s)}_{q+r}$ `then` | |
| [4.2.1] | s $\leftarrow$ s+1 | {*counter for projections*} |
| [4.2.2] | compute $\mathcal{R}^{(s)}_{q+r}$ | {*add integrability conditions*} |
| [4.2.3] | compute $\mathcal{R}^{(s)}_{q+r+1}$ | {*prolong*} |
| [4.2.4] | compute $\mathcal{M}^{(s)}_{q+r}, \mathcal{M}^{(s)}_{q+r+1}$ | {*extract symbols*} |
| [5] | `return` $\mathcal{R}^{(s)}_{q+r}$ | |

Figure 2.3: Algorithm for the Cartan-Kuranishi Theorem (detailed version).

The Cartan-Kuranishi Theorem also shows that it is no restriction to assume from now on that we deal exclusively with involutive equations, as we possess an algorithm to complete every differential equation in a finite number of steps to an involutive one.

**Example 2.19** We demonstrate this algorithm at a standard example in formal theory [120] originally given by Janet. It comprises two linear equations of second order in one dependent and three independent variables:

$$\mathcal{R}_2 : \begin{cases} u_{zz} + y\, u_{xx} &= 0\,, \\ u_{yy} &= 0\,. \end{cases} \tag{2.24}$$

We take the ordering $x^3 = z$, $x^2 = y$, $x^1 = x$ for the definition of the classes. The algorithm in Fig. 2.3 tells us to prolong first until the symbol becomes involutive. $\mathcal{M}_2$ comprises one equation of class 3 and one of class 2; this means there are 5 multiplicative variables but $\operatorname{rank}\mathcal{M}_3 = 6$. Hence according to Definition 2.8 $\mathcal{M}_2$ is not involutive. Since $\mathcal{M}_3$ has 11 multiplicative variables and $\operatorname{rank}\mathcal{M}_4 = 11$, the inner loop terminates after one iteration with the involutive symbol $\mathcal{M}_3$.

We must check whether there are integrability conditions: $\dim\mathcal{R}_4 = 15$ and $\dim\mathcal{M}_4 = 4$, thus with Theorem 2.12 $\dim\mathcal{R}_3^{(1)} = 11 < \dim\mathcal{R}_3 = 12$. This implies the existence of one integrability condition. The construction outlined in Section 2.2 leads to the equation

$$\partial_{yy}(u_{zz} + y\,u_{xx}) - \partial_{zz}u_{yy} - y\partial_{xx}u_{yy} = u_{xxy} = 0\,. \tag{2.25}$$

After the first iteration of the outer loop we have arrived at the equation $\mathcal{R}_3^{(1)}$. We start the same procedure again. Since $\mathcal{M}_3^{(1)}$ is not involutive, we must prolong once to get the involutive symbol $\mathcal{M}_4^{(1)}$. But again an integrability condition occurs, namely

$$\partial_{xxy}(u_{zz} + y\,u_{xx}) - y\partial_{xx}u_{xxy} - \partial_{zz}u_{xxy} = u_{xxxx} = 0\,. \tag{2.26}$$

This completes the second iteration of the outer loop and we must start anew with $\mathcal{R}_4^{(2)}$.

But we detect again a non-involutive symbol. Its prolongation $\mathcal{M}_5^{(2)}$ vanishes ($\dim\mathcal{M}_5^{(2)} = 0$) and is therefore trivially involutive. Since $\dim\mathcal{R}_5^{(3)} = 12 = \dim\mathcal{R}_5^{(2)}$, no integrability conditions occur and we have finished. Because of the vanishing symbol, $\mathcal{R}_5^{(2)}$ is a finite type equation (see Section 3) and has a solution space of dimension $12 = \dim\mathcal{R}_5^{(2)}$. $\mathcal{R}_4^{(2)}$ provides a further example of an equation which is formally integrable, but not involutive.

Note that in this example a direct integration would have been easily possible. It yields a simple polynomial solution depending on 12 arbitrary constants. If we perform, however, a complicated non-linear change of coordinates in (2.19), then we will probably have problems with finding the explicit solution. In contrast the here outlined computation will hardly change and still be readily performed.

As mentioned above, the proof of Theorem 2.18 as given by Sweeny yields an upper bound for the number of prolongations needed to render a symbol $\mathcal{M}_q$ involutive. This bound depends only on the order $q$, the number of independent variables $n$ and the number of dependent variables $m$. It can be expressed in form of a recursion relation [117]:

$$\begin{aligned}
\hat{q}(n,m,q) &= \hat{q}\left(n, m\binom{q+n-1}{n}, 1\right),\\
\hat{q}(n,m,1) &= m\binom{\hat{q}(n-1,m,1)+n}{n-1} + \hat{q}(n-1,m,1) + 1\,,\\
\hat{q}(0,m,1) &= 0\,.
\end{aligned} \tag{2.27}$$

Table 2.4 contains $\hat{q}(n,m,1)$ for different values of $m$ and $n$. One sees that the values explode with growing $n$ and $m$. The situation is still worse for higher order equations. It is not known, whether this bound is sharp, i.e. whether or not there are symbols which need so many prolongations to become involutive. But fortunately, in applications $\hat{q}(n,m,q)$ yields usually a coarse over-estimate of the actually needed number of prolongations.

The "worst" symbol we will encounter occurred in Example 2.10. The symbol of (2.21) becomes involutive after $q-1$ prolongations. Evaluation of (2.27) yields however

$$\hat{q}(2,1,q) = \frac{1}{4}q^4 + \frac{1}{2}q^3 + \frac{9}{4}q^2 + 2q + 2\,. \tag{2.28}$$

This yields e.g. $\hat{q}(2,1,1) = 7$ and $\hat{q}(2,1,2) = 23$.

| $n\backslash m$ | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| 2 | 7 | 14 | 23 | 34 |
| 3 | 53 | 287 | 999 | 2.699 |
| 4 | 29.314 | 8.129.858 | 503.006.503 | 13.151.182.504 |

Figure 2.4: $\hat{q}(n, m, 1)$ for different values of $m$ and $n$.

## 2.5 Power Series Solutions

Definition 2.1 reminds of algebraic geometry, where one associates an algebraic variety with an ideal of polynomials, namely the set of common zeros of all elements of the ideal. There is, however, one big difference between this variety and the submanifolds $\mathcal{R}_q$ considered in the geometric theory of partial differential equations: points in $\mathcal{R}_q$ are no solutions! We will see in this section, how it is possible to obtain something similar for differential equations by studying power series solutions. We will restrict throughout this section our attention to analytic equations.

In order to construct a formal power series solution we need all prolongations of the given differential equation. Thus we need a jet bundle of infinite order. Such a bundle can be introduced using the notions of projective and inductive limits in directed systems of sets [94]. These limits allow also to "transport" the topology and differential structure of the finite dimensional jet bundles. We define

$$J_\infty\mathcal{E} = \operatorname{proj}\lim_{q\to\infty} J_q\mathcal{E}\,, \tag{2.29}$$

for the jet bundles of different order are related by the natural projections $\pi^{q+r}_q$. Local analytic functions on $J_\infty\mathcal{E}$ can be introduced by

$$\mathcal{F}^\omega = \operatorname{ind}\lim_{q\to\infty} \mathcal{C}^\omega(J_q\mathcal{E})\,, \tag{2.30}$$

as the spaces $\mathcal{C}^\omega(J_q\mathcal{E})$ are naturally related by inclusion maps. Note that the definition of an inductive limit implies that every element of $\mathcal{F}^\omega$ is also element of at least one $\mathcal{C}^\omega(J_q\mathcal{E})$, i.e. although these functions are defined on an infinite-dimensional manifold, each of them depends only on a finite number of arguments. In a similar manner, although slightly more complicated, one can construct the tangent and co-tangent bundle of $J_\infty\mathcal{E}$ [11], but this is not necessary for our purposes.

Let us introduce the following ideals of local analytic functions vanishing on the differential equation $\mathcal{R}_q$ and its prolongations:

$$I_{q+r} = \left\{ f \in \mathcal{C}^\omega(J_{q+r}\mathcal{E}) \mid f\big|_{\mathcal{R}_{q+r}} \equiv 0 \right\}. \tag{2.31}$$

Then we can define

$$I_\infty \subset \mathcal{F}^\omega = \operatorname{ind}\lim_{r\to\infty} I_{q+r}\,. \tag{2.32}$$

and consequently the infinite prolongation $\mathcal{R}_\infty = V(I_\infty) \subset J_\infty\mathcal{E}$ as the set of common zeros of all functions in $I_\infty$. For an involutive differential equation we could alternative use that since $\mathcal{R}^{(s)}_{q+r} = \mathcal{R}_{q+r+s}$ for all $r, s$, we have natural projections $\pi^k_l : \mathcal{R}_k \to \mathcal{R}_l$ for $k > l$. Thus we can directly write

$$\mathcal{R}_\infty = \operatorname{proj}\lim_{q\to\infty} \mathcal{R}_q \,. \tag{2.33}$$

$\mathcal{R}_\infty$ is also called the space of formal solutions of $\mathcal{R}_q$ or a *diffiety* [89] as short form for differential variety. It can be considered as differential analogue to an algebraic variety. The study of its properties is the subject of the so-called *formal geometry* [89, 157]. We will not pursue it further; we only need $I_\infty$ to define general formal power series solutions.

The generic form of a formal power series around an arbitrary but fixed point $x_0 \in X$ is given by

$$u^\alpha(x) = \sum_{|\mu|=0}^{\infty} \frac{a^\alpha_\mu}{\mu!}(x - x_0)^\mu \,, \tag{2.34}$$

with the standard notations: $\mu! = \mu_1!\,\mu_2!\cdots\mu_n!$ and $(x-x_0)^\mu = (x^1-x^1_0)^{\mu_1}\cdots(x^n-x^n_0)^{\mu_n}$. To simplify the notations, we will regard from now on the dependent variables $u^\alpha$ as derivatives of order zero, i.e. we will write them also in the form $p^\alpha_\mu$ with $|\mu| = 0$.

Plugging this ansatz into a local representation of $\mathcal{R}_q$ yields algebraic equations for the coefficients $a^\alpha_\mu$. These equations contain still the independent variables $x^i$. If (2.34) is supposed to be a solution, they must be satisfied for all values of $x$ in a neighborhood of $x_0$. Instead of expanding we do the same with all prolongations $\mathcal{R}_{q+r}$ and evaluate each system at $x = x_0$. Then we can identify the jet coordinates $p^\alpha_\mu$ with the Taylor coefficients $a^\alpha_\mu$ and each prolongation leads to one set of equations:

$$\begin{array}{rll} \mathcal{R}_q : & \Phi^\tau(x_0, a^\alpha_\mu) = 0\,, & 0 \le |\mu| \le q\,, \\ \mathcal{R}_{q+1} : & (D_i\Phi^\tau)(x_0, a^\alpha_\mu) = 0\,, & 0 \le |\mu| \le q+1\,, \\ \mathcal{R}_{q+r} : & (D_\nu\Phi^\tau)(x_0, a^\alpha_\mu) = 0\,, & 0 \le |\mu| \le q+r\,, |\nu| = r\,. \end{array} \tag{2.35}$$

The union of the left hand sides of all these equations yields a basis of the ideal $I_\infty(x_0) \subset I_\infty$ of functions defined in a neighborhood of $x_0$. Any solution $a^\alpha_\mu$ of these infinitely many algebraic equations yields a formal power series solution of the differential equation $\mathcal{R}_q$.

**Definition 2.20** *A* general formal power series solution *of the differential equation $\mathcal{R}_q$ in a neighborhood of $x_0 \in X$ consists of a sequence of manifolds $W_0 \subset W_1 \subset W_2 \subset \cdots$ (not necessarily distinct) and mappings*

$$a^\alpha_\mu : W_q \to \mathbb{R}\,, \qquad \begin{cases} \alpha &= 1,\dots,m\,, \\ |\mu| &= q\,, \end{cases} \tag{2.36}$$

*such that*

$$f \in I_\infty(x_0) \iff f\Big(x_0, a^\alpha_\mu(W_q)\Big) \equiv 0 \;\; \text{(where defined)}. \tag{2.37}$$

The manifolds $W_q$ in this definition represent the arbitrary parameters of the general solution. Note that it is decisive in (2.37) that there is an equivalence between the two statements. One direction ensures that (2.34) yields a formal solution for any value of the parameters. The other one ensures that we have indeed a general solution, i.e. no other differential equation than those contained in $I_\infty(x_0)$ are solved by (2.34). (More on defining general solutions can be found in Section 3.1).

Obviously there are many ways to write a general formal power series solution. The most important ones for us are those, where we take a basis of $I_\infty(x_0)$ in which all functions are of the form

$$p^\alpha_\mu = \varphi^\alpha_\mu(x^i, u^\beta, p^\gamma_\nu) \tag{2.38}$$

with different left hand sides. Such a basis entails a natural distinction into principal and parametric derivatives: principal are all derivatives appearing on the left hand side; all others are parametric. The mappings $a^\alpha_\mu$ take now a very simple form. The parametric coefficients are identified with the arbitrary parameters; the principal ones are obtained by using the functions $\varphi^\alpha_\mu$.

**Example 2.21** As a simple example we consider the Heat Equation $u_{xx} = u_t$. Taking $x^2 = x$ and $x^1 = t$ we can also write it as $p_{[0,2]} = p_{[1,0]}$. If we choose $p_{[1,0]}$ as the principal derivative, then the parametric ones are all derivatives of the form $p_{[0,q]}$ (pure $x$-derivatives). We obtain the following general formal power series solution: We set $W_q = \mathbb{R}^{2q+1}$ and denote the coordinates in these spaces with $w_i$, $i = 1, \dots$ Then we take the mappings $a_{[r,s]} = w_{2r+s+1}$. It is straightforward to check that this is a general formal power series solution of the Heat Equation, since $a_{[r,s]} = a[0, 2r+s]$. Note that this provides indeed only a formal solution, as the resulting series will not converge for arbitrary values of the parameters $w_i$.

Our main interest will lie in formal power series solution which are even more specialized. We say that a power series can be constructed *order by order*, if in the basis (2.38) all functions $\varphi^\alpha_\mu$ depend only on derivatives $p^\gamma_\nu$ with $|\nu| \leq |\mu|$, i.e. the principal derivatives of order $q$ depend only on parametric derivatives of the same or lower order. The power series solution constructed for the Heat Equation was not of this type. We expressed the principal coefficient $a_{[1,0]}$ through the parametric one $a_{[0,2]}$, i.e. one of higher order.

We will describe now a general procedure to construct general formal power series solutions for involutive equations order by order. We know from Section 2.2 that the formal derivative is linear in the highest-order derivatives. Thus we can rewrite in (2.35) the equations for $r > 0$ as inhomogeneous linear systems, where the right hand sides depend only on coefficients of lower order and the left hand sides are the symbols $\mathcal{M}_{q+r}$. For the coefficients of order $q + r$ we get from (2.11) the following system

$$\sum_{\alpha,|\mu|=q} \left(\frac{\partial \Phi^\tau}{\partial p^\alpha_\mu}\right)(x_0, a^\beta_\lambda)\, a^\alpha_{\mu+\nu} = B^\tau_\nu(x_0, a^\gamma_\rho)\,, \qquad \begin{cases} |\lambda|\leq q\,, \\ |\nu|=r\,, \\ |\rho|<q+r\,. \end{cases} \tag{2.39}$$

The $B^\tau_\nu$ contain the remaining terms of the formal derivative.

In general, system (2.39) will be under-determined and there will be different possibilities to choose the coefficients $a^\alpha_{\mu+\nu}$ for which the system will be solved. These are the principal coefficients. All other coefficients of order $q + r$ are parametric. It is important to note that no choice is distinguished and that the choices at the different orders can be made completely independently.

We are thus lead to the "algorithm" in Fig. 2.5. It is valid for any method of choosing the principal derivatives in step [2.2]. The remaining problem is essentially to solve the non-linear equations for the coefficients of order $q$ and lower in step [1], i.e. an algebraic problem. Note that in general the solution space of such an algebraic system may contain several components of different dimensions. According to Definition 2.1, $\mathcal{R}_q$ is, however, a submanifold (this can for instance be achieved by selecting one smooth component). Thus we can solve by the Implicit Function Theorem for $\operatorname{codim} \mathcal{R}_q$ principal coefficients of order $q$ or less.

All formal power series solutions generated with this "algorithm" have one thing in common: At order $q+r$ there are exactly $\dim \mathcal{M}_{q+r}$ parametric derivatives. These numbers will be fairly important in Chapter 3 for measuring the arbitrariness. We will give them therefore a special name.

**Algorithm** `PowerSeriesSolution` $(\mathcal{R}_q, maxOrd)$

Input: Involutive equation $\mathcal{R}_q$; maximal order $maxOrd$
Output: Formal power series solution

| | | |
|---|---|---|
| [1] | Compute the general solution of $\Phi^\tau(x_0, a^\alpha_\mu) = 0$ | {*non-linear equations!*} |
| [2] | **for** $r$ **in** $1..maxOrd$ **repeat** | |
| [2.1] | Set up system (2.39) | |
| [2.2] | Choose principal coefficients of order $q + r$ | |
| [2.3] | Solve for principal coefficients of order $q + r$ | {*linear equations!*} |
| [3] | **return** coefficients | |

Figure 2.5: "Algorithm" to compute power series solution of involutive $\mathcal{R}_q$.

**Definition 2.22** *The* Hilbert polynomial *of an involutive symbol* $\mathcal{M}_q$ *is defined by*

$$H_q(r) = \dim \mathcal{M}_{q+r} \,. \tag{2.40}$$

*If* $\mathcal{R}_q$ *is an involutive differential equation with the symbol* $\mathcal{M}_q$*, then it denotes the number of arbitrary Taylor coefficients in a general formal power series solution of* $\mathcal{R}_q$ *which can be constructed order by order.*

The name Hilbert polynomial expresses a similarity between this analysis of power series solutions and the analysis of the coordinate ring of a monomial ideal $I$ in algebraic geometry as introduced by Hilbert [73]. There the affine Hilbert function $HF_I(s)$ is defined as the number of monomials of total degree less than or equal to $s$ not contained in $I$. A well-known theorem [34] asserts that for $s$ large enough the Hilbert function becomes a polynomial $HP_I(s)$. The smallest integer $s_0$ for which $HF_I(s) = HP_I(s)$ is called the index of regularity. (The restriction to monomial ideals is not severe, as an arbitrary ideal has the same Hilbert function as the ideal spanned by all its leading monomials).

These ideas can be directly related to partial differential equations with the mapping (2.23) used in the proof of the Cartan-Kuranishi Theorem. If we set again for simplicity $m = 1$, then it associated with a derivative $p_\mu$ the monomial $\chi_1^{\mu_1} \cdots \chi_n^{\mu_n}$. The total degree of this monomial is obviously the order of the derivative. If we consider the ideal spanned by the monomials obtained from the principal derivatives, then the parametric derivatives correspond to the monomials not contained in the ideal. At this stage it is not at all clear that $H_q(r)$ is indeed a polynomial. But in Section 3.2 we will give an explicit expression for it, from which it will become evident. Thus we see that the order at which a symbol becomes involutive provides an upper bound for the index of regularity of the associated monomial ideal. More on this relationship can be found in [25].

The here used notation for the Hilbert polynomial leads sometimes to confusion. The index $q$ does not distinguish different polynomials but refers to the order used as basis for the argument $r$. Thus the value $H_q(3)$ gives the number of arbitrary coefficients at order $q + 3$ and not at order 3. This is especially important, if one wants to compare the Hilbert polynomials of two different differential equations. Let us assume that their respective orders are $q_1$ and $q_2$ with $q_2 > q_1$. In order to see which equations allows for more arbitrary coefficients at a given order, one must compare $H_{q_1}(r + q_2 - q_1)$ and $H_{q_2}(r)$.

## 2.6 Characteristics and the Tableau

Recalling the definition (2.8) of the symbol map $\sigma$ we define a further linear map parameterized by a one-form over the base manifold $X$. This new way to look at the symbol will turn out to be quite fruitful. It will bring us not only characteristics but will also allow us to define rigorously under- and over-determinacy, respectively, of a differential equation.

**Definition 2.23** *Let $\chi = \chi_i dx^i \in T^*X$ be an arbitrary one-form over $X$ and let $\mathcal{R}_q$ be a differential equation locally defined by the map $\Phi : J_q\mathcal{E} \to \mathcal{E}'$. The* tableau *$\tau_\chi$ of $\mathcal{R}_q$ is the linear mapping*[4]

$$\tau_\chi : \left\{ \begin{array}{ccccc} V\mathcal{E} & \xrightarrow{\iota_\chi} & S_qT^*X \otimes V\mathcal{E} & \xrightarrow{\sigma} & \mathcal{E}' \\ v^\alpha & \longmapsto & \chi_\mu v^\alpha & \longmapsto & \displaystyle\sum_{\alpha,|\mu|=q} \left(\frac{\partial \Phi^\tau}{\partial p^\alpha_\mu}\right) \chi_\mu v^\alpha \end{array} \right. \tag{2.41}$$

*where $\chi_\mu = (\chi_1)^{\mu_1} \cdots (\chi_n)^{\mu_n}$.*

The tableau is sometimes also called *principal symbol* [123]. Its matrix

$$\mathcal{T}^\tau_\alpha[\chi] = \sum_{|\mu|=q} \left(\frac{\partial \Phi^\tau}{\partial p^\alpha_\mu}\right) \chi_\mu \tag{2.42}$$

has $p$ rows and $m$ columns, where $p$ is the dimension of $\mathcal{E}'$. The entries of this matrix are homogeneous polynomials of degree $q$ in the coefficients of $\chi$.

**Definition 2.24** *We call a non-zero one-form $\chi \in T^*X$* characteristic *in $P \in \mathcal{R}_q$, if the tableau $\tau_\chi$ is not injective at the point $P$. The set of all characteristic one-forms at $P$ is the* characteristic variety *$C_P$.*

$C_P$ is indeed a variety in the strict algebraic meaning of the word. The condition that $\tau_\chi$ is not injective can be expressed as the vanishing of certain subdeterminants of the matrix $\mathcal{T}[\chi]$, i.e. as the vanishing of polynomials in $\chi_i$. Characteristics are most important for semi-linear equations, i.e. equations where the symbol depends only on the independent variables. Then we can define them at points $x_0 \in X$.

**Example 2.25** To see the connection between this definition and the usual one let us consider a linear first-order system

$$\mathcal{R}_1 : \left\{ \sum_{i,\alpha} A^{\tau i}_\alpha(x) p^\alpha_i + \sum_\alpha B^\tau_\alpha(x) u^\alpha + C^\tau(x) = 0\,, \quad \tau = 1,\ldots,p \right. \tag{2.43}$$

and an $(n-1)$-dimensional hypersurface $\Sigma$ in solved form

$$\Sigma : x^n = \phi(x^1,\ldots,x^{n-1})\,. \tag{2.44}$$

Such a surface is usually called characteristic, if prescribing the values of $u^\alpha$ on $\Sigma$ does not suffice to compute all derivatives on the surface.

We assume that $u^\alpha\big|_\Sigma = f^\alpha(x^1,\ldots,x^{n-1})$. This yields using the chain rule

$$\left.\frac{\partial u^\alpha}{\partial x^i}\right|_\Sigma + \frac{\partial \phi}{\partial x^i}\left.\frac{\partial u^\alpha}{\partial x^n}\right|_\Sigma = \frac{\partial f^\alpha}{\partial x^i}\,, \quad i = 1,\ldots,n-1\,. \tag{2.45}$$

[4] This is a kind of "dual" definition to [17] where $\tau_\chi$ is called symbol and its solution space tableau.

Substituting these relations into (2.43) yields a linear system for the derivatives $\partial u^\alpha/\partial x^n|_\Sigma$ with matrix

$$A^{\tau n}_\alpha - \sum_{i=1}^{n-1} A^{\tau i}_\alpha \frac{\partial\phi}{\partial x^i}\,. \tag{2.46}$$

If the rank of this matrix is less than $m$, i.e. the associated linear mapping is not injective, then it is not possible to determine all derivatives on $\Sigma$. But this means nothing else than that the one-form $\chi = dx^n - \sum_{i=1}^{n-1}(\partial\phi/\partial x^i)dx^i$ is characteristic according to Definition 2.24. Hence $\chi = d\Phi$ where $\Phi = x^n - \phi(x^1,\dots,x^{n-1}) = 0$ describes $\Sigma$. If we introduce the co-normal variety $N^*_x\Sigma$ as the set of all one-forms annihilating the tangent space of the surface[5], i.e. $N^*_x\Sigma = \{\omega \in T^*_x X \mid \omega(T_x\Sigma) = 0\}$, then we see that $N^*\Sigma$ is generated by $\chi$.

The tableau $\tau_\chi$ allows us to give a precise definition of under- and over-determined equations. Intuitively we would consider a differential equation as under-determined, if its general solution contains one (or more) arbitrary function of all independent variables $x^1,\dots,x^n$, as this implies that the equation poses no restrictions at all for at least one of the dependent variables $u^\alpha$.

**Definition 2.26** *The involutive equation $\mathcal{R}_q$ is said to be* under-determined *at $P \in \mathcal{R}_q$, if there exists no one-form $\chi$ such that $\tau_\chi$ is injective. If there exists a one-form $\chi$ such that $\tau_\chi$ is bijective, then the equation is* well-determined *or* normal. *Otherwise the equation is called* over-determined.

This definition of an under-determined equation is more precise than the one given by Pommaret [117]. It is a straightforward generalization of the one given by Olver [111] for systems with the same number of equations and dependent variables. We will see later in Chapter 3, that it is indeed equivalent to requiring that the general solution contains arbitrary functions of all independent variables. Olver's definition of over-determined equations is not satisfactory, as he does not require the equation to be in involution. Thus it can happen that an over-determined system becomes under-determined by *adding* an integrability condition!

**Theorem 2.27** *Let $\mathcal{R}_q$ be a differential equation with tableau $\tau_\chi$. In different coordinate systems we might get different values for $\beta^{(n)}_q$. Let $\tilde\beta^{(n)}_q$ be the maximum of these values. Then*

$$\max_\chi\ \operatorname{rank}\tau_\chi = \tilde\beta^{(n)}_q\,. \tag{2.47}$$

Proof. By definition of $\tilde\beta^{(n)}_q$ we can find a local description of $\mathcal{R}_q$ where the first $\tilde\beta^{(n)}_q$ equations $\Phi^\tau(x^i,u^\alpha,p^\alpha_\mu) = 0$ satisfy

$$\frac{\partial\Phi^\tau}{\partial p^\alpha_{[0,\dots,0,q]}} = \delta^\tau_\alpha\,. \tag{2.48}$$

The diagonal elements of the top left $\tilde\beta^{(n)}_q \times \tilde\beta^{(n)}_q$ submatrix of $\mathcal{T}[\chi]$ are now of the form $(\chi_n)^q + \dots$ and the monomial $(\chi_n)^q$ occurs in no other place in the matrix. If we choose $\chi = dx^n$ all entries will vanish with the exception of the above mentioned diagonal elements which will become one. Hence the rank of $\tau_\chi$ is at least $\tilde\beta^{(n)}_q$.

Assume now the rank be greater than $\tilde\beta^{(n)}_q$. We choose in a neighborhood of a point $x_0 \in X$ a new coordinate system $\bar x^i$ given by $\bar x^i = \psi^i(x^j)$ where the function $\psi^n$ satisfies

$$\frac{\partial\psi^n}{\partial x^i}(x_0) = \chi_i\,. \tag{2.49}$$

[5] In the language of Appendix A $N^*\Sigma$ provides a Pfaffian representation of $\Sigma$.

Obviously this is always possible. For the derivatives of class $n$ we obtain

$$\frac{\partial\bar{\Phi}^\tau}{\partial\bar{p}^\alpha_{[0,\dots,0,q]}} = \frac{\partial\Phi^\tau}{\partial p^\alpha_\nu}\prod_{i=1}^n\left(\frac{\partial\psi^n}{\partial x^i}\right)^{\nu_i}\,. \tag{2.50}$$

At the point $x_0$ we can evaluate the derivatives of $\psi$ and the right hand side becomes the matrix of the tableau $\tau_\chi$. By assumption its rank is greater than $\tilde{\beta}^{(n)}_q$. But this implies that we have found a coordinate system where more than $\tilde{\beta}^{(n)}_q$ equations are of class $n$ which clearly contradicts our definition of $\tilde{\beta}^{(n)}_q$. •

**Corollary 2.28** *An involutive equation $\mathcal{R}_q$ is under-determined, if and only if $\tilde{\beta}^{(n)}_q < m$. $\mathcal{R}_q$ is normal, if and only if $\tilde{\beta}^{(n)}_q = m$ and all other $\tilde{\beta}^{(k)}_q = 0$.*

Proof. By definition $\mathcal{R}_q$ is under-determined, if and only if there exists no $\chi$ such that $\operatorname{rank}\tau_\chi = m$. But by the preceding theorem this happens if and only if $\tilde{\beta}^{(n)}_q < m$. Similarly we get the value of $\tilde{\beta}^{(n)}_q$ for a normal equation. It follows from the bijectivity of the tableau that the other $\tilde{\beta}^{(k)}_q$ must vanish. •

One-forms $\chi$ for which the rank of the tableau is less than $\tilde{\beta}^{(n)}_q$ are called *systatic* [117]. For equations which are not under-determined systatic is equivalent to characteristic. It is important to note that under-determinacy depends only on $\tilde{\beta}^{(n)}_q$ and not on the total number $p$ of equations used to describe $\mathcal{R}_q$, as only the former value is intrinsically defined (cf. next section). Of course a system which can be locally described with less equations than unknowns will always be under-determined, but the converse is not true.

**Example 2.29** Consider the Maxwell Equation in two dimensions

$$\begin{aligned} u_{tt} - v_{xt} &= 0\,,\\ u_{xt} - v_{xx} &= 0\,. \end{aligned} \tag{2.51}$$

The tableau of this system is given by the matrix

$$\mathcal{T}[\chi] = \begin{pmatrix} (\chi_1)^2 & -\chi_1\chi_2 \\ \chi_1\chi_2 & -(\chi_2)^2 \end{pmatrix}\,. \tag{2.52}$$

If we multiply the first row by $\chi_2$ and the second by $\chi_1$, we obtain the same. Thus an upper bound for the rank of the tableau is 1 and it takes indeed this value for every non-vanishing one-form $\chi$. This yields $\tilde{\beta}^{(2)}_2 = 1$ and it is not possible to find a coordinate system in which there occur two derivatives of class 2. The tableau is never injective and every one-form is characteristic. Hence the system is indeed under-determined, although there are as many equations as dependent variables. We will see later that this is typical for gauge theories and related to a symmetry. Since $\mathcal{T}[\chi]$ has the same rank for every non-vanishing one-form $\chi$ there are no systatic one-forms and every coordinate system is $\delta$-regular (see below).

**Corollary 2.30** *If there exists a one-form $\chi$ such that the tableau $\tau_\chi$ of the differential equation $\mathcal{R}_q$ is surjective, then $\mathcal{R}_q$ is involutive.*

Proof. $\tau_\chi$ can be only surjective, if the number $p$ of equations in $\mathcal{R}_q$ is less than or equal to the number of dependent variables. The maximal rank of the tableau is then $p$. It follows from Theorem 2.27 that all equations are of class $n$. But this implies immediately that $\mathcal{M}_q$ is involutive and that no integrability conditions arise; thus $\mathcal{R}_q$ is involutive. •

An alternative way of dealing with non-linear equations is to define characteristics with respect to a given solution. For this purpose we introduce the *linearization of an equation $\mathcal{R}_q$ at a solution $\tau$*. If $\mathcal{R}_q$ is locally described by $\Phi^\tau(x^i, u^\alpha, p^\alpha_\mu) = 0$ for $\tau = 1, \ldots, p$ and the section $\sigma$ by $x \mapsto (x, f^\alpha(x))$, we obtain the linearization by expanding around $u^\alpha = f^\alpha, p^\alpha_\mu = \partial_\mu f^\alpha$ and taking only the first order terms. This yields the linear equation

$$\mathcal{P}_q(\sigma) : \left\{ \sum_\alpha \frac{\partial \Phi^\tau}{\partial u^\alpha}\bigg|_{u=f} \bar{u}^\alpha + \sum_{\alpha,\mu} \frac{\partial \Phi^\tau}{\partial p^\alpha_\mu}\bigg|_{u=f} \bar{p}^\alpha_\mu = 0 \right., \tag{2.53}$$

where $u = f$ abbreviates the substitution of $f^\alpha(x)$ and its derivatives for $u^\alpha, p^\alpha_\mu$. The one-form $\chi$ is now said to be characteristic for the solution $\sigma$, if $\chi$ is characteristic for $\mathcal{P}_q(\sigma)$ [123].

## 2.7 $\delta$-Regularity and the $k$-Tableaux

$\delta$-Regularity concerns our criterion (2.19) for involutive symbols. We had to introduce the class of a multi-index to define the $\beta^{(k)}_q$, an obviously coordinate dependent concept. One can show, however, that almost every coordinate system will lead to the same values for the $\beta^{(k)}_q$. In this section we will discuss this problem in greater detail.

As the class is defined for multi-indices, we need to consider only changes of the independent coordinates $x^i$. We will see later that it even suffices to take into account only linear changes. Such a transformation is characterized by a matrix $A$ with constant coefficients $a_{ij}$. If we perform a generic change of coordinates, i.e. without specifying the matrix $A$, the coefficients $a_{ij}$ will appear in the symbol in a complicated, non-linear way.

If we order the columns by class and compute a row echelon form of the symbol in this generic coordinate system, we will obtain the maximal value for $\beta^{(n)}_q$ admitted by the considered differential equation which we called in the last section $\tilde{\beta}^{(n)}_q$. Furthermore this yields the maximal admitted values for the sums $\beta^{(n)}_q + \beta^{(n-1)}_q$, $\beta^{(n)}_q + \beta^{(n-1)}_q + \beta^{(n-2)}_q$, $\ldots$, $\beta^{(n)}_q + \cdots + \beta^{(1)}_q$. These are the intrinsically defined values for the $\beta^{(k)}_q$. Let us denote them for a moment by $\tilde{\beta}^{(k)}_q$. Later we will omit the tilde and always assume that the $\beta^{(k)}_q$ take these values. A more direct definition of them can be given using the Spencer cohomology [117]. Definition 2.8 and all the results in Section 2.3 apply only to the $\tilde{\beta}^{(k)}_q$.

**Definition 2.31** *Let $\mathcal{M}_q$ be the symbol of differential equation; let $\beta^{(k)}_q$ be the number of equations of class $k$ in a given coordinate system. Then the coordinate system is called* $\delta$-regular *for $\mathcal{M}_q$, if $\beta^{(k)}_q = \tilde{\beta}^{(k)}_q$.*

Note that this definition depends crucially on the symbol under consideration. Especially after a projection, when new equations are added to the symbol, the coordinate system must be checked again, even if it was $\delta$-regular before. Prolongations are of course harmless. We further see, why it suffices to consider linear changes of the independent variables. For the analysis of the symbol only one aspect is important: which derivatives of order $q$ do occur after the transformation. But this is determined by the Jacobi matrix. Thus the linearized transformation contains already all relevant information.

For special choices of the coefficients $a_{ij}$ a pivot in the row echelon form may vanish and a row may change to a lower class. Then the corresponding matrix $A$ leads to a $\delta$-singular coordinate system. As the coefficients $a_{ij}$ enter the symbol algebraically, the set of $\delta$-singular systems can be described by an algebraic variety in the space of $n \times n$-matrices. If we make a random choice of $A$, the probability to pick a singular one is zero, as this variety is a set of measure zero. Nevertheless it is of course possible to get one! But this offers an alternative to performing the expensive generic calculation: one generates a random coordinate system and completes the differential equation in this one to an involutive system.

**Example 2.32** $\delta$-regularity is sometimes considered as a purely technical nuisance in geometric approaches to differential equations. This is, however, not true, as it is connected with characteristics. As a simple example, we consider the Wave Equation

$$u_{tt} - u_{xx} = 0\,. \tag{2.54}$$

The coordinate system $(x, t)$ is obviously $\delta$-regular, no matter how we order $x$ and $t$. In a general coordinate system $\eta = \eta(x, t), \zeta = \zeta(x, t)$, Equation (2.54) becomes

$$\begin{aligned}[][\eta_t^2 - \eta_x^2]u_{\eta\eta} + 2[\eta_t\zeta_t - \eta_x\zeta_x]u_{\eta\zeta} + [\zeta_t^2 - \zeta_x^2]u_{\zeta\zeta} +\\ [\eta_{tt} - \eta_{xx}]u_\eta + [\zeta_{tt} - \zeta_{xx}]u_\zeta = 0\,.\end{aligned} \tag{2.55}$$

No matter how we order the new coordinates, we will loose the derivative of class 2, if $\zeta_t = \pm\zeta_x$ and $\eta_t = \mp\eta_x$. But these are exactly the equations for the two characteristics of the Wave Equation. It is not harmful if one of the coordinates is characteristic, as we can always choose the other one as $\bar{x}^2$. This comes from the fact that of course only the coordinates $x^k$ with $\beta_q^{(k)} > 0$ are important for $\delta$-regularity.

One could be tempted to think that since almost every coordinate system is $\delta$-regular, $\delta$-regularity is not important in applications. Unfortunately, this is not true. It turns out that one encounters fairly often differential equations which are not given in such a coordinate system (e.g. "light-cone coordinates"). Thus one needs a practical tool to cope with this problem. In principle, the above said solves it: one must simply compute in a generic coordinate system. In practice this is, however, not feasible, as the computations simply blow up.

There are two reasons for this effect. First we introduce $n^2$ indeterminates which enter the computations in a complicated non-linear way. Thus even if the original system has constant coefficients, they become polynomials in these $n^2$ variables. Second, and this is usually more severe, we make the system dense. A typical differential equation of order $q$ does not depend on all jet variables up to order $q$. Introducing the generic coordinate system generates many new jet variables and all the matrices become dense. One can see this effect very clearly at the symbol. Typically it is a fairly sparse matrix. Now it becomes a dense matrix and the entries are much more complicated, as they contain more jet variables and the new indeterminates.

But one could argue that making a complete change of coordinates is an "over-kill". As we have seen in Example 2.32, only those coordinates $x^k$ where $\beta_q^{(k)} > 0$ are important. Thus one might wonder, whether it is not possible to look only at those. Furthermore one could argue whether one really wants to change the coordinate system. There might be good reasons to use a special one. What we really need, is an intrinsic method to compute the numbers $\beta_q^{(k)}$.

We have already found a way to calculate $\beta_q^{(n)}$ intrinsically through Theorem 2.27. The proof of this theorem showed that in a $\delta$-regular coordinate system the tableau $\tau_\chi$ has maximal rank, if we choose $\chi = dx^n$. We will now introduce generalized tableaux depending on up to $n$ one-forms $\chi^{(k)}$ such that in a $\delta$-regular coordinate system these tableaux have maximal rank, if we choose $\chi^{(k)} = dx^k$, and such that conversely, if $\chi^{(k)}$ are a set of one-forms which lead to maximal rank of all these tableaux, then the coordinate system defined by $d\bar{x}^k = \chi^{(k)}$ is $\delta$-regular. Furthermore the maximal ranks of the generalized tableaux will directly determine the $\beta_q^{(k)}$.

The transformation of derivatives under changes of coordinates look rather messy in multi-index notation. We must hence introduce some notations. If $\mu$ is a multi-index of length $q$, we will denote by $\mathcal{S}_\mu$ the set of all possible realizations of it with repeated indices. An element $s \in \mathcal{S}_\mu$ is thus a tuple of $q$ integers containing $\mu_i$ times the value $i$; e.g. with $n = 2$ one gets $\mathcal{S}_{[1,2]} = \{(1,2,2);(2,1,2);(2,2,1)\}$. We will denote the $k$-th element of $s$ by $s_k$. Furthermore we will abbreviate by $r(\mu)$ the realization of $\mu$ as a repeated index in which the entries are sorted (smallest first).

**Definition 2.33** *Let $\chi^{(k)}, \dots, \chi^{(n)}$ be $n-k+1$ linearly independent one-forms in $T^*X$ for $1 \leq k \leq n$ and let $N^k \subset X$ be a submanifold such that the one-forms represent a basis of $T^*N^k$. Let the differential equation $\mathcal{R}_q$ be locally described by the map $\Phi : J_q\mathcal{E} \to \mathcal{E}'$. The $k$-*tableau *$\tau_{[\chi^{(k)},\dots,\chi^{(n)}]}$ is the linear mapping*

$$\tau_{[\chi^{(k)},\dots,\chi^{(n)}]} : \left\{ \begin{array}{ccccc} S_qT^*N^k \otimes V\mathcal{E} & \xrightarrow{\iota[\chi]} & S_qT^*X \otimes V\mathcal{E} & \xrightarrow{\sigma} & \mathcal{E}' \\ v_\nu^\alpha & \longmapsto & C_\mu^\nu v_\nu^\alpha & \longmapsto & \displaystyle\sum_{\alpha,|\mu|=q} \left(\frac{\partial \Phi^\tau}{\partial p_\mu^\alpha}\right) C_\mu^\nu v_\nu^\alpha \end{array} \right. \tag{2.56}$$

*where $\nu$ is a multi-index with $n$ entries of length $q$ and* $\operatorname{class}\nu \geq k$ *and the factor $C_\mu^\nu$ is defined by*

$$C_\mu^\nu = \sum_{s \in \mathcal{S}_\nu} \prod_{i=1}^q \chi_{m_i}^{(s_i)}, \qquad m = r(\mu)\,. \tag{2.57}$$

We see that the only difference to Definition 2.23 lies in the first map of (2.56) which is generalized to one parameterized by several one-forms. The multi-index $\nu$ provides a convenient way to introduce local coordinates in $S_qT^*N^k \otimes V\mathcal{E}$; one could also say that we choose local coordinates such that $N^k$ is given by $x^1 = const, \dots, x^{k-1} = const$. The matrix of the $k$-tableau is then given by

$$\mathcal{T}_\alpha^{\tau\nu}[\chi^{(k)}, \dots, \chi^{(n)}] = \sum_{|\mu|=q} \left(\frac{\partial \Phi^\tau}{\partial p_\mu^\alpha}\right) C_\mu^\nu\,, \qquad \text{with } \operatorname{class}\nu \geq k\,,\ |\nu| = q\,. \tag{2.58}$$

It has $p$ rows indexed by $\tau$ and $mr_k$ columns indexed by $\alpha$ and $\nu$ where $p$ is again the dimension of $\mathcal{E}'$ and $r_k = \dim S_qT^*N^k \otimes V\mathcal{E} = \binom{n-k+q}{n-k}$. The $n$-tableau is the usual one as defined in the last section. Since $N^1 = X$, the 1-tableau as the largest one has the same size as the symbol. Indeed we will later see that we can identify it with the symbol in a generic coordinate system.

**Lemma 2.34** *The matrix of the $k$-tableau contains the matrices of all $\ell$-tableaux with $\ell > k$ as submatrices.*

Proof. We can view the matrix $\mathcal{T}[\chi^{(k)}, \dots, \chi^{(n)}]$ as a horizontal concatenation of $r_k$ matrices with $p$ rows and $m$ columns each generated similar to $\mathcal{T}[\chi]$ in Section 2.6 but with different mappings $\iota_\chi$. The lemma stems from the fact that in (2.58) all multi-indices with $\operatorname{class}\nu \geq k$ are taken into account. This includes of course all multi-indices with $\operatorname{class}\nu \geq \ell$ for $\ell > k$. (2.57) shows that for such multi-indices the entries of the $k$-tableau do not depend on $\chi^{(k)}, \dots, \chi^{(\ell-1)}$. Thus we obtain all columns of the $\ell$-tableau. •

**Theorem 2.35** *Let $\mathcal{R}_q$ be a differential equation with characters $\beta_q^{(n)}$. Then for $1 \le k \le n$*

$$\max_{\chi^{(k)},\dots,\chi^{(n)}} \operatorname{rank} \tau_{[\chi^{(k)},\dots,\chi^{(n)}]} = \sum_{i=k}^{n} \beta_q^{(i)} \,. \tag{2.59}$$

PROOF. The proof is a straightforward generalization of the one of Theorem 2.27. In a $\delta$-regular coordinate system there exists a local description of $\mathcal{R}_q$ such that the first $\beta_q^{(n)}$ equations are solved for derivatives of class $n$, the next $\beta_q^{(n-1)}$ equations are solved for derivatives of class $n-1$ and so on. We order the columns of the $k$-tableau in such a way that the first $m$ columns represent the $n$-tableau, the first $mr_{n-1}$ columns the $n-1$ tableau and so on. This is always possible according to Lemma 2.34.

Choosing $\chi^{(k)} = dx^k, \dots, \chi^{(n)} = dx^n$, we can easily evaluate the factors $C_\mu^\nu$. They vanish whenever $\mu \neq \nu$ and are one otherwise. Thus we have eliminated all contributions to the $k$-tableau by derivatives whose class is less than $k$. If the principal derivative of equation $\tau$ is $p_\mu^\alpha$, then in row $\tau$ the column indexed by $\alpha$ and $\mu$ contains the first non-vanishing entry. Since our local representation was chosen in such a way that the symbol is in solved form, all principal derivatives are different and we have at least $\sum_{i=k}^n \beta_q^{(i)}$ linearly independent rows.

On the other hand assume that there are one-forms $\chi^{(k)}, \dots, \chi^{(n)}$ such that the rank of the $k$-tableau is larger than the claimed value. Then we apply a coordinate transformation $\bar{x}^j = \psi^j(x)$ where the functions $\psi^j$ satisfy in a point $x_0 \in X$

$$\frac{\partial \psi^j}{\partial x^i}(x_0) = \chi_i^{(j)} \,, \quad j = k, \dots, n \,. \tag{2.60}$$

Due to the linear independence of the one-forms $\chi^{(i)}$ this is always possible. The effect of this transformation on the symbol is determined by the highest order part of the transformation law for the derivatives of order $q$. It is given by

$$\frac{\partial p_\mu^\alpha}{\partial \bar{p}_\nu^\alpha} = \sum_{s \in \mathcal{S}_\nu} \prod_{i=1}^{q} \left( \frac{\partial \psi^{s_i}}{\partial x^{m_i}} \right) , \quad |\mu| = |\nu| = q, \; m = r(\mu) \,. \tag{2.61}$$

Since we are interested in the values of $\beta_q^{(k)}, \dots, \beta_q^{(n)}$, we consider only the following derivatives

$$\frac{\partial \bar{\Phi}^\tau}{\partial \bar{p}_\nu^\alpha} = \sum_{|\mu|=q} \left( \frac{\partial \Phi^\tau}{\partial p_\mu^\alpha} \right) \frac{\partial p_\mu^\alpha}{\partial \bar{p}_\nu^\alpha} \,, \quad \operatorname{class} \nu \ge k \,. \tag{2.62}$$

But at $x_0$ we can evaluate the derivatives of $\psi^j$ occurring at the right hand side when we plug in (2.61). Comparing (2.62) and (2.58) we immediately recognize the $k$-tableau at the right hand side. Thus we have found a coordinate system in which we have more equations of class greater than or equal to $k$ than given by $\sum_{i=k}^n \beta_q^{(i)}$. But this contradicts the definition of the $\beta_q^{(k)}$. ∎

**Definition 2.36** *Let $\chi^{(i)} \in T^*X$, $i = 1, \dots, n$ be $n$ linearly independent one-forms. They define a* non-systatic basis *of $T^*X$, if*

$$\operatorname{rank} \tau_{[\chi^{(k)},\dots,\chi^{(n)}]} = \sum_{i=k}^{n} \beta_q^{(i)} \,, \qquad k = 1, \dots, n \,. \tag{2.63}$$

*Coordinates $x^i$ such that $dx^i = \chi^{(i)}$ are called an* associated coordinate system.

Different associated coordinate system differ only by constants, namely the choice of an origin. Note that in this definition only the $\chi^{(k)}$ where $\beta_q^{(k)} > 0$ really matter. For the remaining one-forms one can choose any which complete to a basis of $T^*X$. Theorem 2.35 implies now immediately the following

**Corollary 2.37** *A coordinate system on $X$ is $\delta$-regular, if and only if it is associated to a non-systatic basis of $T^*X$.*

From (2.61) and (2.62) we also see that if we write down the symbol in a generic linear coordinate system, i.e. in coordinates $\bar{x}^j = \sum a_i^j x^i$, then we get the 1-tableau after identifying $a_i^j$ with $\chi_i^{(j)}$. Thus the introduction of the $k$-tableau allows us to introduce these generic coordinates step by step. Since we often must change only some of the coordinates, this can save considerable computing time in typical applications, as one rarely has to go until the 1-tableau. Note furthermore that Lemma 2.34 allows us to organize the computation in an efficient way. If we calculate the rank of the $k$-tableau as column rank, we can keep the results from the $(k+1)$-tableau, as we only add further columns to the matrix.

**Example 2.38** We demonstrate the use of the $k$-tableaux in an example stemming from a concrete application [83]. The analysis of moving pseudo-spherical surfaces in $\mathbb{R}^3$ leads to the so-called Bianchi equation. This is a third-order equation of the form:

$$\begin{aligned}
&u_{xyt} - u_{yt}u_x \cot u + u_{xt}u_y \tan u = 0\,,\\
&\left(\frac{u_{xt}}{\cos u}\right)_x - K(K\sin u)_t - \frac{u_{yt}u_y}{\sin u} = 0\,,\\
&\left(\frac{u_{yt}}{\sin u}\right)_y + K(K\cos u)_t - \frac{u_{xt}u_x}{\cos u} = 0\,,
\end{aligned} \tag{2.64}$$

where $K$ is some given function of $t$. Obviously the used coordinate system is not $\delta$-regular no matter how we order the coordinates, for there is no derivative of class 3. We will use the ordering $x^1 = t$, $x^2 = x$ and $x^3 = y$. Since each equation in (2.64) contains only one third order derivative, the symbol takes after some trivial manipulations the simple form

$$\begin{aligned}
v_{xyt} &= 0\,,\\
v_{xxt} &= 0\,,\\
v_{yyt} &= 0\,.
\end{aligned} \tag{2.65}$$

Now the matrix of the 3-tableau is readily computed.

$$\mathcal{T}[\chi^{(3)}] = \begin{pmatrix} \chi_1^{(3)}\chi_2^{(3)}\chi_3^{(3)} \\ \chi_1^{(3)}(\chi_2^{(3)})^2 \\ \chi_1^{(3)}(\chi_3^{(3)})^2 \end{pmatrix}. \tag{2.66}$$

We find two families of characteristic one-forms

$$\begin{aligned}
\omega_1 &= \alpha\, dx^1\,,\\
\omega_2 &= \beta\, dx^2 + \gamma\, dx^3\,.
\end{aligned} \tag{2.67}$$

This implies especially that the three coordinate forms $dx^1$, $dx^2$, $dx^3$ are all characteristic. Thus it is not surprising that the coordinate system is not $\delta$-regular. The maximal rank of the 3-tableau is 1 and obtained e.g. with the choice $\chi^{(3)} = dx^1 + dx^3$.

Next we compute the 2-tableau. There exist four third order derivatives of class 2, namely [0,0,3], [0,1,2], [0,2,1] and [0,3,0]. Hence its matrix has four columns:

$$\mathcal{T}[\chi^{(2)},\chi^{(3)}] = \left(\begin{array}{ll} \chi_1^{(3)}\chi_2^{(3)}\chi_3^{(3)} & \chi_1^{(2)}\chi_2^{(3)}\chi_3^{(3)}+\chi_1^{(3)}\chi_2^{(2)}\chi_3^{(3)}+\chi_1^{(3)}\chi_2^{(3)}\chi_3^{(2)} \\ \chi_1^{(3)}(\chi_2^{(3)})^2 & \chi_1^{(2)}(\chi_2^{(3)})^2+2\chi_1^{(3)}\chi_2^{(3)}\chi_2^{(2)} \\ \chi_1^{(3)}(\chi_3^{(3)})^2 & \chi_1^{(2)}(\chi_3^{(3)})^2+2\chi_1^{(3)}\chi_3^{(3)}\chi_3^{(2)} \end{array}\right.$$
$$\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \qquad (2.68)$$
$$\left.\begin{array}{cc} \chi_1^{(2)}\chi_2^{(2)}\chi_3^{(3)}+\chi_1^{(2)}\chi_2^{(3)}\chi_3^{(2)}+\chi_1^{(3)}\chi_2^{(2)}\chi_3^{(2)} & \chi_1^{(2)}\chi_2^{(2)}\chi_3^{(2)} \\ \chi_1^{(3)}(\chi_2^{(2)})^2+2\chi_1^{(2)}\chi_2^{(2)}\chi_2^{(3)} & \chi_1^{(2)}(\chi_2^{(2)})^2 \\ \chi_1^{(3)}(\chi_3^{(2)})^2+2\chi_1^{(2)}\chi_3^{(2)}\chi_3^{(3)} & \chi_1^{(2)}(\chi_3^{(2)})^2 \end{array}\right).$$

One can now compute the maximal rank of this matrix using Gaussian elimination. It is, however, much easier to substitute the above mentioned non-characteristic one-form $\chi^{(3)} = dx^1 + dx^3$. This yields the following, much simpler matrix

$$\mathcal{T}[\chi^{(2)}, dx^1+dx^3] = \begin{pmatrix} 0 & \chi_2^{(2)} & \chi_1^{(2)}\chi_2^{(2)}+\chi_2^{(2)}\chi_3^{(2)} & \chi_1^{(2)}\chi_2^{(2)}\chi_3^{(2)} \\ 0 & 0 & (\chi_2^{(2)})^2 & \chi_1^{(2)}(\chi_2^{(2)})^2 \\ 1 & \chi_1^{(2)}+2\chi_2^{(2)} & (\chi_3^{(2)})^2+2\chi_1^{(2)}\chi_3^{(2)} & \chi_1^{(2)}(\chi_3^{(2)})^2 \end{pmatrix}. \qquad (2.69)$$

One sees at once that the rank of this matrix is 3 whenever $\chi_2^{(2)} \neq 0$. Thus we get $\beta_3^{(3)} = 1$ and $\beta_3^{(2)} = 2$ and we have also found a non-systatic basis of $T^*X$

$$\begin{aligned} \chi^{(3)} &= dx^1 + dx^3\,, \\ \chi^{(2)} &= dx^2\,, \\ \chi^{(1)} &= dx^1\,, \end{aligned} \qquad (2.70)$$

An associated $\delta$-regular coordinate system is given by $\bar{x}^3 = y + t$, $\bar{x}^2 = x$ and $\bar{x}^1 = t$. This yields the following form for the symbol

$$\begin{aligned} \bar{v}_{333} + \bar{v}_{331} &= 0\,, \\ \bar{v}_{332} + \bar{v}_{321} &= 0\,, \\ \bar{v}_{322} + \bar{v}_{221} &= 0\,, \end{aligned} \qquad (2.71)$$

where one can easily read off the $\beta_3^{(k)}$ directly.

# Chapter 3

# Arbitrariness

*We will study in some detail how to define "general solutions" for an involutive equation. For certain closed form representations we will compute possible values for the number of arbitrary functions contained in them. This yields a criterion whether a given solution is a general one.*

## 3.1 General Solution

Intuitively it seems to be obvious what the general solution of a differential equation is: an expression containing some arbitrary elements such that any solution of the equation can be obtained through specialization of these. This is, however, usually not possible, as most differential equations admit so-called "singular solutions" which do not fit into such a general expression. In the last century many studies were devoted to the relationship between "general solutions", "complete solutions", "particular solutions" etc. However, no real definition emerged. Later interest in this question ceased.

The geometric theory provides a framework which allows for an easy definition following an idea of Ampère [2, 159]. He proposed that the only differential equations satisfied by a general solution should be those of the original system and those derivable of them (i.e. algebraic combinations and prolongations). In other words, if we eliminate the arbitrary elements of a general solution, we obtain just the original equations or some combinations of them. We have already employed this idea in Definition 2.20 of a general formal power series solution. Following Hermann [72] we can express it more precisely.

**Definition 3.1** *A* general solution *of the differential equation $\mathcal{R}_q$ is a subset $\Theta \subset \Gamma_{loc}(\mathcal{E})$ such that*

*(i)* $\forall r \geq 0 \;\; j_{q+r}(\Theta) \subseteq \mathcal{R}_{q+r}$.

*(ii) There does not exist for any $r \geq 0$ a submanifold $\bar{\mathcal{R}}_{q+r} \subset \mathcal{R}_{q+r}$ of lower dimension such that $j_{q+r}(\Theta) \subseteq \bar{\mathcal{R}}_{q+r}$.*

*$\Theta$ is called a* local general solution at $x_0 \in X$, *if all sections $\sigma \in \Theta$ are defined in a neighborhood of $x_0$.*

Now the question naturally arises how one can describe such subsets $\Theta \subset \Gamma_{loc}(\mathcal{E})$. In Section 2.5 we dealt with power series solutions. In applications one usually wants *closed form solutions.* This is another term notoriously difficult to define. We will use here an idea sketched by Hermann [72].

**Definition 3.2** *Let $Y_k$ be manifolds and $V_k$ vector spaces for $k = 1, \dots, p$ and let $W$ be a further manifold. All manifolds and vector spaces are assumed to be finite dimensional. A* (local) closed form representation at $x_0 \in X$ *of a general solution is a smooth map*

$$s : \mathcal{C}^\infty(Y_1, V_1) \times \cdots \times \mathcal{C}^\infty(Y_p, V_p) \times W \mapsto \Gamma_{loc}(\mathcal{E}) \tag{3.1}$$

*such that its image provides a (local) general solution at $x_0$.*

This definition is still somewhat vague, as it does not contain any further restriction on the map $s$. It should of course be in some sense in closed form, too. One could for instance require that in some coordinate system $s$ can be expressed in terms of some "allowed functions". These could be elementary functions or additionally some special functions. Precise definitions of such terms are currently only known within the framework of differential algebra [85, 86] where one builds by purely algebraic means extensions of fields of rational functions. We do not pursue this problem here any further.

$\dim W$ corresponds to the number of arbitrary constants. Each factor $\mathcal{C}^\infty(Y_k, V_k)$ represents $\dim V_k$ arbitrary functions depending on $\dim Y_k$ arguments. The Taylor coefficients of these functions plus the constants correspond to the coordinates of the manifolds $W_q$ in Definition 2.20. There are now two basic ingredients necessary to build a closed form representation in a neighborhood $U(x_0)$ of $x_0$ : First we need maps $\omega_k : U(x_0) \mapsto Y_k$ to define the arguments of the arbitrary functions; $f \circ \omega_k$ transforms a function $f$ over $Y_k$ into one over $U(x_0)$. Then as a second step we must say how to combine these functions.

**Example 3.3** As a simple example we consider again the Wave Equation $u_{xx} - u_{tt} = 0$. To obtain a closed form representation in the sense of Definition 3.2, we take $p = 1$, $Y_1 = \mathbb{R}$ and $V_1 = \mathbb{R}^2$ (there is no need for a $W$). Then we set

$$s_1 : \begin{pmatrix} f \\ g \end{pmatrix} \mapsto f(x - t) + g(x + t)\,. \tag{3.2}$$

This is the well-known formula for the general solution as superposition of left and right moving waves. Alternatively, we could take the D'Alembert formula

$$s_2 : \begin{pmatrix} f \\ g \end{pmatrix} \mapsto \frac{1}{2}\big[f(x - t) + f(x + t)\big] + \frac{1}{2}\int_{x-t}^{x+t} g(\tau)d\tau\,. \tag{3.3}$$

Restricting our attention to analytic solutions we could say that a Taylor expansion of $s$ at $x_0$ yields a general power series solution. This is equivalent, as condition (2.37) can be directly translated into the two parts of Definition 3.1. The requirement that every function in $I_\infty(x_0)$ must vanish after substituting the power series corresponds to Part (i) and the converse yields Part (ii), as an additional function vanishing on $\mathcal{R}_\infty$ could be used to define for one $r$ a differential equation $\bar{\mathcal{R}}_{q+r}$ of lower dimension than $\mathcal{R}_{q+r}$ which is also satisfied by $s$.

## 3.2 Cartan Characters and Hilbert Polynomial

In Section 2.2 the symbol helped to decide whether or not a differential equation is in involution. The symbol of an involutive equation contains, however, much more information on the solution space. The Cartan characters and the Hilbert polynomial will be our basic tools to extract it.

**Definition 3.4** *The* Cartan characters $\alpha_q^{(k)}$ *of the involutive differential equation* $\mathcal{R}_q$ *in* $n$ *independent and* $m$ *dependent variables are*

$$\alpha_q^{(k)} = m\binom{q+n-k-1}{q-1} - \beta_q^{(k)} \quad k = 1,\dots,n\,. \tag{3.4}$$

$\beta_q^{(k)}$ gives the number of principal derivatives of class $k$ and order $q$, i.e. derivatives which can be computed using $\mathcal{R}_q$ by solving for them; $\alpha_q^{(k)}$ is the number of corresponding parametric derivatives, i.e. derivatives which must be provided by initial or boundary conditions to specify a unique solution.

**Proposition 3.5** *The Cartan characters of an involutive differential equation* $\mathcal{R}_q$ *satisfy*

$$\alpha_q^{(1)} \geq \alpha_q^{(2)} \geq \cdots \geq \alpha_q^{(n)} \geq 0\,. \tag{3.5}$$

PROOF. For $q = 1$ this is obvious from Corollary 2.13, as $\alpha_1^{(k)} = m - \beta_1^{(k)}$. For a higher order equation we exploit the fact (shown in Appendix C.2), that it can be transformed into an equivalent first-order equation with exactly the same Cartan characters. •

For an involutive symbol it is possible to compute the Cartan characters of its prolongations without explicitly determining the equations describing these symbols. They are given by a simple recursion relation.

**Lemma 3.6** *Let* $\mathcal{M}_q$ *be an involutive symbol with characters* $\alpha_q^{(k)}$ *and* $\beta_q^{(k)}$. *Then the characters of its prolongation* $\mathcal{M}_{q+r}$ *are given by*

$$\alpha_{q+r}^{(k)} = \sum_{i=k}^{n}\binom{r+i-k-1}{r-1}\alpha_q^{(i)}\,, \tag{3.6}$$

$$\beta_{q+r}^{(k)} = \sum_{i=k}^{n}\binom{r+i-k-1}{r-1}\beta_q^{(i)}\,. \tag{3.7}$$

PROOF. It suffices to prove (3.7), as (3.6) is then an immediate consequence of (3.4). We use an induction over $r$. Prolonging an equation of class $k$ or higher with respect to $x^k$ yields an equation of class $k$. By Proposition 2.9 these are the only independent equations of order $q+1$ in the prolongation, as $\mathcal{M}_q$ is involutive. Thus $\beta_{q+1}^{(k)} = \beta_q^{(k)} + \cdots + \beta_q^{(n)}$ and we have proven (3.7) for $r = 1$.

Let us now assume that (3.7) holds for $r-1$. Since according to Theorem 2.14 the prolongation of an involutive symbol is again involutive, we can apply the formula for $r = 1$ to get $\beta_{q+r}^{(k)} = \beta_{q+r-1}^{(k)} + \cdots + \beta_{q+r-1}^{(n)}$. Iterating yields

$$\beta_{q+r}^{(k)} = \sum_{i=k}^{n}\sum_{l=k}^{i}\binom{r+l-k-2}{r-2}\beta_q^{(i)} = \sum_{i=k}^{n}\binom{r+i-k-1}{r-1}\beta_q^{(i)} \tag{3.8}$$

where the last equality follows from a well-known identity for binomial coefficients. •

Lemma 3.6 allows us to compute easily the rank and the dimension of a prolonged symbol $\mathcal{M}_{q+r}$. By definition we have $\operatorname{rank}\mathcal{M}_{q+r} = \sum_{k=1}^{n}\beta_{q+r}^{(k)}$ and $\dim\mathcal{M}_{q+r} = \sum_{k=1}^{n}\alpha_{q+r}^{(k)}$. Substituting (3.6) and (3.7) yields

$$\operatorname{rank}\mathcal{M}_{q+r} = \sum_{k=1}^{n}\binom{r+k-1}{r}\beta_q^{(k)}\,, \tag{3.9}$$

$$\dim\mathcal{M}_{q+r} = \sum_{k=1}^{n}\binom{r+k-1}{r}\alpha_q^{(k)}\,. \tag{3.10}$$

**Theorem 3.7** *The Hilbert polynomial of an involutive equation $\mathcal{R}_q$ with Cartan characters $\alpha_q^{(k)}$ is given by*

$$H_q(r) = \sum_{i=0}^{n-1}\left(\sum_{k=i}^{n-1}\frac{\alpha_q^{(k+1)}}{k!}s_{k-i}^{(k)}(0)\right)r^i\,. \tag{3.11}$$

*Conversely, if the Hilbert polynomial of $\mathcal{R}_q$ is given by $H_q(r) = \sum h_i r^i$, then its Cartan characters satisfy the recursion relation*

$$\alpha_q^{(k)} = (k-1)!\,h_{k-1} - \sum_{i=k+1}^{n}\frac{(k-1)!}{(i-1)!}\,\alpha_q^{(k)}s_{i-k}^{(i-1)}(0)\,,\quad k = 1,\dots,n\,. \tag{3.12}$$

Proof. According to Definition 2.22, $H_q(r) = \dim\mathcal{M}_{q+r}$. In order to express it in terms of the Cartan characters we must write the binomial coefficients explicitly as polynomials in $r$. This is most easily done using the modified Stirling numbers introduced in Appendix C.1. (C.2) yields then (3.11). There is obviously a one-to-one correspondence between the coefficients of the Hilbert polynomial and the Cartan characters. A simple inversion of (3.11) yields (3.12). •

An equation $\mathcal{R}_q$ with vanishing Cartan characters is said to be of *finite type*, as there are no free Taylor coefficients of order $q$ or higher. In the older literature (e.g. [159]) the name *Mayer system* was often used for such equations. They have a finite dimensional solution space parameterized by $\dim\mathcal{R}_q$ parameters. System (2.19) was an example for a finite type system. We will assume in what follows that at least one character is non-zero, as we will show in Section 7.1 that the integration of any system of finite type can be reduced to solving systems of ordinary differential equations.

## 3.3 Algebraic Representations

We take now a closer look at those closed form representations of the general solution where the map $s$ in Definition 3.2 involves neither differentiations nor integrations. Such representations are called *algebraic*. Representation $s_1$ of the general solution of the Wave Equation is a prototypical example. Representation $s_2$ is, however, not algebraic, as it involves an integration.

The image of an algebraic map $s$ can be expressed as sections of the form

$$u^\alpha = \Psi^\alpha\Big(x^i,\dots,F(\phi_1(x),\dots,\phi_k(x)),\dots\Big) \tag{3.13}$$

where the functions $\phi_i$ represent the arguments of the arbitrary functions $F$. Thus they are fixed like $\Psi$. $F$ stands here generically for all arbitrary functions involved. We denote by $f_k$ the number of arbitrary functions depending on $k$ arguments. Note that they do not need to have the same arguments.

**Example 3.8** As a second, simple example we consider with $m = 1$ and $n = 2$ the non-linear equation $uu_{xy} - u_xu_y = 0$. $u = \phi_1(x)\phi_2(y)$ represents a general solution. Thus we have an algebraic representation with two functions of one argument: $\Psi(a,b) = ab$ and $F_1 = \{\phi_1,\phi_2\}, F_0 = F_2 = \{\}$.

Our goal will be to determine the possible values for these numbers $f_k$. If we introduce, following [103], as short hand

$$\begin{bmatrix}k\\ \ell\end{bmatrix} = \binom{k+\ell-1}{\ell}\,, \tag{3.14}$$

then an arbitrary function $F$ with $k$ arguments yields $\left[{k \atop q+r}\right]$ arbitrary Taylor coefficients at order $q+r$ (cf. Eq. (2.9)). Expression (3.13) contains thus a total of

$$T_q(r) = \sum_{k=1}^{n} f_k \begin{bmatrix} k \\ q+r \end{bmatrix} \tag{3.15}$$

arbitrary Taylor coefficients of order $q+r$. If we assume that the expansion of (3.13) leads to a power series which can be constructed order by order, then $T_q(r)$ must equal the Hilbert polynomial $H_q(r)$. Expanding both as polynomials in $r$ and equating the coefficients leads to a triangular system of linear equations for the $f_k$ which can be easily solved by back-substitution. Thus we have proved

**Theorem 3.9** *If there exists an algebraic representation of the general solution of the involutive differential equation $\mathcal{R}_q$ with Cartan characters $\alpha_q^{(k)}$ whose power series expansion can be constructed order by order, then it contains $f_k$ free functions of $k$ variables, where the $f_k$ are determined by the recursion relation:*

$$\begin{aligned} f_n &= \alpha_q^{(n)}\,, \\ f_k &= \alpha_q^{(k)} + \sum_{i=k+1}^{n} \frac{(k-1)!}{(i-1)!}\left(\alpha_q^{(i)} s_{i-k}^{(i-1)}(0) - f_i s_{i-k}^{(i-1)}(q)\right)\,, \quad 0 < k < n\,. \end{aligned} \tag{3.16}$$

*Such a representation can exist only, if the solution of this recursion relation contains only non-negative integers*[1]*.*

With the results of Appendix C.1 we can solve the recursion relation (3.16) explicitly for first-order equations and even prove that the solution is non-negative. Like all results in this section, this is a purely formal statement which does not imply anything about the existence of a solution, because we consider only formal power series solutions. Some existence theorems will later be given in the context of initial value problems (cf. Chapter 4).

**Corollary 3.10** *For a first-order equation $\mathcal{R}_1$ the solution of (3.16) is given by*

$$\begin{aligned} f_n &= \alpha_1^{(n)} = m - \beta_1^{(n)}\,, \\ f_k &= \alpha_1^{(k)} - \alpha_1^{(k+1)} = \beta_1^{(k+1)} - \beta_1^{(k)}\,. \end{aligned} \tag{3.17}$$

*It contains only non-negative integers.*

Proof. The proof is by descending induction. The case $k = n-1$ is obvious from (C.7). If we assume that (3.17) holds for $k+1$, we get for $k$ that

$$\begin{aligned} f_k \;=\;& \alpha_1^{(k)} - \alpha_1^{(k+1)} + \\ & (k-1)! \sum_{i=k+2}^{n-1} i!\ \alpha_1^{(i+1)} \left(i\, s_{i-k}^{(i-1)}(1) + s_{i-k+1}^{(i)}(0) - s_{i-k+1}^{(i)}(1)\right)\,. \end{aligned}$$

But Lemma C.3 implies that the expression in the bracket vanishes for all $i, k$. It follows from Proposition 3.5 that the $f_k$ are always non-negative. •

[1] Using a computer algebra system one can show that even for fairly high values of $n$ (3.16) always yields integer solutions. Thus one is tempted to conjecture that this holds for all values of $n$. One can, however, also show that this is highly non-trivial, as $s_{i-k}^{(i-1)}(q)$ is in general not divisible by the denominator of (3.16). Thus a proof of this conjecture would probably need a closed-form solution of this recursion relation.

One might think that every equation should have such an algebraic representation of its general solution. But we will see that it is actually a rare phenomenon for higher-order equations and in most of the examples we will treat no such representation will exist. One could consider algebraic representations as a kind of generalization of the so-called *non-linear superposition* found in some ordinary differential equations. These are equations were the general solution can be built from some *fundamental solutions* $\rho_k$, i.e. it can be written in the form

$$u^\alpha = \Psi^\alpha(\rho_1, \dots, \rho_q, c_1, \dots, c_p) \tag{3.18}$$

with arbitrary constants $c_k$.

In the case of partial differential equations these constants must be replaced by arbitrary functions. Thus we obtain an expression like (3.13). The explicit dependency of the independent variables is now given by the fundamental solutions. Lie [98] proved that in the case of first-order systems of ordinary differential equations such a non-linear superposition rule exists, if and only if the system is of the special form

$$\frac{\partial u^\alpha}{\partial x} = \sum_{k=1}^{r} X_k(x)\xi_k^\alpha(u^\beta) \tag{3.19}$$

where the $X_k$ are arbitrary functions of $x$ and the vector fields

$$V_k = \sum_\alpha \xi_k^\alpha \frac{\partial}{\partial u^\alpha}, \quad k = 1, \dots, r, \tag{3.20}$$

span an $r$-dimensional Lie algebra. Thus it is not surprising that in the case of partial differential equations we will rarely find algebraic representations.

**Example 3.11** Consider a normal equation of order $q > 1$. We know from Corollary 2.28 that for such an equation $\beta_q^{(n)} = m$ and all other $\beta_q^{(k)}$ vanish. Thus the Cartan characters of a normal equation are given by

$$\alpha_N^{(n)} = 0, \quad \alpha_N^{(k)} = m\binom{q+k-1}{q-1}. \tag{3.21}$$

(3.16) yields with these values $f_n = 0$, $f_{n-1} = mq$ and $f_{n-2} = mq(1-q)/2$. Thus for $q > 1$ and $n > 2$ we always get negative values in the solution! $n = 2$ is a special case, because then functions of at most one argument can occur. But for such functions integration or differentiation does not change the number of Taylor coefficients at a given order. Thus from the point of view of our formal analysis such functions can always be considered as algebraic.

We will study in the next section the effect of more general representations than purely algebraic ones. We will, however, always stick to the assumption that a series expansion leads to a power series which can be constructed order by order. This is basically a condition on the functions $\Psi^\alpha$ in (3.13) which is especially then satisfied, if these functions do not depend explicitly on the independent variables $x^i$ as e.g. in the case of the solution (3.2) of the Wave Equation.

## 3.4 Representations with Integrals

Even if Theorem 3.9 yields for a given equation only non-negative values for the $f_k$, it is interesting to study the effect of more general expressions than (3.13), as many solution methods for differential equations lead for example to integral representations. Then further terms appear in the definition of $T_q(r)$ [2]

$$T_q(r) = \sum_{j\in J}\sum_{k=1}^{n} f_{k,j}\begin{bmatrix} k \\ q+r+j \end{bmatrix}. \tag{3.22}$$

$J$ is some finite set of integers containing the considered *differentiation orders*. $j = 0$ represents the algebraic case. Positive values of $j$ correspond to derivatives; negative values denote integrals.

The shifts in the lower entry of the bracket symbol reflect the simple fact that if we integrate a power series the coefficients of order $q$ become coefficients of order $q+1$. Differentiation is slightly more complicated, as it affects only power products which contain the variable with respect to which one differentiates. To avoid tedious case distinctions we assume that if an arbitrary function is differentiated then all its derivatives of the considered order appear. Differentiation will mainly occur in the context of gauge systems (cf. Section 6.1), where this restriction is almost always satisfied.

The equality $H_q(r) = T_q(r)$ leads now to the following linear system of $n$ diophantine equations

$$\sum_{j\in J}\sum_{k=i}^{n-1} \frac{n!}{k!} s_{k-i}^{(k)}(q+j) f_{k+1,j} = n!\, h_i\,, \quad i = 0,\ldots,n-1 \tag{3.23}$$

where $h_i$ denotes the coefficients of the Hilbert polynomial. We have multiplied each equation by $n!$ to avoid fractions. All coefficients including the right hand side are non-negative integers.

If $|J| > 1$, the system is under-determined, i.e. the number of arbitrary functions is no longer uniquely defined. Even if we disregard how a function enters the general solution and consider only the total number $\tilde f_k = \sum_{j\in J} f_{k,j}$ of functions with $k$ variables, we get in general no unique results, as we will show later in this section. There exists, however, one invariant:

**Theorem 3.12** *Let $\alpha_q^{(k_0)}$ be the highest non-vanishing Cartan character of the involutive equation $\mathcal{R}_q$. Then every non-negative solution $\{f_{k,j}\}$ of the diophantine system (3.23) satisfies*

$$f_{k,j} = 0\,, \quad \forall j \in J, k > k_0\,, \tag{3.24}$$

$$\tilde f_{k_0} = \sum_{j\in J} f_{k_0,j} = \alpha_q^{(k_0)}\,. \tag{3.25}$$

Proof. The proof is simple, as under the assumption of the theorem $H_q(r)$ is a polynomial of degree $k_0 - 1$. Setting $h_i = 0$ for $i = k_0,\ldots,n-1$ leads consecutively to the equations $\tilde f_n = 0,\ldots,\tilde f_{k_0+1} = 0$. Since only non-negative solutions are allowed, this entails (3.24). The next order provides (3.25). •

Note that it is essential here that we maintain the assumption that the power series expansion can be constructed order by order. Otherwise the ansatz $H_q(r) = T_q(r)$ is no longer correct and we cannot draw any conclusions. We call $\alpha_q^{(k_0)}$ the *degré d'arbitraire* (or *index of generality*) of the differential equation $\mathcal{R}_q$. $k_0$ is called *Cartan genus* of $\mathcal{R}_q$. No solution of $\mathcal{R}_q$ satisfying the above mentioned assumption can contain an arbitrary function depending on more than $k_0$ arguments.

[2] In order to get a well-defined expression for $j < -q$, we will adopt the convention $\begin{bmatrix} n \\ k \end{bmatrix} = 0$ for $k < 0$.

**Corollary 3.13** *An involutive equation $\mathcal{R}_q$ is under-determined, if and only if its Cartan genus is $n$.*

Proof. By Corollary 2.28 an involutive equation is under-determined, if and only if we have $\beta_q^{(n)} < m$. But then $\alpha_q^{(n)} > 0$ and the Cartan genus is $n$. Note that this implies as predicted in Section 2.6 that the general solution contains arbitrary functions of all $n$ independent variables, hence at least one dependent variable is not constrained at all. •

The condition that all $f_{k,j}$ must be non-negative implies that only a finite number of solutions of (3.23) can exist, as the right hand sides can be seen as bounds on weighted sums of the $f_{k,j}$. One can readily design an algorithm which generates for any given set $J$ all non-negative solutions of (3.23).

Since the number of solutions may be very large, it is nevertheless interesting to study ways to construct special solutions. The easiest possibility is to set $J = \{j_0\}$. The condition $H_q(r) = T_q(r)$ leads then to a well-determined linear system of equations for the $f_{k,j_0}$. Its solution is given by the recursion relation (3.16), if we replace the argument $q$ by $q+j_0$ in the second modified Stirling number.

For two special values of $j_0$ this modified recursion relation can easily be solved. Setting $j_0 = -q$ gives at once

$$f_{k,-q} = \alpha_q^{(k)}, \qquad k = 1, \dots, n. \tag{3.26}$$

Choosing $j_0 = 1-q$, we find a solution with $(q-1)$-dimensional integrals generalizing (3.17):

$$f_{n,1-q} = \alpha_q^{(n)}, \quad f_{k,1-q} = \alpha_q^{(k)} - \alpha_q^{(k+1)}. \tag{3.27}$$

Again Proposition 3.5 implies that it is a non-negative solution. Note that these two solutions yield the same values for the $\tilde{f}_k$ only if $\alpha_q^{(n)} = \cdots = \alpha_q^{(2)} = 0$.

Another way to construct non-negative solutions of (3.23) is to exploit the relation

$$\begin{bmatrix} n \\ k \end{bmatrix} = \begin{bmatrix} n \\ k-1 \end{bmatrix} + \begin{bmatrix} n-1 \\ k \end{bmatrix}, \tag{3.28}$$

which is an immediate consequence of the Pascal triangle. If now $f_k$ is a negative value in the solution of (3.16), we can eliminate it through

$$f_{k+1} \begin{bmatrix} k+1 \\ q+r \end{bmatrix} + f_k \begin{bmatrix} k \\ q+r \end{bmatrix} = (f_{k+1} + f_k) \begin{bmatrix} k+1 \\ q+r \end{bmatrix} - f_k \begin{bmatrix} k+1 \\ q+r-1 \end{bmatrix}, \tag{3.29}$$

i.e., we “convert” some functions of $k+1$ variables into integrals! But this may result in a negative value for the new $f_{k+1}$. Then we can apply the same trick again, if $k+1 < n$. But if $f_n$ becomes negative, the procedure has failed. Thus this method will only work, if $\sum_{i=k}^{n} f_i > 0$. It can be generalized by iteration of (3.28) to include higher-dimensional integrals (see [133]).

Finally, we consider the problem of how to decide whether a solution constructed by some technique is the general solution. Under the assumption that the corresponding power series can be constructed order by order, we must simply count the arbitrary functions and compare the resulting $T_q(r)$ with the Hilbert polynomial $H_q(r)$ of the differential equation.

**Corollary 3.14** *Let a given solution of the involutive differential equation $\mathcal{R}_q$ contain $f_{k,j}$ functions of differentiation order $j$ with $k$ arguments. Let its power series be constructible order by order. Then the solution is a general one, if and only if the numbers $f_{k,j}$ solve the diophantine system (3.23) for a set $J$ containing all occurring values of $j$.*

# Chapter 4

# Initial Value Problems

*We will study in detail the properties of initial value problems. This entails a precise definition of well-posed problems and some existence and uniqueness theorems for analytic systems. For linear equations we will generalize the Holmgren Theorem to obtain uniqueness also for non-analytic solutions. Finally, we will comment on the problem of stability.*

## 4.1 Formally Well-Posed Problems

In Chapter 3 we obtained formal results on the number of arbitrary functions in the general solution of a differential equation. However, we could not prove that there actually exists a representation of the general solution to a certain non-negative solution of the diophantine system (3.23). In this chapter we will try to give a partial answer to this question using initial value problems. We will restrict ourselves almost always to analytic systems.

Before proving any existence and uniqueness theorem we must consider what kind of initial value problems make sense for involutive equations. The classical definition of a *well-posed* initial value problem is apparently due to Hadamard [82]. He required that the solution of the problem

(i) exists for arbitrary Cauchy data,

(ii) is determined uniquely by the Cauchy data,

(iii) depends continuously on the Cauchy data.

This is still fairly vague, as several key words are not precisely defined. We will give a formal and precise version of this definition omitting, however, the third point.

**Definition 4.1** *An initial value problem is called* formally well-posed, *if it has a unique formal power series solution for arbitrary formal power series as Cauchy data.*

One could say that in such a problem exactly the right amount of Cauchy data is prescribed. If we posed more initial conditions, we would obtain compatibility conditions for the data; with less data the solution would no longer be unique. How the solution depends on the Cauchy data is not considered.

**Corollary 4.2** *An initial value problem is formally well-posed, if and only if it is possible to construct a formal general power series solution such that there exists a bijective mapping between the parametric coefficients of the series and the Taylor coefficients of the Cauchy data.*

Obviously there arises now the question, how does one find formally well-posed initial value problems given a differential equation. In a fixed coordinate system Janet-Riquier Theory (cf. Appendix B) provides an easy method. Applying the Reid diagrams introduced in Section 2.3 one can in low-dimensional problems very simply read off correct initial prescriptions. For more complicated examples one can use an algorithm developed by Reid [124] which computes them automatically.

The idea behind this method is simple. So far we have always concentrated on the principal derivatives in our analysis. We have seen in Proposition 2.9, how one can obtain them all in a unique way by prolonging each equation with respect to its multiplicative variables only. Now we analyses in a similar way the parametric derivatives. Instead of a formal treatment we demonstrate the method with a few examples.

Before we should point out, however, that Reid's algorithm is designed for passive systems. We can nevertheless apply it to involutive equations provided we work in a $\delta$-regular coordinate system and write the system in solved form with respect to a class-respecting ranking. This connection between the two concepts is established in Proposition B.3 in Appendix B.

**Example 4.3** As first example we consider the Heat Equation $u_{xx} = u_t$. According to our ranking $u_{xx}$ must be taken as principal derivative. Thus we take $x$ as $x^2$. This leads to the diagram in Fig. 4.1. As usual the shaded area contains the principal derivatives.

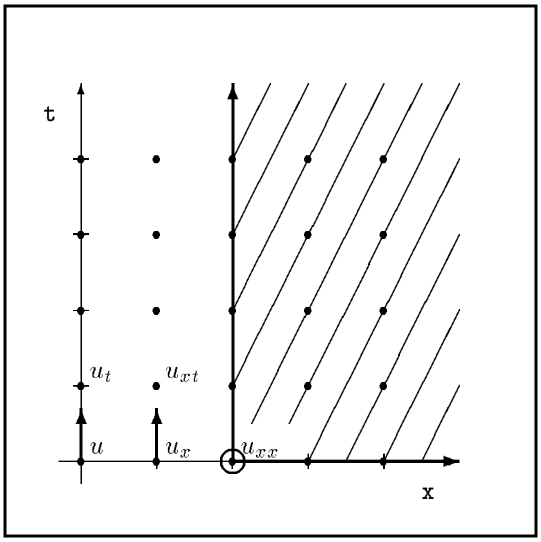


Figure 4.1: Reid diagram for the Heat Equation.

Obviously the parametric derivatives are of the form $p_{[n,0]}$ or $p_{[n,1]}$. Thus they lie all on two rays starting at $u$ and $u_x$, respectively, and we prescribe for an arbitrary value $x_0$

$$u(t, x_0) = f(t)\,, \quad u_x(t, x_0) = g(t)\,. \tag{4.1}$$

It is easy to see that there is a bijection between the Taylor coefficients of $f$ and the derivatives $p_{[n,0]}$, and the Taylor coefficients of $g$ and the derivatives $p_{[n,1]}$, respectively. Thus this initial value problem is formally well-posed.

This is, however, not the initial value problem physicists usually consider for the Heat Equation but

$$u(x, t_0) = h(x) . \tag{4.2}$$

To obtain this one we must choose $u_t$ as principal derivative, i.e. a derivative of lower order. There is a crucial difference between (4.1) and (4.2). Whereas for (4.1) Riquier's existence theorem (cf. Appendix B) ensures the convergence of the formal power series for arbitrary analytic functions $f$ and $g$, we have no such result for the standard initial value problem. Indeed it is well-known that one must pose growth conditions on $h$ to get a convergent series.

The second example considers a simple non-normal system. It shows an important fact that we will study later in more detail, namely that for such systems it is not possible to prescribe all initial data on a hypersurface of codimension 1. This might be obvious for over-determined systems, but here we consider an under-determined equation.

**Example 2.29 (cont.)** Take again the Maxwell Equation in two dimensions

$$\begin{aligned} u_{tt} - v_{xt} &= 0 , \\ u_{xt} - v_{xx} &= 0 . \end{aligned} \tag{4.3}$$

We have already seen in Section 2.6 that this equation is under-determined and that the coordinate system is $\delta$-regular. We choose in each equation the $u$-derivative as principal derivative. Fig. 4.2 contains the Reid diagram for $u$. Since we have no $v$-derivatives as principal derivatives, $v$ is unconstrained by the system and we can choose it arbitrarily.

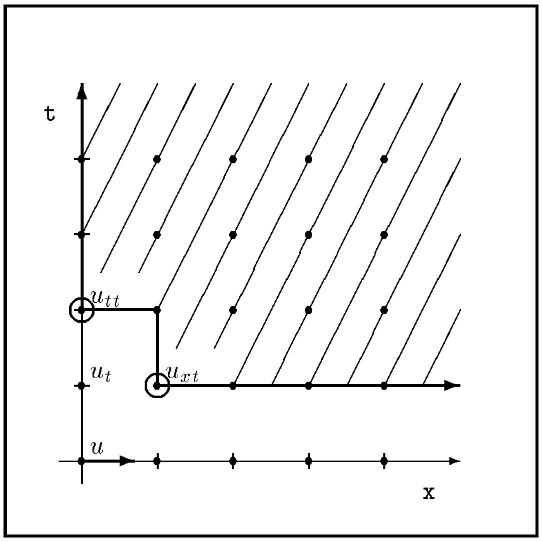


Figure 4.2: Reid diagram for the two-dimensional Maxwell Equation.

We see that in order to get all parametric derivatives, we need three initial conditions

$$u_t(x_0, t_0) = f , \quad u(x, t_0) = g(x) , \quad v(x, t) = h(x, t) . \tag{4.4}$$

Thus we prescribe some data on a zero-dimensional submanifold, the point $(x_0, t_0)$, some data on a one-dimensional submanifold, the axis $t = t_0$, and some data on the whole manifold. This is a typical feature for non-normal systems. As soon as several pivots belong to the same dependent variable, the boundary of the area with the principal derivatives will have a kind of (in general higher-dimensional) echelon form. This entails the necessity to use submanifolds of different dimensions.

Finally, we want to illustrate with a simple example one of the reasons why it is important to use a $\delta$-regular coordinate system. Besides the fact that the system might not be passive, e.g. because we gave a characteristic problem, the Janet-Riquier Theory sometimes yields “initial value problems”, one would not necessarily call so.

**Example 4.4** Consider the Wave Equation in characteristic (or light-cone) coordinates

$$u_{xy} = 0\,. \tag{4.5}$$

Obviously we are not dealing with a $\delta$-regular coordinate system, but the system is nevertheless involutive and passive. Reid's algorithm leads to the “initial value problem”

$$u(x, y_0) = f(x)\,, \quad u_y(x_0, y) = g(y)\,. \tag{4.6}$$

(One could also take $u$ in the second condition, but then one must require that $f(x_0) = g(y_0)$.) This represents, however, more a kind of boundary value problem than an initial value problem.

## 4.2 Intrinsic Description of Initial Value Problems

In the classical point of view of initial value problems Cauchy data are represented by an embedding of a submanifold of $X$ with codimension 1 in an appropriate jet bundle [56]. This also corresponds to physical intuition: the data are prescribed on a space-like manifold and the differential equation determines its evolution. Furthermore the Cauchy-Kowalevsky Theorem (see next section) shows that in the case of normal systems this leads to unique solutions.

But the example of the two-dimensional Maxwell Equations in the last section has shown that this point of view can not be upheld as soon as we leave the category of normal equations. Physicists speak often of constrained systems: if Cauchy data is nevertheless prescribed on a submanifold of codimension 1, it cannot be chosen completely arbitrarily but must satisfy “constraints” (we will discuss this approach in Chapter 6 in more detail).

Janet-Riquier Theory shows how to avoid these constraints: initial data is prescribed on a flag of submanifolds instead of just one. Although this point of view is implicit in the Cartan-Kähler analysis of exterior systems, we know of only two articles [87, 88] (by the same author) where it has been used explicitly in a geometric approach to differential equations. There it was, however, always assumed that the problem is formally well-posed and no analysis of the conditions for this to be the case is given. We will give in the sequel an intrinsic description of these conditions using the $k$-tableaux. Simply said, we will show how to pick principal derivatives in a geometric way independent of a coordinate system.

The method to construct formally well-posed initial value problems hinted in the last section is very general on one side. By using the notion of a passive system one can treat on the same footing also characteristic problems, problems like (4.2) where one chooses derivatives of lower order as principal ones, and many more. The particularities of these problems are caught by the ranking used and different rankings will lead to different passive systems. On the other hand this method depends crucially on the used coordinate system. We prescribe the Cauchy data always on coordinate hyperplanes and rankings can be defined only with respect to a given coordinate system.

We will restrict our attention for most of the remainder of this chapter on first order equations. We will further assume that no lower order, i.e. algebraic, equations are present. They could always be solved for some of the dependent variables and thus used to eliminate them. These assumptions are mainly for notational simplicity. We know from Appendix C.2 that any equation can be transformed into an equivalent first order equation, thus there is no loss of generality.

Let $\chi^{(k)}, k = 1, \dots, n$ be a non-systatic basis of $T^*X$ as defined in Section 2.7. We associate with it two flags of submanifolds. Let $X^k \subset X$ be a one-dimensional submanifold ("coordinate axis") such that $T^*X^k = < \chi^{(k)} >$, then we introduce $N_k = X^1 \times \cdots \times X^{k-1}$ ($N_1 = \{x_0\}\}$ for an arbitrary but fixed point $x_0 \in X$) and $N^k = X^k \times \cdots \times X^n$. The $N^k$ have already been used in the definition of the $k$-tableaux. We call a flag such as $N_k$ a *non-systatic flag of submanifolds.*

Since we consider only first order equations, the $k$-tableau is given by a mapping from $T^*N^k \otimes V\mathcal{E}$ to $\mathcal{E}'$. We search now for a flag of fibered submanifolds $M^k$ of $\mathcal{E}$ such that $\emptyset \subseteq M^1 \subseteq \cdots \subseteq M^n \subseteq M^{n+1} = \mathcal{E}$ and

$$\operatorname{rank} \tau_{[\chi^{(k)},\dots,\chi^{(n)}]}\Big|_{T^*X^k \otimes VM^k} = \beta_q^{(k)} . \tag{4.7}$$

The subset condition reflects the fact that in a $\delta$-regular coordinate system not the individual values $\beta_q^{(k)}$ become maximal but the sums $\sum_{i=k}^n \beta_q^{(i)}$.

**Lemma 4.5** *Such a flag $M^k$ exists for every involutive first-order equation $\mathcal{R}_1$. Furthermore one can choose $M^k$ such that* $\dim M^k = \beta_1^{(k)}$ *(with $\beta_1^{(n+1)} = m$).*

Proof. We choose a coordinate system associated with the $\chi^{(k)}$. Let $(x_0, u_0) \in \mathcal{E}$ be an arbitrary but fixed point. In a neighborhood of it we can describe the $N^k$ by the equations $x^i = x_0^i$ for $i = 1, \dots, k-1$ and we claim that the fibered submanifolds given by the equations $u^\alpha = u_0^\alpha$ for $\alpha = \beta_1^{(k)} + 1, \dots, m$ can be taken as $M^k$.

Proposition 2.12 implies that if we choose the principal derivatives according to the inverse lexicographic ranking, then with $p_k^\alpha$ are also all $p_i^\alpha$ with $i > k$ principal derivatives. By the Implicit Function Theorem there exists a local description of $\mathcal{R}_1$ of the form

$$\Phi_k^\alpha(x, u, p) = p_k^\alpha - \varphi_k^\alpha(x^i, u^\beta, p_j^\gamma) = 0\,, \qquad \begin{cases} \alpha = 1,\dots,\beta_1^{(k)}\,, \\ j = 1,\dots,k\,, \\ \gamma = \beta_1^{(j)}+1,\dots,m\,. \end{cases} \tag{4.8}$$

Thus there exist principal derivatives of class $k$ only for $1 \leq \alpha \leq \beta_1^{(k)}$. But restricting the $k$-tableau to $T^*X^k \otimes VM^k$ means nothing else but taking only the corresponding equations into account for its construction. Hence we get for its rank $\beta_1^{(k)}$. The $M^k$ define indeed a flag due to Corollary 2.13 and have obviously the correct dimensions. •

From (4.8) it is easy to read off a formally well-posed initial value problem:

$$\begin{aligned} &u^\alpha(x_0) = f^\alpha \ (= Const.)\,, \qquad \alpha = 1, \dots, \beta_1^{(1)}\,, \\ &u^\alpha(x^1, \dots, x^k, x_0^{k+1}, \dots, x_0^n) = f^\alpha(x^1, \dots, x^k)\,, \\ &\qquad\qquad\qquad\qquad \alpha = \beta_1^{(k)} + 1, \dots, \beta_1^{(k+1)}\,. \end{aligned} \tag{4.9}$$

In order to give an intrinsic description of these initial conditions, we introduce the fibered submanifolds $M_k$ by $M^k = M_1 \times \cdots \times M_k$. Thus $\dim M_k = \beta_1^{(k)} - \beta_1^{(k-1)}$. Let us denote by $\mu_k$ the natural projection from $\mathcal{E}$ to $M_k$[1]. Instead of the functions $f^\alpha$ we use now fiber

[1] One should perhaps remark that in general $M_k$ is *not* a subbundle of $\mathcal{E}$. This will, however, always be the case, if $\mathcal{E}$ is trivial. In principle it makes no sense at all to speak about global properties of the $M_k$, as they exist in general only locally. Furthermore it may happen that the values $\beta_q^{(k)}$ vary on $\mathcal{R}_q$; then one must restrict the definition of the $M_k$ to a neighborhood with constant values.

preserving mappings $\rho_k : N_k \to M_k$. If the section $\sigma \in \Gamma_{loc}(\mathcal{E})$ is a local solution of $\mathcal{R}_1$, then (4.9) can be written as

$$\mu_k(\sigma)\Big|_{N_k} = \rho_k\,. \tag{4.10}$$

**Proposition 4.6** *$\mathcal{R}_1$ together with (4.10) is a formally well-posed initial value problem.*

**Example 2.29 (cont.)** To make this fairly awkward construction more transparent, we detail the involved submanifolds for the two-dimensional Maxwell Equation. In order to rewrite it as a first-order system, we introduce the additional dependent variables $w^2 = u_t, w^1 = u_x$ and $z^2 = v_t, z^1 = v_x$

$$\begin{aligned} &u_t - w^2 = 0\,, && u_x - w^1 = 0\,,\\ &v_t - z^2 = 0\,, && v_x - z^1 = 0\,,\\ &w^2_t - z^1_t = 0\,, && w^2_x - z^1_x = 0\,,\\ &w^1_t - w^2_x = 0\,, &&\\ &z^1_t - z^2_x = 0\,. && \end{aligned} \tag{4.11}$$

This yields $\beta_1^{(2)} = 5$ (left column) and $\beta_1^{(1)} = 3$ (right column). We take in each equation the first term as principal derivative. A non-systatic basis of $T^*X$ is given by $\chi^{(1)} = dx, \chi^{(2)} = dt$.

We can now choose the submanifolds $N_3 = X$, $N_2 = \{t = t_0\}$, $N_1 = \{(x_0, t_0)\}$ and $M^3 = \mathcal{E}$, $M^2 = \{z^2 = z^2_0\}$, $M^1 = \{z^2 = z^2_0, z^1 = z^1_0, w^1 = w^1_0\}$. This can be seen as follows: The system (4.11) has been set up in such a way that it is preferable to choose $x^2 = t, x^1 = x$. Hence the values of the $N_k$. $M^2$ comes from the fact that there are no principal derivatives involving $z^2$. For all other dependent variables, we have a principal derivative of class 2. To get $M^1$ we observe that only for $u, v, w^2$ principal derivatives of class 1 occur.

Summarizing we get the initial value problem

$$\begin{aligned} &z^2(x,t) = f_1(x,t)\,, && z^1(x,t_0) = f_2(x)\,, && w^1(x,t_0) = f_3(x)\,,\\ &w^2(x_0,t_0) = f_4\,, && v(x_0,t_0) = f_5\,, && u(x_0,t_0) = f_6\,. \end{aligned} \tag{4.12}$$

The connection with (4.4) is given by

$$f = f_4\,, \quad g(x) = \int_{x_0}^{x} f_3(\rho)d\rho + f_6\,, \quad h(x,y) = \int_{t_0}^{t} f_1(x,\tau)d\tau + \int_{x_0}^{x} f_2(\rho)d\rho + f_5\,. \tag{4.13}$$

The initial conditions (4.10) are sufficient to select a unique formal power series solution. The natural question now is under what conditions does this formal series converge, i.e. we should look for a strong existence theorem. For general equations this is only possible, if everything (equations, Cauchy data, solution) is analytic. This will be the topic of the next section. In Section 4.4 we will extend the uniqueness also to non-analytic solutions, however only for linear equations.

**Example 4.7** Getting rid of the restriction to analytic equations is much harder due to the existence of smooth involutive equations without solutions. Lewy [96] found the first example of such a system. He showed that there exists a $\mathcal{C}^\infty$ function $F$ such that the linear(!), normal system

$$\begin{aligned} u_x - v_y + 2yu_z + 2xv_z &= F(x,y,z)\,,\\ v_x + u_y - 2xu_z + 2yv_z &= 0 \end{aligned} \tag{4.14}$$

has no $\mathcal{C}^\infty$ solution (actually not even a $\mathcal{C}^1$ solution).

## 4.3 Cartan-Kähler Theorem

In this section we will consider only analytic equations and we will look only for analytic solutions. This implies that the Cauchy data as well as the submanifolds on which they are prescribed are also analytic. The "mother of all existence theorems" in this category is the well-known Cauchy-Kowalevsky Theorem which covers normal equations.

**Theorem 4.8 (Cauchy-Kowalevsky)** *Let $\phi^\alpha(x^i, u^\beta, p^\beta_j)$ and $f^\alpha(x^1, \dots, x^{n-1})$ where $\alpha, \tau = 1, \dots, m$, $i = 1, \dots, n$, $j = 1, \dots, n-1$ be (real) analytic functions of all their arguments in a neighborhood of the origin. Then the initial value problem*

$$\begin{aligned} p^\alpha_n &= \phi^\alpha(x^i, u^\beta, p^\beta_j)\,, \\ u^\alpha(x^1, \dots, x^{n-1}, 0) &= f^\alpha(x^1, \dots, x^{n-1})\,, \end{aligned} \tag{4.15}$$

*has a unique solution that is analytic in a neighborhood of the origin of $X$. Unique meaning that no other analytic solution exists.*

Since the proof of this theorem can be found in almost any text book on partial differential equations (e.g. [82, 116]), we omit it here. Most books consider only quasi-linear equations. One constructs a power series solution along the lines described in Section 2.5 and proves then (this is the hard part) its convergence by the method of majorants. One can easily reduce the non-linear case to the quasi-linear one by differentiating (4.15) with respect to $x^n$ and introducing the first order derivatives $u^\alpha_{x^i}$ as new dependent variables $p^\alpha_i$ [33]. This leads to the quasi-linear system

$$\begin{aligned} u^\alpha_{x^n} &= p^\alpha_n\,, \\ p^\alpha_{j,x^n} &= p^\alpha_{n,x^j}\,, \qquad\qquad j = 1, \dots, n-1\,, \\ p^\alpha_{n,x^n} &= \Phi^\tau_{x^n} + \sum_{\beta=1}^{m} \Phi^\tau_{u^\beta} p^\beta_n + \sum_{\beta=1}^{m}\sum_{j=1}^{n-1} \Phi^\tau_{p^\beta_j} p^\beta_{n,x^j}\,, \end{aligned} \tag{4.16}$$

which is obviously again in the form required by Theorem 4.8. If we augment it with the initial conditions

$$\begin{aligned} u^\alpha(x^1, \dots, x^{n-1}, 0) &= f^\alpha(x^1, \dots, x^{n-1})\,, \\ p^\alpha_j(x^1, \dots, x^{n-1}, 0) &= \partial_{x^j} f^\alpha(x^1, \dots, x^{n-1})\,, \\ p^\alpha_n(x^1, \dots, x^{n-1}, 0) &= \Phi^\tau(x^1, \dots, x^{n-1}, 0, f^\beta, \partial_{x^j} f^\beta)\,, \end{aligned} \tag{4.17}$$

then every solution of it yields a solution of (4.15) and vice versa.

We can compare the results of Theorem 4.8 and Theorem 3.9. Since for normal equations $\beta^{(n)}_1 = m, \beta^{(n-1)}_1 = \cdots = \beta^{(1)}_1 = 0$, we obtain $f_{n-1} = m, f_n = f_{n-2} = \cdots = f_1 = 0$ as solution of the recursion relation (3.16) according to Corollary 3.10 in perfect agreement with (4.15) where $m$ functions of $n-1$ variables are prescribed. Note, however, the crucial difference between the two theorems: Theorem 3.9 makes no statement about the existence or uniqueness of solutions.

The Cauchy-Kowalevsky Theorem can be extended from normal equations to arbitrary involutive ones. This generalization is usually known as Cartan-Kähler Theorem. As the name already indicates, it stems originally from the Cartan-Kähler Theory of exterior differential systems (cf. Section A). A first non-constructive proof within the framework of formal theory was given by Goldschmidt [58] using the Spencer cohomology.

We will give here an adaption of the original proof successively applying the Cauchy-Kowalevsky Theorem, as this demonstrates further interesting properties of involutive equations. It will also allow us to generalize other classical results like the Holmgren Theorem (cf. Section 4.4).

**Theorem 4.9 (Cartan-Kähler)** *Let the differential equation $\mathcal{R}_1$ be analytic and involutive. There exists one and only one analytic solution $\sigma$ of $\mathcal{R}_1$ satisfying the initial conditions (4.10) for analytic mappings $\rho_k$ and analytic flags $M_k, N_k$.*

Proof. We will use the local formulation of this initial value problem given by (4.8,4.9) and construct the solution step by step. In the first step, we treat only the equations of class 1:

$$p_1^\alpha = \varphi_1^\alpha(x^i, u^\beta, p_1^\gamma)\,, \qquad \begin{cases} \alpha = 1,\dots,\beta_1^{(1)}\,, \\ \gamma = \beta_1^{(1)}+1,\dots,m\,. \end{cases} \tag{4.18}$$

If we evaluate (4.18) at $x^j = x_0^j$ for $j > 1$, we can substitute $u^\beta$ for $\beta > \beta_1^{(1)}$ by the $f^\beta$ (restricted to $N_1$) and we obtain a normal system with dependent variables $u^\alpha$ for $\alpha = 1, \dots, \beta_1^{(1)}$, independent variable $x^1$ and initial conditions $u^\alpha(0) = f^\alpha$. The Cauchy-Kowalevsky Theorem yields thus a unique analytic solution $U^\alpha(x^1, x_0^2, \dots, x_0^n)$ satisfying $U^\alpha(x_0) = f^\alpha$.

The next step takes the equations of class 2 to extend this one-dimensional solution to a two-dimensional one $U^\alpha(x^1, x^2, x_0^3, \dots, x_0^n)$ applying again the Cauchy-Kowalevsky Theorem after the corresponding substitution. This inductive process continues till class $n$. We omit the details and go directly to step $n$. The other steps run similarly.

We suppose that we know a solution $U^\alpha(x^1, \dots, x^{n-1}, x_0^n)$ for all equations up to class $n-1$. The equations of class $n$ are

$$p_n^\alpha = \varphi_n^\alpha(x^i, u^\beta, p_j^\gamma)\,, \qquad \begin{cases} \alpha = 1,\dots,\beta_1^{(n)} \\ \gamma = \beta_1^{(j)},\dots,m\,, \\ j = 1,\dots,n\,, \end{cases} \tag{4.19}$$

and they form a normal system, this time with dependent variables $u^\alpha$ for $\alpha = 1, \dots, \beta_1^{(n)}$ and independent variables $x^1, \dots, x^n$. As initial conditions we take

$$u^\alpha(x^1, \dots, x^{n-1}, x_0^n) = \begin{cases} U^\alpha(x^1, \dots, x^{n-1}, x_0^n) & \text{if } 1 \le \alpha \le \beta_1^{(n-1)}\,, \\ f^\alpha(x^1, \dots, x^{n-1}) & \text{if } \beta_1^{(n-1)} + 1 \le \alpha \le \beta_1^{(n)}\,. \end{cases} \tag{4.20}$$

Since these functions are all analytic, we can invoke the Cauchy-Kowalevsky Theorem. The solution of the initial value problem yields the wanted functions $U^\alpha(x)$. $u^\alpha$ for $\alpha > \beta_1^{(n)}$ is unconstrained by the system (4.8) and directly given by the initial conditions (4.9).

We must, however, still show that these functions $U^\alpha(x)$ also solve the full equations of lower class, for in the previous steps we have always treated some independent variables as constants. That is where involution proves to be crucial. If the functions $U^\alpha$ satisfy all equations of system (4.8), then the functions

$$\Delta_i^\alpha(x) = \Phi_i^\alpha(j_1(U^\beta)(x)) \tag{4.21}$$

must vanish in the considered neighborhood of $x_0$ (this is a formal way to express the substitutions $u^\alpha = U^\alpha(x)$, $p_i^\alpha = \partial U^\alpha(x)/\partial x^i$).

Since system (4.8) is involutive, the prolongation of an equation with respect to a non-multiplicative variable must lead to a linear combination of equations obtained by differentiating all equations with respect to their multiplicative variables only (Proposition 2.9, cf. also the discussion following Equation (2.16)). Thus we have for $i < n$ (among others) the following relations between the prolonged equations

$$D_n\Phi_i^\alpha = \sum_{\substack{1\leq k\leq n, 1\leq l\leq k\\ 1\leq\beta\leq\beta_1^{(k)}}} A_{l\beta}^k(x,u,p)D_l\Phi_k^\beta + \sum_{k,\beta} B_\beta^k(x,u,p)\Phi_k^\beta\,, \tag{4.22}$$

as we differentiate on the left hand side with respect to a non-multiplicative variable. The ranges of the indices on the right hand side are so chosen, that only prolongations with respect to multiplicative variables are taken into account.

If we substitute the functions $U^\alpha$ and their derivatives for the jet variables $u^\alpha, p_i^\alpha, \ldots$, we can easily evaluate the formal derivatives

$$(D_j\Phi_i^\alpha)(j_2(U^\alpha)(x)) = \frac{\partial}{\partial x^j}\Phi_i^\alpha(j_1(U^\alpha)(x))\,. \tag{4.23}$$

Now (4.22) becomes a normal system

$$\frac{\partial}{\partial x^n}\Delta_i^\alpha = \sum A_{l\beta}^k\frac{\partial}{\partial x^l}\Delta_k^\beta + \sum B_\beta^k\Delta_k^\beta \tag{4.24}$$

and according to our induction hypothesis that $U^\alpha(x^1,\ldots,x^{n-1},x_0^n)$ satisfy the equations up to class $n-1$, if $x^n = x_0^n$, we get the initial conditions

$$\Delta_j^\beta(x^1,\ldots,x^{n-1},x_0^n) = 0\,. \tag{4.25}$$

Applying for the last time the Cauchy-Kowalevsky Theorem, we get that this system has one and only one analytic solution which is trivially $\Delta_j^\beta(x)\equiv 0$. •

One should note that formal integrability is not sufficient for this proof. The first part, i.e. the step by step construction of $U^\alpha(x^1,\ldots,x^n)$ could of course be done for any differential equation even for a not formally integrable one. It is, however, unclear for such a system, whether the so obtained $U^\alpha$ also solve the equations of lower class. Only involution provides the key to set up the final normal system (4.24). Nevertheless, we can conclude that every analytic formally integrable system has an analytic solution, as it will become involutive after a finite number of prolongation.

Again it is instructive to compare this theorem with the results of Chapter 3. The Cauchy data prescribed in (4.9) consist of $f_n = m - \beta_1^{(n)}$ functions of $n$ variables and $f_k = \beta_1^{(k+1)} - \beta_1^{(k)}$ functions of $k$ variables. These are exactly the values predicted by Corollary 3.10 for an algebraic representation of the general solution in the case of a first-order equation.

If we start with an equation $\mathcal{R}_q$ of order $q$, it can be reduced by one of the two methods outlined in Appendix C.2 to a first-order equation. Directly reducing $\mathcal{R}_q$ leads by (C.21) to the prescription of $\alpha_q^{(k)} - \alpha_q^{(k+1)}$ functions of $k$ arguments. This corresponds exactly to the values computed in (3.27) for a representation with $(q-1)$-dimensional integrals. If we reduce $\mathcal{R}_{q+1}$, we prescribe according to (C.23) $\alpha_q^{(k)}$ functions of $k$ arguments. Comparing with (3.26) we see that this corresponds to a representation with $q$-dimensional integrals. These integrations occur of course during the transformation of a solution of the reduced equation into one of the original equation $\mathcal{R}_q$.

**Example 4.10** A simple example [17] may clarify this construction and the importance of involution. We consider a first-order system with two independent and three dependent variables of the form

$$\begin{aligned} u^1_x &= F(x,y,u^1,u^2,u^3)\,, \\ u^1_y &= G(x,y,u^1,u^2,u^3,u^1_x,u^2_x,u^3_x)\,, \\ u^2_y &= H(x,y,u^1,u^2,u^3,u^1_x,u^2_x,u^3_x) \end{aligned} \tag{4.26}$$

with the initial conditions

$$u^1(0,0)=f^1\,,\;\; u^2(x,0)=f^2(x),\;\; u^3(x,y)=f^3(x,y)\,. \tag{4.27}$$

We have two equations of class 2 and one of class 1. We begin with the latter one. This leads to the initial value problem

$$u^1_x = F\big(x,0,u^1,f^2(x),f^3(x,0)\big)\,,\quad u^1(0,0)=f^1 \tag{4.28}$$

which has the unique solution $U^1(x,0)$. In the next step, we consider the problem

$$\begin{aligned} u^1_y &= G\big(x,y,u^1,u^2,f^3(x,y),u^1_x,u^2_x,f^3_x(x,y)\big)\,, \\ u^2_y &= H\big(x,y,u^1,u^2,f^3(x,y),u^1_x,u^2_x,f^3_x(x,y)\big)\,, \\ u^1(x,0) &= U^1(x,0), \quad u^2(x,0)=f^2(x)\,. \end{aligned} \tag{4.29}$$

Its unique solution be the two functions $U^1(x,y), U^2(x,y)$. $U^3(x,y)$ is of course simply $f^3(x,y)$.

But now we must check whether $U^1,U^2,U^3$ also solve the first equation. We define

$$\Delta(x,y) = U^1_x(x,y) - F\big(x,y,U^1(x,y),U^2(x,y),U^3(x,y)\big)\,. \tag{4.30}$$

Because of the first step $\Delta(x,0)=0$. Furthermore,

$$\begin{aligned} \Delta_y = \; & G_{u^1}\Delta + \\ & \Big[G_x + \textstyle\sum\big(G_{u^\alpha}u^\alpha_x + G_{u^\alpha_x}u^\alpha_{xx}\big) - F_y - \sum F_{u^\alpha}u^\alpha_y\Big]\Big|_{u^\alpha=U^\alpha}\,. \end{aligned} \tag{4.31}$$

The square bracket contains the integrability condition obtained by cross-differentiating the first and the second equation of (4.26). If we assume that the system is involutive, it must vanish. Thus $\Delta$ satisfies

$$\Delta_y = G_{u^1}\Delta\,,\quad \Delta(x,0)=0\,. \tag{4.32}$$

By the usual uniqueness theorem for ordinary differential equations, it follows that $\Delta$ vanishes everywhere.

## 4.4 Holmgren Theorem

In this section we will try to extend the uniqueness to non-analytic solutions. Using a technique apparently introduced by Holmgren [77] this is fairly easily possible for linear systems. Before we get to the proof of Holmgren's theorem, we have to introduce a few facts about linear differential operators. We will again restrict without loss of generality to first order equations. Let $L$ be a linear first order differential operator

$$Lu = \sum_{i,\alpha} A^{i\tau}_\alpha(x)p^\alpha_i + \sum_\alpha B^\tau_\alpha(x)u^\alpha\,. \tag{4.33}$$

Its *adjoint operator* $\tilde{L}$ is given by

$$\tilde{L}v = -\sum_{\tau,i}\partial_{x^i}\Big[((A^i)^T)^\alpha_\tau(x)v^\tau\Big] + \sum_\tau (B^T)^\alpha_\tau(x)v^\tau\,. \tag{4.34}$$

(Note that we do not require that the matrices $A^i$ and $B$ are square.) Obviously $\tilde{L}$ is normal, if and only if $L$ is normal.

The Lagrange-Green Identity [82] asserts that for vector valued $\mathcal{C}^1$ functions $u$ and $v$

$$\int_\Omega v^T L u \, dV = \int_\Omega (\tilde{L} v)^T u \, dV + \int_{\partial\Omega} v^T A u \, dS \,, \tag{4.35}$$

where $\partial\Omega$ denotes the boundary of a region $\Omega \subset X$ and $A = \sum A^i(x)\zeta_i$ with the exterior unit normal $\zeta$ of $\partial\Omega$. The notion of an adjoint operator and the Lagrange-Green identity can readily be extended to higher order operators. The surface integral in (4.35) becomes then more complicated and its form is no longer unique but depends on the order in which one performs the partial integrations needed to prove (4.35).

**Theorem 4.11 (Holmgren)** *Let $L$ be a normal linear differential operator with analytic coefficients. Let $M \subset X$ be a non-characteristic submanifold of dimension $n-1$. Any $\mathcal{C}^1$ solution of the initial value problem*

$$\begin{aligned} Lu &= 0 \,, \\ u\big|_M &= 0 \end{aligned} \tag{4.36}$$

*vanishes in a neighborhood of $M$.*

Proof. This is the classical theorem as given by Holmgren [77]. We will follow the proof outlined in [33]. The linearity of $L$ is now of course also a question of the coordinate system. We will assume in the sequel always that all considered regions are so small that they can be covered by one $\delta$-regular chart and that in it $L$ is linear.

Let $U(x_0) \subset M$ be a neighborhood of a point $x_0 \in M$. We consider a region $\Omega \subset X$ bounded by $U(x_0)$ and another $(n-1)$-dimensional submanifold $N$. The initial value problem

$$\begin{aligned} \tilde{L} v &= p \,, \\ v\big|_N &= 0 \,, \end{aligned} \tag{4.37}$$

where the components of $p$ are polynomials in the chosen coordinate system, possesses by the Cauchy-Kowalevsky Theorem a solution defined in $\Omega$, if $\Omega$ is chosen sufficiently small.

Applying the Lagrange-Green Identity (4.35) yields

$$\int_\Omega p^T u \, dV = 0 \,. \tag{4.38}$$

Since this must hold for arbitrary $p$ and the polynomials lie dense in $\mathcal{C}^1$, this implies that $u \equiv 0$ in $\Omega$ as claimed. •

This proof is at one point a bit weak. We assumed tacitly that we can find a finite region $\Omega$ where a solution of (4.37) exists for all $p$ with polynomial components. One can remedy this and at the same time extend Theorem 4.11 to larger regions by introducing the notion of an *analytic field* $S_\lambda$, i.e. a family of submanifolds in co-dimension 1 which sweeps out a region $\Omega \subset X$ for $\lambda \in [a, b] \subset \mathbb{R}$ [82]. Such a region $\Omega$ is then called a *domain of determinacy* for $M$.

**Theorem 4.12** *Let $L$ be an involutive linear differential operator with analytic coefficients. Let $x_0 = N_0 \subset N_1 \subset \cdots \subset N_n = X$ be a non-systatic flag of submanifolds with* $\dim N_i = i$. *Then any $\mathcal{C}^1$ solution of the initial value problem*

$$\begin{aligned} &Lu = 0 \,, \\ u^\alpha\big|_{N_k} = 0 \,, \qquad &\alpha = \beta_1^{(k)} + 1, \ldots, \beta_1^{(k+1)} \end{aligned} \tag{4.39}$$

*vanishes in a neighborhood of $x_0$.*

PROOF. This extension to a more general situation is straightforward considering the proof we gave for the Cartan-Kähler Theorem. Let $u$ be any solution of the considered initial value problem. Then $u|_{N_1}$ must satisfy (4.18) with vanishing Cauchy data. Thus according to Theorem 4.11 $u|_{N_1} \equiv 0$. Similarly we can conclude following step by step the construction in the proof of Theorem 4.9 that $u|_{N_k} \equiv 0$ for $k = 1, \dots, n$. Each of these intermediate initial value problems will have vanishing Cauchy data as a consequence of the previous step and thus Holmgren's Theorem is throughout applicable. •

## 4.5 Stability

To conclude this chapter on initial value problems we make a few remarks on the problem of stability being omitted in our definition of a formally well-posed problem. It is especially of importance for numerical calculations, for they make sense only if the solutions of problems with "similar" Cauchy data are "similar". We have constructed in Section 4.2 a formally well-posed problem for any involutive differential equation. But this problem will be unstable for many equations.

**Example 4.13** The classical example of an ill-posed problem is the Cauchy problem for the Laplace Equation

$$u_{xx} + u_{yy} = 0\,. \tag{4.40}$$

Hadamard [33] was the first to show that the solution of the following sequence of initial value problems does not depend continuously on the Cauchy data:

$$u(x,0) = 0\,, \quad u_y(x,0) = f_n(x) = \frac{1}{n^2}\sin(nx)\,. \tag{4.41}$$

Obviously each problem is formally well-posed; the formal power series solutions even converge, as the solution is given by

$$u_n(x,y) = \frac{1}{n^2}\sin(nx)\sinh(ny)\,. \tag{4.42}$$

$f_n(x) \to 0$ for $n \to \infty$, but the solutions $u_n(x,y)$ do not converge to the solution of the initial value problem with $u_y(x,0) = 0$ which is of course $u(x,y) = 0$. Thus the larger $y$ the larger the difference between two solutions with arbitrarily close Cauchy data becomes.

The question is how to define rigorously when the Cauchy data of two different initial value problems are close or similar. One must distinguish here between two possibilities to deform Cauchy data. In numerical analysis one considers usually the situation found in Example 4.13: how is the solution affected by a slight variation in the functions prescribed. But in a geometric formulation like (4.10) one must also take deformations of the submanifolds on which the data is given into account.

Shih used in his thesis [141] the fact that it is possible to define topologies on the space of embeddings [59] to introduce a topology on the space of all Cauchy data for a given equation. He defined a problem to be well-defined, if it is an interior point of the subspace of problems having at least one solution. Thus "small" is defined to mean that there exist an open neighborhood in the used topology such that every problem in it is well-posed.

This approach tackles the second possibility of deformation while neglecting the first one. It considers the Laplace Equation as well-posed, for it only requires that the "neighboring" initial value problems have solutions. But it does not require any relation between their solutions. Shih's formulation is restricted to the standard problem where all data is prescribed on a submanifold of codimension 1. But it should be straightforward to extend it to our more general formulation.

Restricting ourselves to this notion of stability, we can easily prove that our formulation is stable under deformation of the flag $N_k$. As described in Section 4.2 these nested submanifolds are characterized by a non-systatic basis of $T^*X$. Thus to study a deformation of them it suffice to analyze deformations of such bases. This avoids the use of complicated topologies, since we can use the standard topology of $\mathbb{R}^n$.

**Proposition 4.14** *A non-systatic basis of $T^*X$ is stable under deformations.*

Proof. In a given coordinate system the conditions for a basis to be non-systatic can be expressed by requiring that certain subdeterminants of the $k$-tableaux do not vanish. But this implies that systatic bases can be characterized by the vanishing of certain rational functions of the coefficients of the forms. Thus we can define a homeomorphic mapping $f$ from the space of all bases of $T^*X$ into an $\mathbb{R}^p$ such that $f^{-1}(0)$ gives all systatic bases. But the image of a closed set is again closed, thus the complement of $f^{-1}(0)$, the space of all non-systatic bases is open. Since transition functions are smooth, this statement holds independently of the particular coordinate system used. •

# Chapter 5

# Symmetries and Reduction

*After a brief introduction into symmetries of differential equations and different reduction methods, we will show how the formal theory can be applied to some problems arising in this context. Especially we will consider the loss of generality through such a reduction and study the question, how one can find "good" differential constraints.*

## 5.1 Introduction

Some of the most important geometric techniques for studying differential equations belong to the realm of symmetry analysis. It was originally developed at the end of the last century by Lie searching for a "Galois Theory" of differential equations. Meanwhile it is nearly impossible to survey the literature on this subject. We refer only to the basic textbooks [15, 111, 146].

**Definition 5.1** *A* symmetry *of a differential equation is a transformation that maps solutions into solutions.*

This is the most general way to introduce the notion of a symmetry. It is, however, fairly vague, as it does not specify what kind of transformations are considered. The geometric theory provides a natural setting for specifying this in more detail. We will restrict our attention here to continuous point transformations, i.e. to diffeomorphism of the manifold $\mathcal{E}$ of independent and dependent variables. There exist many generalizations. One can e.g. add discrete transformations, one can consider generalized or Lie-Bäcklund symmetries or diffeomorphisms on covering manifolds leading to non-local symmetries.

**Definition 5.2** *A diffeomorphism $\phi : \mathcal{E} \mapsto \mathcal{E}$ is called a* Lie point symmetry *of the differential equation $\mathcal{R}_q$, if $\phi^*\mathcal{R}_q = \mathcal{R}_q$.*

There arises now of course the (often over-looked) question to what extend these two definitions are equivalent. Even if we restrict "transformation" in Definition 5.1 to the diffeomorphisms used in Definition 5.2, the first one yields in general more symmetries than the second one.

**Example 2.3 (cont.)** Consider again the differential equation given by

$$\mathcal{R}_1 : \begin{cases} u_z + yu_x &= 0\,, \\ u_y &= 0\,. \end{cases}$$

$\mathcal{R}_1$ is obviously not invariant under $y$-translations, as $y$ occurs explicitly. Adding the integrability condition $u_x = 0$ leads to the trivial system $u_x = u_y = u_z = 0$ which is invariant under $y$-translation and so is of course every solution of $\mathcal{R}_1$. This symmetry was lost, because it was requested that the manifold $\mathcal{R}_1$ should remain invariant instead of the smaller one $\mathcal{R}_1^{(1)}$.

If a differential equation is not formally integrable, then it is obvious from this example that Definition 5.2 poses a stricter condition than Definition 5.1. There are points in $\mathcal{R}_q$ which are never reached by a solution, i.e. let $\Theta \subset \Gamma_{loc}(\mathcal{E})$ be the solution space of $\mathcal{R}_q$, then the union of the prolongations of all sections in $\Theta$ is in general only a subset of $\mathcal{R}_q$

$$\bigcup_{\sigma\in\Theta} j_q(\sigma) \subseteq \mathcal{R}_q\,. \tag{5.1}$$

One can show that for differential equations where the equal sign holds in (5.1) the two definitions of a symmetry are equivalent [111].

**Definition 5.3** *A differential equation $\mathcal{R}_q$ is called* locally solvable*, if there exists for every point $P \in \mathcal{R}_q$ a solution $\sigma$ such that $P \in j_q(\sigma)$, i.e. every point lies on the prolongation of a solution.*

Note that we do not require uniqueness of this solution. But this is obvious, as the prescribed data are of course in general not sufficient. Local solvability becomes a difficult problem, if we consider only smooth equations or solutions, as we have already seen in the Example 4.7 of Lewy. In the analytic category, we get it as a simple corollary of the Cartan-Kähler Theorem.

**Corollary 5.4** *Every analytic and involutive differential equation $\mathcal{R}_q$ is locally solvable.*

Proof. Let $P = (x_0^i, u_0^\alpha, p_{\mu,0}^\alpha)$ be a point on $\mathcal{R}_q$. We can choose for the construction of a power series solution the principal derivatives according to the inverse lexicographic ranking. $P$ prescribes values for all parametric derivatives of order less than or equal to $q$ at the point $x_0 \in X$. If we take all other parametric derivatives as zero, we obtain an initial value problem with polynomial Cauchy data for which the Cartan-Kähler Theorem applies. Hence this problem has a unique analytic solution $f$ and by construction $P \in j_q(f)$.

It is of course not the only possibility to take all other parametric derivatives as zero. Any convergent Taylor series with the same coefficients up to order $q$ will do. Thus we get a unique solution through $P$ only for an equation of finite type, where the only free parameters are the parametric derivatives of order less than $q$. •

Determining all diffeomorphisms satisfying the condition in Definition 5.2 is a difficult task using finite transformations. It was Lie's great insight to discover that it suffices to study infinitesimal transformations. Let

$$\mathbf{v} = \sum_{i=1}^{n} \zeta^i(x,u)\frac{\partial}{\partial x^i} + \sum_{\alpha=1}^{m} \eta^\alpha(x,u)\frac{\partial}{\partial u^\alpha} \tag{5.2}$$

be a vector field on $\mathcal{E}$. Using the chain rule it is straight-forward to define its prolongation $\mathrm{pr}^{(q)}\mathbf{v}$, a vector field acting on $J_q\mathcal{E}$, by

$$\mathrm{pr}^{(q)}\mathbf{v} = \mathbf{v} + \sum_{\alpha,|\mu|\leq q} \eta_\mu^\alpha \frac{\partial}{\partial p_\mu^\alpha} \tag{5.3}$$

where the coefficients can be determined recursively

$$\eta^\alpha_{\mu+1_k} = D_k\eta^\alpha_\mu - \sum_{i=1}^n D_k\zeta^i p^\alpha_{\mu+1_i}\,. \tag{5.4}$$

If now a local representation of $\mathcal{R}_q$ is given by $\Phi^\tau = 0$ and

$$\mathrm{pr}^{(q)}\mathbf{v}(\Phi^\tau)\Big|_{\mathcal{R}_q} = 0 \tag{5.5}$$

holds for all functions $\Phi^\tau$, then the flow generated by $\mathbf{v}$ is a symmetry in the sense of Definition 5.2.

One usually calls (5.5) the *determining system* of $\mathcal{R}_q$. It is an, in general over-determined, system of linear partial differential equations. This shows the decisive advantage of symmetry analysis: even if $\mathcal{R}_q$ is a non-linear equation, its determining system is always linear and thus usually much easier to tackle. This linearization corresponds to the fact that we use instead of the Lie group of finite transformations its Lie algebra.

From a computational point of view the biggest problem in evaluating (5.5) lies in the restriction to $\mathcal{R}_q$. Usually this is solved by assuming that one is given a local representation of $\mathcal{R}_q$ where each equation is solved for some derivative. Then one must simply perform the corresponding substitution. For more general equation one could use Gröbner basis techniques (cf. Section 8.6).

As a trivial application of the results of Chapter 3 one can calculate the size of the symmetry algebra, if it is not possible to solve explicitly the determining system (5.5). Reid [125, 126] has shown that it is even possible to obtain the structure constants of the algebra without solving (5.5), although there are some problems in the case of an infinite-dimensional algebra.

Symmetries have many applications. They are e.g. connected with the so-called complete integrability of an equation or they help to decide whether or not a system can be transformed into a linear one. They also give a group theoretic fundament to separation ansätze. We will not discuss any of these topics but refer again to the textbooks mentioned at the beginning of this section. But we will consider in some detail in the next section the perhaps most important application of symmetries, namely the construction of special solutions using reduction methods.

## 5.2 Reduction of Differential Equations

Reduction with respect to a symmetry group provides a standard technique for the construction of solutions. It is well-known that only for ordinary differential equations one can reconstruct the general solution of the original equation from the general solution of the reduced equation by quadratures. In the case of partial differential equations one obtains only special solutions.

The idea behind symmetry reduction is to look for solutions which are invariant under a symmetry group, i.e. the symmetry transformations map such a solution into itself. If the symmetry is generated by a vector field $\mathbf{v}$ of the form (5.2), then a group invariant solution $u^\alpha(x^i)$ satisfies not only the considered differential equation $\mathcal{R}_q$ but in addition the *invariant surface condition*

$$\sum_{i=1}^n \zeta^i(x,u)p^\alpha_i = \eta^\alpha(x,u)\,, \qquad \alpha = 1,\dots,m\,. \tag{5.6}$$

These are quasi-linear, first-order equations. In the case of a higher-dimensional symmetry algebra we must add one such set of conditions for each generator. If one can compute the general solution of (5.6), one substitutes it into the equations of $\mathcal{R}_q$ to obtain a reduction. Or, if we assume without loss of generality that $\zeta^n \neq 0$, we can solve (5.6) for the derivatives $p_n^\alpha$ and thus eliminate in $\mathcal{R}_q$ all derivatives with respect to $x^n$. This yields a differential equation in $n-1$ independent variables.

The latter approach is a special case of a more general symmetry reduction method using *invariants*. These are functions $\delta(x,u)$ satisfying $\mathbf{v}\delta = 0$ for every symmetry generator $\mathbf{v}$. Since the generators form a Lie algebra, they represent the solutions of a complete system (cf. Section 7.1).

Let us assume that the symmetry group with respect to which the reduction is to be performed acts regularly on $\mathcal{E}$ with $s$-dimensional orbits. It is well-known that there exist locally $m+n-s$ functionally independent invariants. We divide them into two classes

$$\begin{aligned} y^1 &= \xi^1(x,u)\,, \ \ldots,\ y^{n-s} = \xi^{n-s}(x,u)\,,\\ v^1 &= \vartheta^1(x,u)\,, \ldots,\ v^m = \vartheta^m(x,u)\,. \end{aligned} \tag{5.7}$$

The $y^i$ will be the new independent variables, the $v^\alpha$ the new dependent variables. The result of the reduction does not really depend on this division, as the different reduced systems are related by some kind of "hodograph" transformations.

Now we must assume that we can solve (5.7) for $n-s$ of the $x^i$ and for all of the $u^\alpha$, i.e. that the symmetry group acts transversally. The remaining $x^i$ play the role of parameters. Using the chain rule these solved expressions can be substituted in the differential equation. The resulting equation will be equivalent to one without the parametric $x^i$. Thus we have obtained an equation containing only $n-s$ independent variables.

Bluman and Cole [14] were the first to observe that for this construction it is not necessary that $\mathbf{v}$ generates a symmetry of the differential equation. It suffices, if $\mathbf{v}$ yields a symmetry of $\mathcal{R}_q$ augmented by the invariant surface condition (5.6)[1]. Since the coefficients $\zeta^i, \eta^\alpha$ of $\mathbf{v}$ are now also part of the differential equation, the resulting determining system is non-linear opposed to the situation in the classical Lie theory. This is a serious draw-back in the concrete application of this so-called *non-classical method*.

The transformations generated by such vector fields are sometimes called *conditional symmetries* [95], as in general they do not map all solutions of $\mathcal{R}_q$ into other solutions. They also do not form a Lie algebra, because every conditional symmetry depends on its own invariant surface condition (5.6).

The non-classical method often yields additional reductions: addition of further equations leads to a smaller submanifold in the jet bundle, thus to less restrictions for symmetry generators. Olver and Rosenau [113, 114] generalized this idea to so-called *weak symmetries* or the method of *differential constraints*. There is no reason why one should add only invariant surface conditions. Every additional equation or differential constraint will lead to a lower-dimensional submanifold and thus potentially to further conditional symmetries. Behind this approach lies the simple observation that *any* solution of a differential equation remains invariant under some group.

The choice of suitable constraints represents, however, a big unsolved problem. Most will simply lead to an inconsistent equation, i.e. to an equation without solution. Others will not yield new symmetries, yet others will lead to intractable equations etc. For this reason the method of differential constraints has not yet found much application. It is

[1] This is especially easy to see using exterior systems (cf. Proposition A.5 in Section A.1).

simply too general. In the context of formal theory it is interesting, because it shifts the emphasis from group theory to the theory of over-determined equations.

The *direct reduction method* of Clarkson and Kruskal [30] could be viewed as an answer to the last problem. It was originally developed for an equation with $m = 1$ dependent variable $u$ and $n = 2$ independent variables $x, t$. It consists of the solution ansatz

$$u(x,t) = U(x,t,w(z)), \qquad z = \xi(x,t) \tag{5.8}$$

where one requires that there exist functions $U(x,t,w), \xi(x,t)$ such that $\mathcal{R}_q$ reduces to an ordinary differential equation for $w$. The extension to more variables is straight-forward.

The direct reduction method was introduced as an ad hoc ansatz without a rigorous theoretical foundation. Levi and Winternitz [95] were the first to point out that it was closely related to the non-classical method; later Arrigo *et al.* [4] gave exact conditions, when the two are equivalent. Olver [112] showed recently that there is a one-to-one correspondence between direct reduction ansätze (5.8) and invariant surface conditions of the restricted form

$$\zeta(x,t)u_x + \tau(x,t)u_t = \eta(x,t,u), \tag{5.9}$$

i.e. the corresponding vector field $\mathbf{v} = \zeta\partial_x + \tau\partial_t + \eta\partial_u$ generates a group of fiber preserving transformations. $\xi$ is the (more or less) unique invariant of this group: $\mathbf{v}(\xi) = 0$.

Galaktionov [55] proposed a generalization of the direct reduction method using the ansatz

$$u(x,t) = U(x,t,w_1(z),w_2(z)) \tag{5.10}$$

and the requirement that the reduction yields a system of ordinary differential equations for $w_1$ and $w_2$. This can obviously be extended to an ansatz of *n-th order* using $n$ functions $w_i$ of the same similarity variable $z$. Olver [112] showed that this ansatz is equivalent to an $n$-th order differential constraint of the form

$$\mathbf{v}^n(u) = \Phi\Big(x,t,u,\mathbf{v}(u),\dots,\mathbf{v}^{n-1}(u)\Big) \tag{5.11}$$

with a vector field $\mathbf{v} = \zeta\partial_x + \tau\partial_t$ satisfying $\mathbf{v}(\xi) = 0$. Note that, (5.9) and (5.11) become parameterized ordinary differential equations after transforming to rectifying coordinates for $\mathbf{v}$.

Estévez [49, 50] has combined the direct method with the singular manifold method from Painlevé analysis [162, 163]. This leads basically to the possibility of obtaining reductions to ordinary differential equations of lower order than the original partial differential equation. The method seems, however, to be applicable only to equations which have at least the conditional Painlevé property.

## 5.3 Loss of Generality

As already mentioned in the last section reduction methods yield only special solutions in the case of partial differential equations. Our goal in this section will be to show how the loss of generality can be quantified using the results of Chapter 3. One must, however, be aware that this loss represents only one side of the problem. The less we reduce, i.e. the smaller the dimension of the used symmetry group, the smaller the loss of generality. But usually the more we reduce the easier the resulting equation. Indeed, as we have seen in the case of the direct reduction method, in applications one tries very often to obtain an ordinary differential equation, although this represents of course the maximal loss of generality.

We will start with the special case of a classical symmetry reduction of a normal equation, as here it is possible to exhibit general results. We assume that we are given a local representation in solved form

$$p^\beta_{[0,\dots,q]} = \Phi^\beta(x^i, u^\alpha, p^\alpha_\mu)\,, \quad \beta = 1,\dots,m \tag{5.12}$$

where the right hand sides do not contain any derivative of order $q$ which is purely with respect to $x^n$ (cf. Section 4.3). We computed already in Example 3.11 the Cartan characters for such an equation:

$$\alpha_N^{(n)} = 0\,,\ \alpha_N^{(n-k)} = m\binom{q+k-1}{q-1}\,. \tag{5.13}$$

Let us further assume that the symmetry group acts regularly with $s$-dimensional orbits and that after the reduction the equation is still normal. Then we obtain a system with $\bar{n} = n - s$ independent and $\bar{m} = m$ dependent variables of order $\bar{q} = q - \Delta q$. The Cartan characters of the reduced system are $\bar{\alpha}_{\bar{q}}^{(\bar{n})} = 0, \bar{\alpha}_{\bar{q}}^{(\bar{n}-k)} = m\binom{\bar{q}+k-1}{\bar{q}-1}$. Thus if $\Delta q = 0$, then $\bar{\alpha}_q^{(k)} = \alpha_q^{(k+s)}$. If $\Delta q > 0$, we get an additional loss of generality as to be expected.

Since the Cartan characters form a descending sequence (Proposition 3.5), this result means that we do not only loose all arbitrary functions with more than $n - s$ arguments, but that even the number of arbitrary functions with less arguments decreases.

Even in more general situations the loss of arbitrariness can be computed without explicit construction of the reduced equation. One must simply compare the Cartan characters of the original equation alone with the characters of the equation augmented by the invariant surface conditions (5.6) — after completion to an involutive equation of course. It is, however, difficult to obtain this way general results like the above ones for normal equations, as it is unclear, how one can predict the characters of the augmented equation.

An extension of this method to reductions with respect to generalized or to weak symmetries is straight-forward. For instance, the analysis of Lie-Bäcklund symmetries of normal first-order equations leads to the addition of differential constraints of order $r$ [108]

$$\Psi^\tau(x^i, u^\alpha, p^\alpha_\mu) = 0\,, \qquad \begin{cases} \tau &= 1,\dots,p\,, \\ |\mu| &\leq r\,, \\ \mu_n &= 0\,. \end{cases} \tag{5.14}$$

Note that $x^n$ occurs here only as parameter, because all derivatives with respect to $x^n$ can be eliminated using (5.12).

We assume that system (5.14) is in involution and comprises $\tilde{\beta}_r^{(k)}$ equations of class $k$ for $k = 1,\dots,n-1$. The combined system (5.12,5.14) has an involutive symbol, if and only if there exist matrices $(M^i)^\tau_\rho$ such that

$$(M^i)^\tau_\rho \frac{\partial \Psi^\rho}{\partial p^\alpha_\mu} = -\frac{\partial \Psi^\tau}{\partial p^\beta_\mu}\frac{\partial \Phi^\beta}{\partial p^\alpha_i}\,, \qquad \begin{cases} i=1,\dots,n-1\,, \\ |\mu|=r\,. \end{cases} \tag{5.15}$$

No integrability conditions arise, i.e. the system is involutive, if and only if additionally

$$D_n\Psi^\tau - \sum_{\alpha,|\mu|=r} \frac{\partial \Psi^\tau}{\partial p^\alpha_\mu}(p^\alpha_{\mu+1_n} - D_\mu\Phi^\alpha) - \sum_{i=1}^{n-1}(M^i)^\tau_\rho D_i\Psi^\rho = 0\,. \tag{5.16}$$

Since (5.14) contains no derivatives with respect to $x^n$, we get for the combined system $\beta_r^{(n)} = m$, $\beta_r^{(n-k)} = m\binom{r+k-2}{r-2} + \tilde{\beta}_r^{(n-k)}$. This yields the Cartan characters

$$\alpha_r^{(n)} = 0\,, \ \alpha_r^{(n-k)} = m\binom{r+k-2}{r-1} - \tilde{\beta}_r^{(n-k)}\,. \tag{5.17}$$

This generalizes the results of [108] to non-linear equations and to constraints which depend on $x^n$ and which do not form a normal system.

## 5.4 Finding Differential Constraints

We have already said that the big problem in the application of the method of differential constraints is the choice of the constraints. Formal theory can help here, if we follow the basic idea behind the direct reduction method. One should, however, stress already now the word "help": we will not obtain an algorithm like in the direct reduction method.

The goal in reduction is usually to obtain ordinary differential equations. We will prove later in Section 7.1 that the general solution of every equation of finite type can be constructed by solving only ordinary differential equations. Lie [97] showed furthermore that conversely every family of functions parameterized by a finite number of constants can be written as the solution of a finite type equation. Thus our goal should be to obtain such an equation after the reduction. This is considerably more general as the direct reduction method.

Unfortunately this approach has two severe draw-backs. We will see later on that for single second-order equations in two independent variables it yields almost no restriction on the form of the constraints. Furthermore the conditions take usually the form of inequalities. Thus there exists no direct method to find constraints satisfying them.

For most of this section we will concentrate on the case of a single second-order equation

$$\Delta(x, t, u, u_x, u_t, u_{xx}, u_{xt}, u_{tt}) = 0 \tag{5.18}$$

and the addition of one differential constraint. We will assume throughout that the considered constraints are compatible. Conditions for this will become evident during the analysis.

**Proposition 5.5** *Adding a compatible first-order differential constraint to (5.18) leads always to an equation of finite type.*

Proof. We can assume without loss of generality that the constraint is solved for $u_x$

$$u_x - \phi(x, t, u, u_t) = 0\,. \tag{5.19}$$

Hence all $x$-derivatives in Eq. (5.18) can be eliminated and we obtain

$$\tilde{\Delta}(x, t, u, u_t, u_{tt}) = 0\,. \tag{5.20}$$

We must now distinguish between a degenerate case when $\tilde{\Delta}_{u_{tt}} = 0$ and the generic case where (5.20) is still a second order equation. This distinction can be characterized intrinsically by the rank of a symbol.

*Generic case:* We have to investigate, when the equation

$$\mathcal{R}_2 : \begin{cases} u_x - \phi(x, t, u, u_t) = 0\,, \\ u_{xx} - \phi_{u_t} u_{xt} - \phi_u u_x - \phi_x = 0\,, \\ u_{xt} - \phi_{u_t} u_{tt} - \phi_u u_t - \phi_t = 0\,, \\ \tilde{\Delta}(x, t, u, u_t, u_{tt}) = 0 \end{cases} \tag{5.21}$$

is involutive. Its symbol $\mathcal{M}_2$ has not full rank, if and only if

$$\tilde{\Delta}_{u_{tt}} = \Delta_{u_{xx}}\phi_{u_t}^2 + \Delta_{u_{xt}}\phi_{u_t} + \Delta_{u_{tt}} = 0\,. \tag{5.22}$$

Then rank $\mathcal{M}_2 = 2$ and we are in the degenerate case.

Thus we have $\dim \mathcal{M}_2 = 0$ in the generic case and the symbol is trivially involutive. Prolongation to third order shows that there may be an integrability condition of the following form:

$$\tilde{\Delta}_x - \tilde{\Delta}_t\phi_{u_t} + \tilde{\Delta}_u(\phi - \phi_{u_t}u_t) + \tilde{\Delta}_{u_t}(D_t\phi - u_{tt}) + \tilde{\Delta}_{u_{tt}}D_{tt}\phi = 0\,. \tag{5.23}$$

If this equation vanishes identically due to (5.19) and (5.20), then $\mathcal{R}_2$ is an involutive finite type equation with a two-dimensional solution space. A prototypical example for this situation is the direct reduction with the Clarkson-Kruskal ansatz (5.8) yielding a second order ordinary differential equation.

If, however, (5.23) does not vanish identically, then an integrability condition arises. In general it will be second order. Because of the vanishing symbol $\mathcal{M}_2$, one can always rewrite it as a first order (or perhaps even algebraic) equation. If it is a first order equation, then we can solve it for $u_t$ ($u_x$ can again be eliminated using (5.19)). Depending on whether this new constraint reduces (5.20) to zero or to an algebraic equation, we get a zero or a one-dimensional solution (an algebraic equation is of course an implicit solution, as we assumed that (5.19) is a compatible constraint). The case of a one-dimensional solution should correspond to the reductions obtained by Estévez.

*Degenerate case:* In this case we are left with a first order equation

$$\mathcal{R}_1 : \begin{cases} u_x - \phi(x,t,u,u_t) = 0\,, \\ \tilde{\Delta}(x,t,u,u_t) = 0\,. \end{cases} \tag{5.24}$$

If $\tilde{\Delta}_{u_t} = 0$, we have found a reduction to an algebraic equation, i.e. an implicit solution. Otherwise the symbol vanishes and is thus involutive. If there occurs an integrability condition, it must be an algebraic equation, i.e. it yields again an implicit solution. If $\mathcal{R}_1$ is involutive, then we get a one-dimensional solution. •

It should be noted that it may not always be possible to find constraints such that the degeneracy condition (5.22) is satisfied. Especially for equations which are non-linear in the highest-order derivatives this will usually not be the case. For quasi-linear systems we rediscover the usual distinction in hyperbolic, parabolic and elliptic systems. One must, however, be careful with the coordinate system, as we have solved the constraint for a distinguished derivative.

**Proposition 5.6** *Adding a compatible second-order differential constraint to (5.18) leads almost always to an equation of finite type. A positive Cartan genus occurs, if and only if the combined system is involutive and the symbol of its prolongation has rank 3.*

Proof. Again we can assume without loss of generality that the constraint is in solved form

$$u_{xx} - \psi(x,t,u,u_x,u_t,u_{xt},u_{tt}) = 0\,. \tag{5.25}$$

Eliminating $u_{xx}$ from (5.18) leads to three different cases depending on the form of the reduced equation (and in Case B on an additional condition for the constraint):

$$\begin{array}{ll} \text{A)} & u_{tt} - \tilde{\Delta}(x,t,u,u_x,u_t,u_{xt}) = 0\,, \\ \text{B)} & u_{xt} - \tilde{\Delta}(x,t,u,u_x,u_t) = 0\,, \\ \text{C)} & \tilde{\Delta}(x,t,u,u_x,u_t) = 0\,. \end{array} \tag{5.26}$$

Case C is trivial, one just considers the reduced equation as a first order constraint to the second order equation (5.25) and recovers the situation of Proposition 5.5.

The distinction between the three cases is again induced by the rank of symbols. We are interested in the equation

$$\mathcal{R}_2 : \begin{cases} u_{xx} - \psi(x,t,u,u_x,u_t,u_{xt},u_{tt}) = 0\,, \\ \Delta(x,t,u,u_x,u_t,u_{xx},u_{xt},u_{tt}) = 0\,. \end{cases} \tag{5.27}$$

In Case C the symbol $\mathcal{M}_2$ has not maximal rank. Cases A and B differ in the rank of the prolonged symbol $\mathcal{M}_3$.

*Generic case:* This is Case A. Independent of the way we order the variables, we find that $\mathcal{M}_2$ has three multiplicative variables (one equation of class 2 and one of class 1). To check whether or not $\mathcal{M}_2$ is involutive, we must compute the rank of its prolongation

$$\mathcal{M}_3 : \begin{cases} v_{xxx} - \psi_{u_{xt}} v_{xxt} - \psi_{u_{tt}} v_{xtt} = 0\,, \\ v_{xxt} - \psi_{u_{xt}} v_{xtt} - \psi_{u_{tt}} v_{ttt} = 0\,, \\ v_{xtt} - \tilde{\Delta}_{u_{xt}} v_{xxt} = 0\,, \\ v_{ttt} - \tilde{\Delta}_{u_{xt}} v_{xtt} = 0\,. \end{cases} \tag{5.28}$$

Since it is obviously four, $\mathcal{M}_2$ is not involutive according to (2.19). But $\mathcal{M}_3$ is involutive, as it vanishes. If there arises no integrability condition in the prolongation from $\mathcal{R}_3$ to $\mathcal{R}_4$, then $\mathcal{R}_3$ is an involutive finite type equation with a four-dimensional solution space. The prototypical example for this case is given by direct reduction using Galaktionov's ansatz (5.10) leading to two second order ordinary differential equations.

One can compute the general form of the possible integrability condition, but we omit it here, as it is a fairly lengthy expression. Since $\mathcal{M}_3$ vanishes, we can conclude that the integrability condition is at most of second order. There arise now many case distinctions depending on the order of the condition and on whether or not the projected system is involutive. We omit these, too, and conclude by noting that we have in any case a finite type equation with a maximally three-dimensional solution space.

*Degenerate case:* In Case B we can write the symbol $\mathcal{M}_3$ in the following form

$$\mathcal{M}_3 : \begin{cases} v_{xxx} - \psi_{u_{xt}} v_{xxt} - \psi_{u_{tt}} v_{xtt} = 0\,, \\ v_{xxt} - \psi_{u_{xt}} v_{xtt} - \psi_{u_{tt}} v_{ttt} = 0\,, \\ v_{xxt} = 0\,, \\ v_{xtt} = 0\,. \end{cases} \tag{5.29}$$

Its rank depends now on $\psi_{u_{tt}}$. If it is not zero, the rank of the symbol is still four and we are in Case A. If it vanishes, however, then $\operatorname{rank} \mathcal{M}_3 = 3$ and according to (2.19) the symbol $\mathcal{M}_2$ is now involutive.

During the prolongation from $\mathcal{R}_2$ to $\mathcal{R}_3$ the following integrability condition arises

$$D_x\tilde{\Delta} - \psi_{u_{xt}} D_t\tilde{\Delta} - D_t^{(1)}\psi = 0\,, \tag{5.30}$$

where the superscript (1) indicates that only the jet variables up to order 1 are taken into account. If it vanishes identically, then $\mathcal{R}_2$ is involutive. However, it is not a finite type equation, as $\dim \mathcal{M}_2 = 1$. Since the Cartan characters are $\alpha_2^{(2)} = 0$ and $\alpha_2^{(1)} = 1$, we find a Cartan genus of 1.

If (5.30) truly provides an integrability condition, then the projected equation $\mathcal{R}_2^{(1)}$ is of finite type. Again we get a flood of case distinctions depending on the order of (5.30) and on whether or not $\mathcal{R}_2^{(1)}$ is involutive. In any case the solution space is at most three-dimensional. •

The two preceding propositions can be extended to equations of higher than second order. We will restrict ourselves mainly to the respective generic cases. There arise now much more case distinctions, as there are more possibilities for the order of the integrability conditions. But we have already seen, that this affects basically only the dimension of the solution space and not the principal results. Furthermore, the generic case is the most interesting one, as it yields the maximal dimension.

The case of first order constraints runs completely analogously to Proposition 5.5. Again all derivatives with respect to $x$ can be replaced by $t$-derivatives using the constraint; the reduced equation is a parameterized ordinary differential equation. The degeneracy condition (5.22) is replaced by

$$\Delta_{u_{[q,0]}}\phi_{u_t}^q + \Delta_{u_{[q-1,1]}}\phi_{u_t}^{q-1} + \cdots + \Delta_{u_{[0,q]}} = 0 \tag{5.31}$$

where $q$ denotes the order of the differential equation and where we have used the notation

$$u_{[r,s]} = \frac{\partial^{r+s}u}{\partial x^r\,\partial t^s}\,. \tag{5.32}$$

Prolongation of the constraint (5.19) to order $q$ yields $q$ independent equations. One can easily see, that the symbol $\mathcal{M}_q$ of the equation $\mathcal{R}_q$ which arises by adjoining these equations to the original equation always vanishes, if condition (5.31) does not hold. In the case that no integrability conditions occur, $\mathcal{R}_q$ is thus an involutive finite type equation with a $q$-dimensional solution space. Again the direct reduction with the Clarkson-Kruskal ansatz (5.8) provides the prototypical example for this situation.

Proposition 5.6 is generalized by considering a constraint of order $q$. For simplicity we assume again that the constraint is in solved form

$$u_{[q,0]} = \psi(x,t,u,u_x,u_t,\ldots,u_{[0,q-1]},u_{[q-1,1]},\ldots,u_{[0,q]})\,. \tag{5.33}$$

Eliminating $u_{[q,0]}$ yields the equation

$$\tilde{\Delta}(x,t,u,u_x,u_t,\ldots,u_{[0,q-1]},u_{[q-1,1]},\ldots,u_{[0,q]}) = 0\,. \tag{5.34}$$

We must now analyze the equation $\mathcal{R}_q$ comprising (5.33) and (5.34).

Omitting the analog of Case C in the proof of Proposition 5.6, rank $\mathcal{M}_q = 2$ and there are three multiplicative variables. The prolonged symbols $\mathcal{M}_{q+r}$ are given by

$$\mathcal{M}_{q+r}\ :\ \begin{cases} v_{[q+i,r-i]} - \displaystyle\sum_{j=1}^{q} \psi_{u_{[q-j,j]}} v_{[q-j+i,r+j-i]} = 0 \\ \displaystyle\sum_{j=1}^{q} \tilde{\Delta}_{u_{[q-j,j]}} v_{[q-j+i,r+j-i]} = 0 \end{cases} \qquad i = 0,\ldots,r\,. \tag{5.35}$$

As long as these symbols have maximal rank, we get rank $\mathcal{M}_{q+r} = 2r+2$ and $\mathcal{M}_{q+r}$ has $2r+3$ multiplicative variables. Since this symbol is defined in a vector space of the dimension $q+r+1$ (the number of derivatives of order $q+r$), we will reach a vanishing and thus involutive symbol for $r = q-1$. The corresponding system $\mathcal{R}_{2q-1}$ will comprise $q(q+1)$ equations and thus its solution space has dimension $q^2$. This case will arise with a direct reduction ansatz of order $q$ leading to $q$ ordinary differential equations of order $q$.

We see that as long as every symbol has maximal rank

$$\operatorname{rank}\mathcal{M}_{q+r+1} = \#\text{multiplicative variables}(\mathcal{M}_{q+r}) + 1 \tag{5.36}$$

holds. The number of multiplicative variables of a symbol provides a lower bound for the rank of its prolongation (Proposition 2.9). Thus if one of the symbols $\mathcal{M}_{q+r}$ has not maximal rank, then its rank is $2r+1$ and the preceding symbol is involutive! If now no integrability conditions arise, i.e. the corresponding system is involutive, too, then we have not an equation of finite type. This situation will, however, never occur, if $\psi_{[0,q]} \neq 0$, as one can easily see from (5.35). Note that this is only a sufficient and not a necessary criterion.

The analysis becomes much more involved, if we consider differential equations in more than two independent variables. In general, one constraint is now of course not sufficient to obtain an equation of finite type. For first order constraints we can extend Proposition 5.5 by considering $n-1$ constraints of the form

$$u_{x^k} = \phi(x^i, u, u_{x^j})\,, \qquad k = 1, \dots, n-1, \quad j \neq k\,, \tag{5.37}$$

where $n$ denotes the number of independent variables.

The only way to be sure that $q$-th order constraints lead to an equation of finite type is to add so many constraints that the symbol $\mathcal{M}_q$ vanishes. But sometimes just one constraint may suffice, if it generates enough integrability conditions. If we take the equation

$$u_{zz} + y u_{xx} = 0\,, \tag{5.38}$$

then we know already from Example 2.19 of Janet that addition of the constraint

$$u_{yy} = 0 \tag{5.39}$$

suffices to obtain an equation of finite type.

The principal way to use the formal theory for finding differential constraints would be thus to *choose* a scenario for the completion algorithm, i.e. at which step where an integrability condition shall appear, and an ansatz for the constraint. This yields on one side further hints for the form of the constraints from the integrability conditions one wants to have and on the other side differential equations for the constraint from the ones which should vanish. Unfortunately this procedure contains a lot of guessing and we have so far not been able to produce this way a new non-trivial reduction.

# Chapter 6

# Field Theories

*We will apply the results of the previous chapters to some systems of physical interest. We will start by considering the effect of gauge symmetries; then we will show, how the notion of strength is related to the Cartan characters. Systems with constraints will be analyzed from the point of view of formal theory. As concrete examples the Yang-Mills Equations and the Einstein Equations are treated.*

## 6.1 Gauge Systems

Gauge transformations are a special kind of symmetries; they preserve the fibers of the bundle $\mathcal{E}$. We call a differential equation a *gauge system*, if it is invariant under transformation of the form

$$\begin{aligned} \bar{x}^i &= \Omega^i(x^j)\,, \\ \bar{u}^\alpha &= \Lambda^\alpha(x^i, u^\beta, \lambda_a^{(0)}(x), \partial\lambda_a^{(1)}(x), \dots, \partial^p\lambda_a^{(p)}(x)) \end{aligned} \tag{6.1}$$

where $\gamma_0$ gauge functions $\lambda_a^{(0)}$ are entering algebraically, $\gamma_1$ gauge functions $\lambda_a^{(1)}$ are entering through their first derivatives etc. We assume that all derivatives of order $l$ of the functions $\lambda^{(l)}$ are explicitly occurring in (6.1) and that the gauge functions depend on all independent variables.

The only reason for these assumptions is simplicity. The formalism can handle more complicated situations, but general results would look rather awkward. In such cases it is better to use the pseudogroup approach presented later in this section. In order to be a gauge system a differential equation must obviously be under-determined, as (6.1) implies that its general solution contains arbitrary functions of all independent variables.

Such systems occur frequently in physics, a typical example being the Yang-Mills Equations which will be studied in Section 6.6. Here the arbitrariness as considered in Chapter 3 is not really interesting, because physically solutions which are connected by a gauge transformation can be identified. They lead to the same values for all observables. Thus we must eliminate the effect of the symmetry (6.1) to obtain the number of physical degrees of freedom. Furthermore we will use our results for the analysis of gauge fixing conditions.

The symmetry (6.1) implies that

$$G_q(r) = \sum_{l=0}^{p} \gamma_l \begin{bmatrix} n \\ q+r+l \end{bmatrix} \tag{6.2}$$

Taylor coefficients of order $q+r$ can be arbitrarily set by a gauge fixing. Thus they must be subtracted from the Hilbert polynomial to obtain the physically relevant number of free coefficients. We introduce the *gauge corrected Hilbert polynomial*

$$\bar{H}_q(r) = H_q(r) - G_q(r)\,. \tag{6.3}$$

Its coefficient of $r^k$ is given by

$$\bar{h}_k = h_k - \frac{1}{(n-1)!}\sum_{l=0}^{p}\gamma_l s^{(n-1)}_{n-k-1}(q+l)\,, \quad k=0,\dots,n-1\,. \tag{6.4}$$

Using (3.12) we can associate with the corrected Hilbert polynomial *gauge corrected Cartan characters*

$$\bar{\alpha}^{(k)}_q = (k-1)!\,\bar{h}_{k-1} - \sum_{i=k+1}^{n}\frac{(k-1)!}{(i-1)!}\bar{\alpha}^{(k)}_q s^{(i-1)}_{i-k}(0)\,, \quad k=1,\dots,n\,. \tag{6.5}$$

To obtain now the number of physically relevant free functions in the general solution of our equation, we simply apply the results of Chapter 3 using the gauge corrected Cartan characters and Hilbert polynomial, respectively. This yields for instance the following modification to Theorem 3.9.

**Theorem 6.1** *If the physical solution space, i.e. the solution space modulo the gauge symmetry (6.1), has an algebraic representation, then it is spanned by $p_k$ arbitrary functions with $k$ arguments, where $p_k = f_k - g_k$ with*

$$\begin{aligned} g_n &= \sum_{l=0}^{p}\gamma_l\,,\\ g_k &= \frac{(k-1)!}{(n-1)!}\sum_{l=0}^{p}\gamma_l\, s^{(n-1)}_{n-k}(q+l) - \sum_{i=k+1}^{n}\frac{(k-1)!}{(i-1)!}g_i\, s^{(i-1)}_{i-k}(q)\,. \end{aligned} \tag{6.6}$$

Our formulation of a gauge symmetry has two disadvantages: It is not intrinsic, as there might be different formulations yielding different values for the $\gamma_l$ and it does not cover the general case, as we had to make the assumption that all derivatives of the specified order of a gauge function occur and that all gauge functions depend on all independent variables. All these problems can be circumvented by the use of a Lie pseudogroup [117] to describe the gauge transformation.

We will not give here a complete introduction into the theory of Lie pseudogroups. We will take the simple point of view that instead of an explicit expression like (6.1) for the gauge transformation we have a differential equation whose general solution is (6.1). In other words we eliminate all the parameters and gauge functions through differentiation and obtain a differential equation with the old coordinates $(x^i, u^\alpha)$ as independent variables and the new coordinates $(\bar{x}^i, \bar{u}^\alpha)$ as dependent variables. The special form of (6.1) implies that $\partial\bar{x}^i/\partial u^\alpha = 0$.

**Example 6.2** As a trivial example of a (finite-dimensional) Lie pseudogroup we consider translations

$$\bar{x}^i = x^i + a^i\,, \quad \bar{u}^\alpha = u^\alpha + b^\alpha\,. \tag{6.7}$$

They can be described by the following differential equation (of finite type)

$$\begin{aligned} \frac{\partial\bar{x}^i}{\partial x^j} &= \delta^i_j\,, & \frac{\partial\bar{x}^i}{\partial u^\beta} &= 0\,,\\ \frac{\partial\bar{u}^\alpha}{\partial x^j} &= 0\,, & \frac{\partial\bar{u}^\alpha}{\partial u^\beta} &= \delta^\alpha_\beta\,. \end{aligned} \tag{6.8}$$

An example of a Lie pseudogroup where it is not possible to write down an explicit expression of the form (6.1) for the transformations is given by the group of volume preserving mappings. They satisfy

$$\begin{aligned} \frac{\partial \bar{x}^i}{\partial u^\alpha} &= 0\,, & \frac{\partial \bar{u}^\alpha}{\partial x^j} &= 0\,, \\ \det\left(\frac{\partial \bar{x}^i}{\partial x^j}\right) &= 1\,, & \det\left(\frac{\partial \bar{u}^\alpha}{\partial u^\beta}\right) &= 1\,. \end{aligned} \tag{6.9}$$

Let the Lie pseudogroup be realized as a differential equation $\mathcal{G}_g$ of order $g$. Its Hilbert polynomial $H_g^{(G)}(r)$ yields the number of arbitrary coefficients at order $g+r$ in the explicit transformation law (6.1). An intrinsic way to compute $G_q(r)$ is given by

$$G_q(r) = H_g^{(G)}(r + q - g)\,. \tag{6.10}$$

Obviously this expression is only valid for $r > \max(g - q, 0)$. Possible values for the numbers $\gamma_l$ can now be computed using the methods of Chapter 3. The form (6.1) arises, if there exists a solution of the diophantine system (3.23) with $f_{n,l} = \gamma_l$ and all other $f_{k,j}$ vanish.

We can easily compute the remaining gauge freedom after some gauge conditions have been imposed, if they take the form of differential equations. These equations are added to the original system. This yields in general an over-determined system. It should be invariant only under a proper subgroup of the gauge group (It is, however, possible that new symmetries arise!).

A necessary condition for a complete gauge fixing is that the Cartan characters of this augmented equation are the same as the gauge corrected Cartan characters of the original equation (one can of course also use the Hilbert polynomial). Note, however, that this is only a necessary criterion, as there may still remain a finite-dimensional symmetry depending on some arbitrary *constants*. This cannot be checked with the Cartan characters, for they are only related with the number of arbitrary functions.

One should note that the gauge correct Cartan characters are computed in a formal, algebraic manner. Hence one cannot conclude that there exists necessarily a differential equation leading to exactly these characters. Especially, they do not necessarily satisfy Proposition 3.5. If this happens, it is not possible to specify a complete gauge fixing using differential equations (this comprises of course purely algebraic conditions like the Coulomb gauge). One must use other conditions like initial or boundary conditions.

## 6.2 Strength

Einstein [47] introduced another measure for the arbitrariness, the *strength*. He was lead to such formal considerations during his search for a unified field theory. His philosophy was: if we have two systems of field equations and we cannot decide which is the better one for physical reasons, then we should take the one which restricts the solution stronger. He discussed this problem already earlier with Cartan [28], who tried to explain him his theory of involutive systems. The strength has not found much attention in the literature, only a few papers [20, 74, 103, 105, 106, 110, 115, 133, 147] have been devoted to it.

We introduce

$$Z_q(r) = H_q(r)/\left[{n \atop q+r}\right]. \tag{6.11}$$

$Z_q$ is a rational function of $r$. The denominator is of degree $n-1$; the numerator has a degree less than or equal to $n-1$. $Z_q$ describes the growth of the number of free Taylor coefficients in the general solution relative to a function of $n$ variables. We expand $Z_q(r)$ in powers of $1/r$

$$Z_q(r) = Z_q^{(0)} + Z_q^{(1)}/r + O(1/r^2)\,. \tag{6.12}$$

**Definition 6.3** *We call* $Z_q^{(0)}$ *the* compatibility coefficient *and* $Z_q^{(1)}$ *the* strength *of* $\mathcal{R}_q$. *An equation with* $Z_q^{(0)} = 0$ *is said to be* absolutely compatible.

Einstein introduced the strength only for absolutely compatible equations. It is, however, well-defined for any differential equation. The next theorem relates it to the Cartan characters and thus to the results of Chapter 3.

**Theorem 6.4** *The compatibility coefficient and the strength of an involutive equation* $\mathcal{R}_q$ *with Cartan characters* $\alpha_q^{(k)}$ *are given by*

$$Z_q^{(0)} = \alpha_q^{(n)}\,, \tag{6.13}$$

$$Z_q^{(1)} = (n-1)\Big(\tfrac{1}{2}n\,\alpha_q^{(n)} + \alpha_q^{(n-1)}\Big)\,. \tag{6.14}$$

Proof. Using (3.11) and (C.2), $Z_q^{(0)}$ is readily obtained by taking the limit $r \to \infty$. Subtracting $Z_q^{(0)}$, multiplying (6.12) by $r$ and taking again the limit yields $Z_q^{(1)}$. •

If there exists an algebraic representation of the general solution, then we can express the compatibility coefficient and the strength through the $f_k$ instead of the $\alpha_q^{(k)}$:

$$\begin{aligned} Z_q^{(0)} &= f_n\,, \\ Z_q^{(1)} &= (n-1)\Big(\tfrac{1}{2}(n+2q)f_n + f_{n-1}\Big)\,. \end{aligned} \tag{6.15}$$

Thus a system is absolutely compatible, if it is not under-determined. For such a system the strength measures the dominant term in the Hilbert polynomial, the coefficient of $r^{n-2}$.

Since the strength is usually computed for gauge systems, we should correct it for the gauge symmetry. Using the notation of Section 6.1, we obtain:

$$\begin{aligned} \bar{Z}_q^{(0)} &= \alpha_q^{(n)} - \sum_{l=0}^{p}\gamma_l\,, \\ \bar{Z}_q^{(1)} &= (n-1)\Big(\frac{1}{2}n\alpha_q^{(n)} + \alpha_q^{(n-1)} - \sum_{l=0}^{p}(\frac{1}{2}n + q + l)\gamma_l\Big)\,. \end{aligned} \tag{6.16}$$

Alternatively, we could apply Theorem 6.4 using the gauge corrected Cartan characters. Absolute compatible means now that after the gauge correction no arbitrary functions of $n$ arguments are admitted in the general solution.

Finally, we comment on some remarks by Sué [147]. He compared the index of generality (or degré d'arbitraire) (cf. Section 3.4) with the strength and came to the conclusion that the latter one is more useful when dealing with gauge system. He compares, however, a gauge corrected strength with a not gauge corrected index of generality! As soon as we correct both, the result is somewhat different. For most systems of physical interest both contain the same amount of information, as for them $\bar{\alpha}_q^{(n)} = 0$ and $\bar{\alpha}_q^{(n-1)} \neq 0$. But if we encounter a system with $\bar{\alpha}_q^{(n-1)} = 0$, then the index of generality contains further information about the lower Cartan characters which are not available using the strength. Sué did not realize that the reason to consider only the highest non-vanishing Cartan character is not a lack of information about the system but Theorem 3.12. If we know the index of generality, then we always know the Cartan genus, too. The strength, however, gives the Cartan genus only, if it does not vanish.

## 6.3 Constrained Dynamics I — Point Mechanics

The classical treatment of systems with constraints was developed by Dirac [43, 44, 45] (see also [65, 148]). The purpose of this section is to show that in the case of finite-dimensional systems his algorithm corresponds exactly to rendering the equations of motion involutive. In the language of exterior differential systems this was already noted by Hartley *et al.* [68]. We will, however, see in the next section that in the case of infinite-dimensional systems the Dirac analysis alone is in general not sufficient to prove that the field equations are consistent, as it involves only part of the completion procedure of Section 2.4.

Let the bundle $\mathcal{E}$ be given by $\mathcal{E} = Q \times T$, where $Q$ is the $N$-dimensional configuration space with coordinates $q^i$ and $T$ the time axis, and the natural projection $\mathcal{E} \to T$. A dynamical system can then be specified by a function on $J_1\mathcal{E}$, its *Lagrangian* $L(q^i, \dot{q}^i)$. For simplicity, we assume that $L$ does not depend explicitly on $t$. The dynamics of the system is determined by the condition that the action

$$S = \int L(q^i, \dot{q}^i)\, dt \tag{6.17}$$

is stationary along trajectories $q^i(t)$. It is well-known from the calculus of variation that this leads to the *Euler-Lagrange Equations*

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}^i}\right) - \frac{\partial L}{\partial q^i} = 0\,, \qquad i = 1, \ldots, N\,. \tag{6.18}$$

(6.18) represents a second-order equation whose symbol is determined by the matrix

$$M_{ij} = \frac{\partial^2 L}{\partial \dot{q}^i \partial \dot{q}^j}\,. \tag{6.19}$$

If the symbol has rank $N$, the Euler-Lagrange Equations are normal and no constraints occur. Its general solution is parameterized by $2N$ arbitrary constants.

If, however, the symbol has lower rank, it is possible to eliminate the second-order derivatives in some of the equations. Now it is no longer obvious, whether or not (6.18) is involutive. Since we are dealing with an ordinary differential equation, the symbol is always involutive. But the prolongation of the lower order equations might lead to integrability conditions, if the resulting equations are independent of the remaining second-order equations in (6.18). Then we have to check whether some of these conditions are again of lower order; in that case we have to repeat the procedure. After a finite number of iterations we will obtain either an involutive equation or an inconsistency.

We have seen in Section 6.1 that a gauge system must be under-determined. If it comprises at least as many equations as dependent variables, this implies that the system cannot be normal, i.e. there are equations of either lower order (that is for ordinary differential equations the only possibility) or of lower class. Thus we see that there is a intimate connection between gauge symmetries and constraints.

To relate this approach to constrained dynamics to the classical one by Dirac we must pass by a Legendre transformation to the Hamiltonian formulation. At the level of the differential equations this means that instead of the configuration space $Q$ the phase space $P$ is used to construct $\mathcal{E}$. In other words, we introduce $N$ additional dependent variables $p_i$ and transform (6.18) into a first-order equation

$$\mathcal{R}_1 : \begin{cases} \dot{p}_i = \dfrac{\partial L}{\partial q^i}\,, \\[2ex] p_i = \dfrac{\partial L}{\partial \dot{q}^i}\,, \end{cases} \qquad i = 1, \ldots, N\,. \tag{6.20}$$

The first set of equations consists of course just of the Euler-Lagrange Equations with the second set of equations plugged in.

If the matrix $M_{ij}$ has full rank, the second set of equations can be solved for the $\dot{q}^i$ and (6.20) is a normal equation. Otherwise we obtain some algebraic equations of the form $\phi_\ell(q,p) = 0$. Dirac called these *primary constraints*. Just as in the Lagrangian formulation, involution of (6.20) depends on the behavior of their prolongation[1]

$$D_t\phi_\ell = \left(\frac{\partial\phi_\ell}{\partial q^i}\dot{q}^i + \frac{\partial\phi_\ell}{\partial p_i}\dot{p}_i\right) = 0\,. \tag{6.21}$$

We are only interested in these equations restricted to $\mathcal{R}_1$. This corresponds to the "weak equalities" considered by Dirac.

For each of the equations in (6.21) there are three possibilities for the result of the restriction: (i) it yields an inconsistency, i.e. an equation of the form $1 = 0$; (ii) it vanishes, i.e. it becomes $0 = 0$; (iii) we obtain a new constraint, a *secondary constraint* in Dirac's terminology. We will exclude (i), as it implies that $L$ is not a valid Lagrangian. (iii) can be further divided into two cases: the secondary constraint is either algebraic or it can be solved for a velocity $\dot{q}^j$ yielding an additional differential equation. Dirac introduced here the notion of *first* and *second class constraints*, respectively.

If secondary constraints appear, we must repeat the procedure to check the consistency of the equations of motion (6.20). After a finite number of steps we will either have found an inconsistency or we will have constructed an involutive equation of the form

$$\mathcal{R}_1^{(s)}\ :\ \begin{cases} \dot{p}_i = \dfrac{\partial L}{\partial q^i}\,, & i = 1,\dots,N\,,\\[2ex] \dot{q}^j = f^j(q,\dot{q},p)\,, & j = 1,\dots,M\,,\\[2ex] \phi_\ell(q,p) = 0\,, & \ell = 1,\dots,K\,. \end{cases} \tag{6.22}$$

To make contact with the classical Dirac algorithm we introduce the so-called *canonical Hamiltonian*

$$H_C = p_i\dot{q}^i - L(q,\dot{q})\,. \tag{6.23}$$

Its differential is given by

$$dH_C = \left[\dot{q}^i dp_i - \frac{\partial L}{\partial q^i}dq^i + \left(p_i - \frac{\partial L}{\partial \dot{q}^i}\right)d\dot{q}^i\right]\;; \tag{6.24}$$

thus on $\mathcal{R}_1$ we obtain

$$dH_C\Big|_{\mathcal{R}_1} = \left(\dot{q}^i dp_i - \frac{\partial L}{\partial q^i}dq^i\right)\,. \tag{6.25}$$

This implies that on $\mathcal{R}_1$ we can substitute $\dot{q}^i$ by $\partial H_C/\partial p_i$ and $\dot{p}_i$ by $-\partial H_C/\partial q^i$. This allows to express the prolongation of the constraints in the familiar way through *Poisson brackets*

$$D_t\phi_\ell\Big|_{\mathcal{R}_1} = \left(\frac{\partial\phi_\ell}{\partial q^i}\frac{\partial H_c}{\partial p_i} - \frac{\partial\phi_\ell}{\partial p_i}\frac{\partial H_c}{\partial q^i}\right) = \{\phi_\ell, H_C\}\,. \tag{6.26}$$

We see that indeed the Dirac algorithm coincides with the completion algorithm outlined in Section 2.4. Note that it is of no importance here whether we use the canonical Hamiltonian (6.23) or the so-called total one which contains in addition the primary constraints, as we always work on $\mathcal{R}_1$ where both coincide.

[1] We use from now the Einstein convention that a summation is understood over repeated indices, if not explicitly stated otherwise.

One of the fundamental goals in constrained dynamics is to count the number of degrees of freedom of the system. We will give now an alternative method using the formal analysis of (6.22) to express this number by intrinsic quantities. We have $\beta_1^{(1)} = N + M$ and hence $\alpha_1^{(1)} = N - M$. Thus the general solution of (6.22) depends on $N - M$ arbitrary functions.

In the classical language this arbitrariness stems from the gauge transformations generated by the first class constraints. To obtain the number of degrees of freedom we must count the number of constants necessary to characterize a physical state. Since $\dim J_1\mathcal{E} = 4N$, we obtain $\dim \mathcal{R}_1^{(s)} = 3N - M - K$. Thus in a power series expansion of the general solution this number of zeroth and first order coefficients can be chosen arbitrarily.

If we identify the arbitrary functions with $\dot{q}^i$ for $i > M$, as these are not restricted by (6.22), then we must subtract $N - M$ constants coming from the arbitrary functions. Thus an initial state at $t = t_0$ is specified by $2N - K$ constants. Depending on the choice of the arbitrary functions we will, however, obtain different values for $q^i(t_1)$ and $p_i(t_1)$ at some later instant $t_1$. As these correspond nevertheless, by definition, to the same physical state, we must subtract further $N - M$ constants for the gauge symmetry. Thus a physical state is specified by $N + M - L$ constants and the number of degrees of freedom $F$ is half of this number

$$F = \frac{1}{2}(N + M - K)\,. \tag{6.27}$$

Expressing $M$ by $N$ and $\alpha_1^{(1)}$ and similarly $K$ by $N$, $\alpha_1^{(1)}$ and $\dim \mathcal{R}_1^{(s)}$, we obtain an intrinsic expression for $F$ independent of any specific representation of the manifold $\mathcal{R}_1^{(s)}$

$$F = \frac{1}{2}\dim \mathcal{R}_1^{(s)} - \alpha_1^{(1)}\,. \tag{6.28}$$

Note that this expression remains valid, if we use instead of the Hamiltonian equations of motion (6.20) the Euler-Lagrange Equations (6.18), as we will obtain exactly the same dimension and Cartan character for the final involutive equation no matter whether we work in first- or second-order because of the different dimensions of the base spaces.

Classically $F$ is determined by the number of first and second class constraints, $N_F$ and $N_S$, respectively. These numbers can easily be written in intrinsic terms, too. Comparing $F = N - N_F - 1/2N_S$ with (6.28) yields at once

$$N_F = \alpha_1^{(1)}\,, \quad N_S = 2N - \dim \mathcal{R}_1^{(s)}\,. \tag{6.29}$$

By the same argument as above this result remains valid in the Lagrangian formalism.

**Example 6.5** Consider the classical problem of a particle whose movement is restricted to the surface of a sphere in a $D$-dimensional space but otherwise free, often also called the rigid rotator [52]. Without loss of generality we can take the radius of the sphere as one and start with the Lagrangian

$$L = \frac{m}{2}\dot{q}^2 + \lambda(q^2 - 1) \tag{6.30}$$

where $\lambda$ is a multiplier. The canonically conjugate momenta $p_i$, $\pi$ are given by

$$\begin{aligned} p_i &= m\dot{q}_i\,, \qquad i = 1, \dots, D\,, \\ \pi &= 0\,. \end{aligned} \tag{6.31}$$

If we introduce a further multiplier $\mu$, we can write the total Hamiltonian as

$$H_T = \frac{1}{2m}p^2 - \lambda(q^2 - 1) + \mu\pi \tag{6.32}$$

Obviously there is one primary constraint, namely $\pi = 0$. The next steps of the Dirac algorithm lead to the constraints $q^2 = 1$, $pq = 0$, and finally $p^2 = -2m\lambda$. It is easy to see that all these constraints are second class. The system contains thus $D-1$ degrees of freedom which can be formally calculated by subtracting from the dimension of the configuration space — $D$ coordinates $q_i$ plus one coordinate $\lambda$ — half the number of second class constraints, i.e. 2.

The last constraint plus the primary one allow us to eliminate $\lambda$ and its conjugate momentum $\pi$ from the theory. The other two could in principle be solved for some conjugate pair $q_i, p_i$. We will, however, compute instead for later use the corresponding Dirac brackets [45]. A straightforward calculation yields

$$\begin{aligned} \{q_i, q_j\}_* &= 0\,, \\ \{q_i, p_j\}_* &= \delta_{ij} - q_i q_j\,, \\ \{p_i, p_j\}_* &= q_j p_i - p_j q_i\,. \end{aligned} \tag{6.33}$$

Now we will obtain the number of degrees of freedom using a formal analysis of the Euler-Lagrange Equations

$$\mathcal{R}_2 : \begin{cases} m\ddot{q}_i - 2\lambda q_i = 0\,, \\ q^2 - 1 = 0\,. \end{cases} \tag{6.34}$$

The completion to an involutive equation requires four projections leading to the integrability conditions $q\dot{q} = 0$, $q\ddot{q} + \dot{q}^2 = 0$, $\dot{\lambda} = 0$ and finally $\ddot{\lambda} = 0$. After some trivial manipulations we have thus the involutive equation

$$\mathcal{R}_2^{(4)} : \begin{cases} m\ddot{q}_i - 2\lambda q_i = 0\,, \\ \ddot{\lambda} = 0\,, \\ m\dot{q}^2 + 2\lambda = 0\,, \\ q\dot{q} = 0\,, \\ \dot{\lambda} = 0\,, \\ q^2 - 1 = 0\,. \end{cases} \tag{6.35}$$

It is easy to see that this represents a finite type equation and thus there are no first class constraints. Since $\dim \mathcal{R}_2^{(4)} = 2D - 2$, (6.28) yields $F = D - 1$ degrees of freedom. Furthermore we obtain from (6.29) that there are four second class constraints. Both results are in perfect agreement with the Dirac analysis.

## 6.4 Constrained Dynamics II — Field Theory

In field theories the equations of motion become partial differential equations. Thus involution is now a more complicated concept. The prolongation of lower order equations is no longer the only way to generate integrability conditions. Constraints are now mostly equations of lower class. In a typical field theory the base space $X$ of independent variables is a $D$-dimensional space-time. We can identify the variable $x^D$ with time. Thus simplified spoken, the equations of class $D$ are the evolutionary ones; the remaining ones are constraints.

The relation between the Dirac algorithm and our completion procedure becomes now much less clear than in the case of point mechanics. Dirac prolongs all constraints only with respect to time. If all constraints are of class $D-1$, this corresponds to our approach, because by Proposition 2.9 it suffices to analyze the prolongations with respect to the non-multiplicative variables and we find in this case only one, namely $x^D$.

But now the question arises what happens if constraints of lower class appear. Then prolongations with respect to the other multiplicative variables, i.e. with respect to some spatial coordinates, may lead to additional integrability conditions not considered by the Dirac algorithm.

**Example 6.6** In order to exhibit this effect in "pure form" we begin with a highly unphysical example without any kinematical term in the Lagrangian density. But we will later show that this is not the important point. Consider the system described by the Lagrangian density

$$\mathcal{L} = \mu[\partial_x \phi - f(\phi)] + \lambda[\partial_y \phi - g(\phi)] \tag{6.36}$$

on a three-dimensional space-time with coordinates $x, y, t$. $\mu, \lambda$ are two multipliers, $f, g$ are fixed but so far arbitrary functions. Variation with respect to $\phi$ yields the equation

$$\partial_x \mu + \partial_y \lambda + f'(\phi)\mu + g'(\phi)\lambda = 0\,. \tag{6.37}$$

More interesting equations are obtained from the multipliers. They generate an over-determined system for $\phi$

$$\begin{aligned} \partial_x \phi - f(\phi) &= 0\,, \\ \partial_y \phi - g(\phi) &= 0\,. \end{aligned} \tag{6.38}$$

Obviously this system is consistent, if and only if $f, g$ satisfy the equation

$$f'g = fg'\,. \tag{6.39}$$

This requires that $f$ is a multiple of $g$. It is easy to see that under this condition the Euler-Lagrange Equations are involutive.

Now we look at the outcome of the Dirac algorithm. Obviously all three canonically conjugate momenta $\pi_\phi, \pi_\mu, \pi_\lambda$ vanish and represent thus the primary constraints. The total Hamiltonian density is given by

$$\mathcal{H}_T = -\mu[\partial_x \phi - f(\phi)] - \lambda[\partial_y \phi - g(\phi)] + u\pi_\phi + v\pi_\mu + w\pi_\lambda\,. \tag{6.40}$$

This yields the following secondary constraints

$$\begin{aligned} \{\pi_\phi, \mathcal{H}_T\} &= -[\partial_x \mu + \partial_y \lambda + f'(\phi)\mu + g'(\phi)\lambda]\,, \\ \{\pi_\mu, \mathcal{H}_T\} &= \partial_x \phi - f(\phi)\,, \\ \{\pi_\lambda, \mathcal{H}_T\} &= \partial_y \phi - g(\phi)\,. \end{aligned} \tag{6.41}$$

Thus we obtain exactly the Euler-Lagrange Equations above. But note the crucial difference in the further analysis. Following Dirac we look only whether these equations are *algebraically* related, i.e. whether one vanishes, if we take the others into account. But this does not happen here.

Since all constraints are second class, we continue to compute the tertiary constraints in order to fix the multipliers $u, v, w$ introduced with the total Hamiltonian density. They are

$$\begin{aligned} &\partial_x v + \partial_y w + f''(\phi)\mu u + f'(\phi)v + g''(\phi)\lambda u + g'(\phi)w = 0\,, \\ &\partial_x u - f'(\phi)u = 0\,, \\ &\partial_y u - g'(\phi)u = 0\,. \end{aligned} \tag{6.42}$$

We find now the interesting phenomenon that although there are six second class constraints for three degrees of freedom $\phi, \mu, \lambda$, they do not fix all multipliers. $v, w$ appear only in the first equation, hence one of them can be chosen arbitrarily. $u$ is the solution of an over-determined system which is consistent, if $f$ and $g$ satisfy (6.39).

This example shows that the real failure of the Dirac algorithm lies in the purely algebraic treatment of the constraints. In point mechanics we could always substitute one constraint in another one, if both contained the same coordinate $q^i$ or a (time) derivative of it. This is no longer possible in field theories, because there might be derivatives with respect to different coordinates present. To check the consistency of such constraints requires the analysis of integrability conditions. But then the question arise why treat the time derivatives different from the spatial ones? It is conceptually simpler to treat everything on the same footing.

**Example 6.6 (cont.)** One might argue that (6.36) is a rather peculiar Lagrangian density. But the addition of a kinetic term for $\phi$ does not really change the outcome, although the computations are slightly more complicated due to the appearance of further integrability conditions. Take

$$\bar{\mathcal{L}} = \frac{1}{2}(\partial_t\phi)^2 + \mathcal{L}\,. \tag{6.43}$$

Its Euler-Lagrange Equations are

$$\begin{aligned} &\partial_{tt}\phi + \partial_x\mu + \partial_y\lambda + f'(\phi)\mu + g'(\phi)\lambda = 0\,,\\ &\partial_x\phi - f(\phi) = 0\,,\\ &\partial_y\phi - g(\phi) = 0\,. \end{aligned} \tag{6.44}$$

Only the first equation has changed. But since the only second-order derivative is one of $\phi$, our completion procedure generates besides (6.39) two further integrability conditions involving second-order derivatives of $\mu, \lambda$. We omit them here, because they are rather complicated.

We also do not show the Dirac analysis of this system. It suffices to note that the two new conditions (in Hamiltonian form) are also found by the Dirac algorithm. But this was to be expected, because their construction requires prolongations with respect to time. Only (6.39) is again overlooked, for no spatial cross-derivatives are performed.

The natural question now is whether systems of this form are unphysical for some reason or whether this effect occurs often and might lead to a wrong analysis. Such purely spatial cross-derivatives (or to be more precise: linear combinations of spatial prolongations) are only necessary, if the field equations form an over-determined system. This might appear at first sight quite unusual but one can find such systems in the literature. A typical example is a gauge fixing condition built into the Lagrangian density.

On the other hand, it is difficult to overlook this effect, for it shows of course in the constraints. It was already mentioned above that in principle one could still use the Dirac algorithm but that one must augment it with a kind of involution analysis, if constraints like in Example 6.6 appears. But again: why not use the involution analysis from the very beginning?

Another point which might be worth while mentioning concerns covariance. It is well-known that the introduction of a Hamiltonian breaks covariance, as a time coordinate must be explicitly chosen. But using our involution based approach it is at least possible to exhibit the constraint structure in a covariant way by completing the Euler-Lagrange Equation.

Given a Lagrangian density $\mathcal{L}(\phi_a, \partial_\mu\phi_a)$ the Legendre transformation is usually performed by introducing the momenta

$$\pi^a = \frac{\partial\mathcal{L}}{\partial\partial_0\phi_a}\,. \tag{6.45}$$

In general, the Hamiltonian field equations are then still second order because of the spatial derivatives. To obtain a first-order description one must define additional "momenta" for them. But this can be done covariantly by writing

$$\pi^a_\mu = \frac{\partial\mathcal{L}}{\partial\partial_\mu\phi_a}\,. \tag{6.46}$$

This transformation yields a "covariant form" of the Hamiltonian field equations which can then be completed until involution.

## 6.5 Constrained Dynamics III — Symplectic Formalism

To conclude our analysis of systems with constraints we want to comment on the so-called *Hamiltonian* or *symplectic formalism* of Faddeev and Jackiw [51]. It was first applied by Floreanini and Jackiw [54] for the quantization of self-dual fields in two dimensions. Their circumvention of the classical Dirac algorithm sparked some discussion [12, 32] whether or not it is correct and necessary, respectively. For simplicity we will restrict most of our exposition to finite-dimensional systems.

This formalism is applicable to first-order Lagrangians, i.e. they are linear in the velocities:

$$L = a_i(q)\dot{q}^i - V(q)\,. \tag{6.47}$$

The corresponding Euler-Lagrange Equations are

$$\mathcal{R}_1\,:\,\left\{A_{ij}\dot{q}^j - \frac{\partial V}{\partial q^i} = 0\,,\quad i = 1,\ldots,N\right. \tag{6.48}$$

where the matrix $A$ is given by cross-derivatives of the $a_i$

$$A_{ij} = \frac{\partial a_j}{\partial q^i} - \frac{\partial a_i}{\partial q^j}\,. \tag{6.49}$$

Obviously, $\mathcal{R}_1$ is an involutive first-order equation provided this matrix is non-singular. Hence from the formal point of view there is no need to introduce any constraints; the system is unconstrained.

If one follows the Dirac algorithm, one is forced to introduce $N$ constraints, as all canonically conjugate momenta are independent of the velocities. Mathematically this corresponds to insisting on treating (6.48) as a *second-order* differential equation. Since $\mathcal{R}_1$ is involutive, its prolongation is involutive, too, and no secondary constraints arise. Furthermore, all constraints are second class, for (6.48) is of course a finite-type equation.

The decisive observation of Faddeev and Jackiw [51] was that it is unnecessary to introduce the canonically conjugate momenta, because (6.48) has already the structure of a Hamiltonian system with Hamiltonian $V$ provided we define the Poisson bracket by

$$\left\{q^j, q^i\right\} = (A^{-1})^{ij}\,. \tag{6.50}$$

This definition makes sense as long as the matrix $A$ is non-singular. Cronström and Noga [35] demonstrated recently, how one can explicitly construct the corresponding canonical coordinates using essentially only linear algebra.

We will show later that exactly the same bracket structure arises, if one follows Dirac and eliminates the second class constraints using Dirac brackets. This was already noted by Govaerts [61] and earlier explicitly demonstrated for the special Lagrangian treated in [54] by Costa and Girotti [32].

In the case that the matrix $A$ is singular, we have "true" constraints. Faddeev and Jackiw resort then to Darboux's theorem to eliminate them explicitly. If this elimination is too complicated, they recommend to apply the usual Dirac algorithm. At least for the construction of all constraints, there is no need for this. Since we are dealing with a system of ordinary differential equations, we have always an involutive symbol and our completion algorithm will never require a prolongation. Thus we can always obtain an involutive, *first-order* equation.

The only point is that it is now no longer possible to use the matrix $A$ to define a symplectic structure. But even if one wants to follow Dirac's approach, it is computationally much simpler to complete a first-order system and to prolong it to second-order afterwards than the other way round. The true constraints are algebraic equations even in the Lagrangian formalism, and the other constraints are just prolongations of them.

On the other hand it depends on the constraint structure whether one gains something by transforming a second-order Lagrangian into a first-order one by extending the configuration space. This amounts essentially to transforming the Euler-Lagrange Equations (6.18) into a first-order equation along the lines of Appendix C.2. One obtains of course the same number of degrees of freedom, since this reduction does not change the Cartan characters. Computationally one might save a few steps compared to the Dirac algorithm, especially if second class constraints are present.

For systems having only second class constraints one can use an idea of Barcelos-Neto and Wotzasek [7, 8] to compute the Dirac brackets in a direct manner. Thus we consider now the case that the matrix $A$ has not maximal rank. Let a basis of its null space be given by the vectors $v^i_\alpha$, i.e.

$$v^i_\alpha(q)A_{ij}(q) = 0\,, \quad \alpha = 1,\dots,M^{\,2}\,. \tag{6.51}$$

This implies the existence of $M$ "primary" constraints

$$\phi_\alpha = v^i_\alpha \frac{\partial V}{\partial q^i} = 0\,. \tag{6.52}$$

Following Barcelos-Neto and Wotzasek we introduce $M$ additional coordinates $\lambda^\alpha$ and continue with the modified Lagrangian

$$\tilde{L} = \left(a_i + \lambda^\alpha \frac{\partial \phi_\alpha}{\partial q^i}\right)\dot{q}^i - V\,. \tag{6.53}$$

The new equations of motion are

$$\tilde{\mathcal{R}}_1 : \begin{cases} A_{ij}\dot{q}^j + \dfrac{\partial \phi_\alpha}{\partial q^i}\dot{\lambda}^\alpha - \dfrac{\partial V}{\partial q^i} = 0\,, \\ \dfrac{\partial \phi_\alpha}{\partial q^i}\dot{q}^i = 0\,. \end{cases} \tag{6.54}$$

Let us assume that the second set of equations is independent of the original equations of motions, i.e. that the prolongation of the constraints leads to new ones. (6.54) can be written again in the form of (6.48) on the enlarged configuration space, if we introduce the "deformed" $(N+M)\times(N+M)$ matrix $\tilde{A}$

$$\begin{aligned} \tilde{A}_{ij} &= \frac{\partial a_i}{\partial q^j} - \frac{\partial a_j}{\partial q^i}\,, \\ \tilde{A}_{i\alpha} &= -\frac{\partial \phi_\alpha}{\partial q^i}\,, \\ \tilde{A}_{\alpha\beta} &= 0\,. \end{aligned} \tag{6.55}$$

If the matrix $\tilde{A}$ is non-singular, Barcelos-Neto and Wotzasek use it to define a Poisson structure. Otherwise they repeat the process. There are two possible outcomes: either one obtains after a finite number of iterations a regular matrix or the process stops, because the constraints do not generate new ones. In the latter case gauge symmetries, i.e. first class

[2] It could of course happen, that $M$ varies on the configuration space. But we will assume that it is constant.

constraints, are present. We will concentrate on the first case and assume for simplicity that already $\tilde{A}$ has maximal rank.

From the point of view of formal theory this assumption implies that the modified Euler-Lagrange Equations $\tilde{\mathcal{R}}_1$ are involutive. Indeed they are now even normal, as contraction of the first set of equations with the vectors $v^i_\alpha$ leads to the equations

$$v^i_\alpha \frac{\partial \phi_\beta}{\partial q^i}\dot{\lambda}_\beta - \phi_\alpha = 0\,. \tag{6.56}$$

The condition on the rank of $\tilde{A}$ implies that the matrix

$$B_{\alpha\beta} = v^i_\alpha \frac{\partial \phi_\beta}{\partial q^i} \tag{6.57}$$

has maximal rank. Thus we can obtain the solved form (4.15) used in the Cauchy-Kowalevsky Theorem by a simple linear transformation.

To prove that the inverse of $\tilde{A}$ yields the correct Dirac brackets we compute them in the standard way. As already mentioned we find $N$ primary constraints

$$\psi_i = p_i - a_i(q) = 0\,. \tag{6.58}$$

The total Hamiltonian of the system is given by

$$H_T = V(q) + \mu^i \psi_i\,. \tag{6.59}$$

The primary constraints lead to the consistency conditions

$$\{\psi_i, H_T\} = A_{ij}\mu^j - \frac{\partial V}{\partial q^i} = 0\,. \tag{6.60}$$

If the matrix $A$ is non-singular, these secondary constraints can be used to determine all multipliers $\mu^i$ and the Dirac algorithm stops here. In the case of a singular matrix we can solve only for some of them, as contraction with $v^i_\alpha$ yields the "true" constraints $\phi_\alpha$. Their consistency conditions

$$\{\phi_\alpha, H_T\} = \mu^i \frac{\partial \phi_\alpha}{\partial q^i} \tag{6.61}$$

allow us to determine the remaining multipliers under the above made assumptions. Thus tertiary constraints do not appear.

To determine the Dirac bracket we must compute the structure constants of the Poisson algebra generated by the constraints

$$\begin{aligned} \{\psi_i, \psi_j\} &= A_{ij}\,,\\ \{\psi_i, \phi_\alpha\} &= -\frac{\partial \phi_\alpha}{\partial q^i}\,,\\ \{\phi_\alpha, \phi_\beta\} &= 0\,. \end{aligned} \tag{6.62}$$

Note that despite its appearance this is a closed algebra under our assumptions. In the right hand sides we rediscover the matrix $\tilde{A}$ defined above by the modified Lagrangian $\tilde{L}$.

**Example 6.5 (cont.)** To treat the rigid rotator with this formalism we must first rewrite its Lagrangian as a first order one. This is most easily done by using the momenta as additional "configuration" space coordinates. Furthermore it is simpler to replace the multiplier $\lambda$ by the time derivative of another one [8] (this corresponds to adding a total derivative). We will, however, continue to call it $\lambda$ and obtain thus the Lagrangian

$$L = p\dot{q} + \dot{\lambda}(q^2 - 1) - \frac{1}{2}p^2\,. \tag{6.63}$$

It leads to the constraint $pq = 0$ and to the modified Lagrangian $\tilde{L} = L + \dot{\mu}pq$. The corresponding matrix $\tilde{A}$ is given by (if we label the columns $q_i, p_i, \lambda, \mu$)

$$\tilde{A} = \begin{pmatrix} 0 & -\delta_{ij} & q_i & p_i \\ \delta_{ij} & 0 & 0 & q_i \\ -q_j & 0 & 0 & 0 \\ -p_j & -q_j & 0 & 0 \end{pmatrix} \tag{6.64}$$

and is obviously non-singular. Thus this approach is computationally much simpler than the Dirac algorithm. Inversion of $\tilde{A}$ yields exactly the Dirac brackets (6.33).

Barcelos-Neto and Wotzasek omit in their papers [7, 8] a discussion of the precise relation between the equations of motion (6.48) derived from the original Lagrangian $L$ and those (6.54) obtained from the modified one $\tilde{L}$. This is, however, quite important for understanding the meaning of the multipliers $\lambda^\alpha$.

For this purpose we introduce vectors $w^i_\rho$, $\rho = M+1, \dots, N$ such that $\{v^i_\alpha, w^i_\rho\}$ is a linearly independent set. Under the above mentioned assumptions we find the following local representation of the equation $\mathcal{R}^{(1)}_1$ obtained from completing (6.48) consisting only of independent equations

$$\mathcal{R}^{(1)}_1 \,:\, \begin{cases} w^i_\rho\left(A_{ij}\dot{q}^j - \dfrac{\partial V}{\partial q^i}\right) = 0\,, \\ \dfrac{\partial \phi_\alpha}{\partial q^i}\dot{q}^i = 0\,, \\ \phi_\alpha = 0\,. \end{cases} \tag{6.65}$$

Similarly we obtain a local description of $\tilde{\mathcal{R}}_1$ containing under our assumptions only independent equations, too,

$$\tilde{\mathcal{R}}_1 \,:\, \begin{cases} w^i_\rho\left(A_{ij}\dot{q}^j + \dfrac{\partial \phi_\alpha}{\partial q^i}\dot{\lambda}^\alpha - \dfrac{\partial V}{\partial q^i}\right) = 0\,, \\ \dfrac{\partial \phi_\alpha}{\partial q^i}\dot{q}^i = 0\,, \\ v^i_\beta\dfrac{\partial \phi_\alpha}{\partial q^i}\dot{\lambda}^\alpha + \phi_\beta = 0\,. \end{cases} \tag{6.66}$$

Although both represent finite type equations note the crucial difference that $\tilde{\mathcal{R}}_1$ is normal and does not contain algebraic equations. This entails especially a relaxation of the constraints: we do no longer require that $\phi_\alpha = 0$ but only $\phi_\alpha = const$! Furthermore while the solution space of $\mathcal{R}^{(1)}_1$ is $(N-M)$-dimensional, $\dim \tilde{\mathcal{R}}_1 = N + M$. But this discrepancy is removed as soon as we require that $\lambda^\alpha(t) \equiv const$, because then both systems are identical and possess of course the same solutions.

Thus we can conclude that the geometric idea behind this approach is the embedding of the original system into another one living in a larger phase space. Projecting the trajectories of the latter one on a leave of the foliation $\lambda^\alpha = const$ recovers the solutions of the former one. The choice of the leave is not important, for $\tilde{\mathcal{R}}_1$ is invariant under translations in $\lambda^\alpha$.

**Example 6.7** Consider a system with a three-dimensional configuration space whose dynamics is determined by the Lagrangian [6, 91]

$$L = (q_2 + q_3)\dot{q}_1 + k\dot{q}_3 + \frac{1}{2}(k^2 - 2q_2q_3 - q_3^2) \tag{6.67}$$

where $k$ is a constant. In the formal analysis we start with the Euler-Lagrange Equations. After a trivial algebraic manipulation they can be written as

$$\mathcal{R}_1 : \begin{cases} \dot{q}_2 + \dot{q}_3 = 0\,, \\ \dot{q}_1 - q_3 = 0\,, \\ q_2 = 0\,. \end{cases} \tag{6.68}$$

Obviously this equation becomes involutive after one projection adding the integrability condition $\dot{q}_2 = 0$. It is also trivial to integrate it in closed form

$$q_1(t) = at + b\,, \quad q_2(t) = 0\,, \quad q_3(t) = a\,. \tag{6.69}$$

For the Dirac analysis we need the canonically conjugate momenta

$$p_1 = q_2 + q_3\,, \quad p_2 = 0\,, \quad p_3 = k\,. \tag{6.70}$$

These are at the same time the primary constraints. The total Hamiltonian is

$$H_t = \frac{1}{2}(q_3^2 + 2q_2q_3 - k^2) + \mu_1(p_1 - q_2 - q_3) + \mu_2 p_2 + \mu_3(p_3 - k)\,. \tag{6.71}$$

This leads to the secondary constraints $\mu_2 + \mu_3 = 0$, $q_3 = \mu_1$ and $q_2 = 0$. The tertiary constraints are $\mu_2 = \mu_3 = 0$. Hence we have four second class constraints $p_2 = q_2 = 0$ and $p_3 = k, q_3 = p_1$. By direct inspection one sees that there is only one dynamical degree of freedom, namely $(q_1, p_1)$, having as Dirac bracket its standard Poisson bracket.

In the symplectic formalism one finds in the first step the "true" constraint $q_2 = 0$ and continues with the modified Lagrangian $\tilde{L} = L + \lambda\dot{q}_2$. From it we derive the matrix $\tilde{A}$

$$\tilde{A} = \begin{pmatrix} 0 & -1 & -1 & 0 \\ 1 & 0 & 0 & -1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \tag{6.72}$$

where the columns and rows are labeled by $q_1, q_2, q_3, \lambda$. This matrix is obviously invertible and yields the correct bracket, if we identify $q_3$ with the momentum $p_1$. If we compute, however, the equations of motion, we obtain after some trivial manipulations

$$\tilde{\mathcal{R}}_1 : \begin{cases} \dot{q}_3 = 0\,, \\ \dot{q}_2 = 0\,, \\ \dot{q}_1 - q_2 - q_3 = 0\,, \\ \dot{\lambda} - q_2 = 0\,. \end{cases} \tag{6.73}$$

As expected the equation $q_2 = 0$ is missing and the solution space is consequently larger

$$q_1(t) = (\bar{a} + \bar{b})t + \bar{c}\,, \quad q_2(t) = \bar{b}\,, \quad q_3(t) = \bar{a}\,, \quad \lambda(t) = \bar{b}t + \bar{d}\,. \tag{6.74}$$

However, as soon as we require that $\lambda(t) \equiv const$ we recover the correct solution, as this implies $\bar{b} = 0$.

This example also serves well to demonstrate the difference between first- and second-order systems. Actually it stems from a slightly more complicated example where $k$ is not treated as constant but as an additional coordinate [75]. But the real starting point is the following system of second-order equations

$$\ddot{x} = -\dot{y}\,, \quad \ddot{y} = -y\,. \tag{6.75}$$

One can prove that these equations neither are nor can be transformed into the Euler-Lagrange Equations of some Lagrangian [76]. But if we rewrite the system as a first-order one

$$\dot{x} = z\,, \quad \dot{y} = w\,, \quad \dot{z} = -w\,, \quad \dot{w} = -y\,, \tag{6.76}$$

then a Lagrangian exist, namely

$$L = (y+z)\dot{x} + w\dot{z} + \frac{1}{2}(w^2 - 2yz - z^2)\,. \tag{6.77}$$

A normal form can be obtained by setting

$$q^1 = x\,, \quad q^2 = z\,, \quad p_1 = y+z\,, \quad p_2 = w\,. \tag{6.78}$$

Note, however, that we are now using as configuration space coordinates $x$ and $\dot{x}$, if we compare with the original second-order system. In first-order systems the distinction between configuration and phase space begins to blur. This is the basis of the symplectic method.

Barcelos-Neto and Wotzasek [7] apply their method also to field theories. But again there are problems with systems like the ones discussed in Example 6.6. Now they are worse, because the algorithm does no longer necessarily terminate even if there are only second class constraints present. The reason is obviously that only the prolongation of the constraints with respect to time is considered in the construction of the "deformed" Lagrangian density, as it contains only the time derivative of the new multiplier appears. However, it might be possible to design a modification of this approach which also uses the spatial derivatives of the multipliers.

**Example 6.6 (cont.)** Consider the following slight modification of the first-order Lagrangian density (6.36)

$$\mathcal{L} = \nu\partial_t\phi + \mu[\partial_x\phi - f(\phi)] + \lambda[\partial_y\phi - g(\phi)]\,. \tag{6.79}$$

$\nu$ is an additional multiplier. If we order the fields $\phi, \nu, \mu, \lambda$ the matrix $A$ is given by

$$A = \begin{pmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}\,. \tag{6.80}$$

Thus there exist two eigenvectors and we obtain the two obvious constraints

$$\Omega_1 = \partial_x\phi - f(\phi)\,, \quad \Omega_2 = \partial_y\phi - g(\phi)\,. \tag{6.81}$$

Following the same steps as in the finite-dimensional case, we are lead to consider the modified Lagrangian density

$$\tilde{\mathcal{L}} = \nu\partial_t\phi + \partial_t\alpha[\partial_x\phi - f(\phi)] + \partial_t\beta[\partial_y\phi - g(\phi)] \tag{6.82}$$

where $\alpha, \beta$ are the new multipliers. The disappearance of the multipliers $\mu, \lambda$ is not surprising, because they remained completely arbitrary in the previous Euler-Lagrange Equations. But we have gained nothing by this deformation, as the new matrix $\tilde{A}$ still has only rank 2 and the only difference to the previous field equations is that we get now the prolongations of the constraints with respect to time. However, we know from our previous discussion that these are unimportant.

## 6.6 Yang-Mills Theory

In this and the following sections we will change our notation and use the standard notions of theoretical physics. This includes the Einstein convention that summation over repeated indices is always understood if not stated explicitly otherwise. Indices $\mu, \nu, \dots$ denote now no longer multi-indices but space-time indices; $a, b, c, \dots$ are used for Lie algebra indices. Since the metric is usually not important in the formal analysis, we will sometimes be a bit sloppy with raising and lowering indices. Most of the time we will assume that the metric has the form $\mathrm{diag}(\eta_\mu)$ without fixing the signature.

We consider the field equations of a Yang-Mills theory with a $d$-dimensional gauge group in an $n$-dimensional space-time. The dependent variables are the $m = dn$ components $A^a_\mu$ of the vector potential. Using the field strengths $F^a_{\mu\nu} = \partial_\mu A^a_\nu - \partial_\nu A^a_\mu + C^a_{bc} A^b_\mu A^c_\nu$ where $C^a_{bc}$ are the structure constants of the Lie algebra of the gauge group we get $dn$ field equations

$$\phi^a_\nu[A^b_\mu] = \partial_\mu F^a_{\mu\nu} + C^a_{bc} A^b_\mu F^c_{\mu\nu} = 0\,, \quad \begin{cases} a = 1, \dots, d\,, \\ \nu = 1, \dots, n\,. \end{cases} \tag{6.83}$$

In order to check for involution and to compute the Cartan characters, we must first analyze the symbol or the principal part of (6.83). It is given by $\partial_{\mu\mu} A^a_\nu - \partial_{\mu\nu} A^a_\mu$. Thus for $\nu \neq n$ we can always take $\partial_{nn} A^a_\nu$ as principal derivative. If however $\nu = n$, then this term disappears and we get $\partial_{n,n-1} A^a_{n-1}$ as principal derivative. This yields the values $\beta^{(n)}_{YM} = d(n-1)$, $\beta^{(n-1)}_{YM} = d$. All other $\beta^{(k)}_{YM}$ vanish. Note that this result is indeed not influenced by the metric.

Since in principle the maximal possible value for $\beta^{(n)}_{YM}$ is $dn$, we must check carefully whether the lower value obtained is intrinsic or only the consequence of a not $\delta$-regular coordinate system. For this purpose we determine the tableau of (6.83). Following Definition 2.23 it is given by

$$\tau_\chi \,:\, A^a_\mu \mapsto \left(\frac{\partial \phi^b_\nu}{\partial(\partial_{\tau\rho} A^a_\mu)}\right) \chi_\rho \chi_\tau A^a_\mu\,. \tag{6.84}$$

(6.83) yields for its matrix

$$\mathcal{T}^{b\mu}_{a\nu}[\chi] = \delta^b_a \Big[\delta^\mu_\nu \Big(\sum_{\rho\neq\nu} \eta_\rho \chi^2_\rho\Big) + (\delta^\mu_\nu - 1)\eta_\mu \chi_\nu \chi_\mu\Big]\,. \tag{6.85}$$

The indices $b, \nu$ label the rows, $a, \mu$ the columns. It is easy to see that for any values of $a, b, \nu$

$$\chi_\mu \mathcal{T}^{b\mu}_{a\nu}[\chi] = 0\,. \tag{6.86}$$

This equation yields $d$ identities and we obtain an upper bound for the rank of the matrix $\mathcal{T}[\chi]$, namely $d(n-1)$. But this provides at the same time an upper bound for $\beta^{(n)}_{YM}$ due to (2.27). Since our analysis above yielded this value, our coordinate system is $\delta$-regular and the computed values for $\beta^{(k)}_{YM}$ are correct.

But the identities (6.86) further implies that the prolonged symbol does not have maximal rank. Indeed at the level of the symbol, multiplication with $\chi_\mu$ corresponds to prolonging with respect to $x^\mu$. Now the prolonged symbol is a matrix with $dn^2$ rows and thus the maximal value for its rank is given by $d(n^2-1)$. On the other hand, we know from the obtained $\beta^{(k)}_{YM}$ that this is also the minimal value for its rank. Thus we can conclude that the symbol of the Yang-Mills Equations is involutive.

For the Cartan characters we get

$$\alpha^{(n)}_{YM} = d, \quad \alpha^{(n-1)}_{YM} = d(2n-1), \quad \alpha^{(n-k)}_{YM} = dn(k+1)\,. \tag{6.87}$$

We recover of course the identities (6.86) also at the level of the differential equations. Since the symbol has not maximal rank, we must show that the corresponding integrability conditions are satisfied as a consequence of the field equations (6.83). After a somewhat lengthy computation one can show that

$$\partial_\nu \phi^a_\nu + C^a_{bc} A^b_\nu \phi^c_\nu = 0 \tag{6.88}$$

holds as a consequence of the properties of the $C^a_{bc}$. Thus involution of the Yang-Mills Equations (6.83) is equivalent to $C^c_{ab}$ being the structure constants of a Lie algebra. (6.88) are often called *Noether identities*. If we introduce an external current $J^a_\nu$ at the right hand side of (6.83), then a solution can only exist, if it satisfies the compatibility condition

$$\mathcal{D}_\nu J^a_\nu = \partial_\nu J^a_\nu + C^a_{bc} A^b_\nu J^c_\nu = 0\,, \tag{6.89}$$

i.e. its covariant derivative must vanish.

Since by (6.86) every one-form $\chi$ is characteristic, the Yang-Mills Equations are underdetermined as expected. This stems of course from the invariance of (6.83) under the gauge transformations

$$\bar{A}^a_\mu = A^a_\mu + \partial_\mu \lambda^a + C^a_{bc} A^b_\mu \lambda^c \tag{6.90}$$

with $\gamma_1 = d$ gauge functions $\lambda^a$. This yields a gauge corrected strength

$$\bar{Z}^{(1)}_{YM} = 2(n-1)(n-2)d\,. \tag{6.91}$$

For simplicity, we restrict ourselves for the remainder of the section to $n = 4$. (3.7) leads to the Hilbert polynomial

$$H_{YM}(r) = 36d + \frac{73}{3}dr + \frac{9}{2}dr^2 + \frac{1}{6}dr^3\,. \tag{6.92}$$

Adjusting for the gauge symmetry (6.90) yields the corrected polynomial

$$\bar{H}_{YM}(r) = 16d + 12dr + 2dr^2 \tag{6.93}$$

and the corrected Cartan characters

$$\bar{\alpha}^{(4)}_{YM} = 0, \quad \bar{\alpha}^{(3)}_{YM} = 4d, \quad \bar{\alpha}^{(2)}_{YM} = \alpha^{(1)}_{YM} = 6d. \tag{6.94}$$

We see that the dimension $d$ of the gauge group appears everywhere linearly. Thus we get just $d$-times the values of the Maxwell Equations. This implies that the results are independent of the gauge group, as the structure constants of the algebra do not appear in the symbol. Especially it makes no difference whether we treat an Abelian or a non-Abelian theory.

To conclude the analysis of the Yang-Mills equations, we consider as an example of a gauge fixing the *Lorentz gauge* $\partial_\mu A^a_\mu = 0$. Adding this condition to the field equations (6.83) leads to a simple over-determined system with the Cartan characters

$$\alpha^{(4)}_{Lor} = 0, \quad \alpha^{(3)}_{Lor} = 4d, \quad \alpha^{(2)}_{Lor} = 12d, \quad \alpha^{(1)}_{Lor} = 16d\,. \tag{6.95}$$

A comparison with (6.94) shows that although the first two characters have the same values as the gauge corrected ones, the lower two are too large. Thus the Lorentz condition does not completely fix the gauge.

## 6.7 Electrodynamics

In the special case of the Maxwell Equations there exists also a first-order formulation using the field strengths as dependent variables. In potential formulation there are 4 dependent variables $A_\mu$ and the same number of second-order equations

$$\partial_\mu \partial_\mu A_\nu - \partial_\mu \partial_\nu A_\mu = 0\,; \tag{6.96}$$

in field strength formulation there are 6 dependent variables $F_{\mu\nu}$ and the same number of first-order equations

$$\begin{aligned} \partial_\mu F_{\mu\nu} &= 0\,, \\ \partial_{[\rho} F_{\mu\nu]} &= 0\,. \end{aligned} \tag{6.97}$$

The second set of equations represent the conditions for the existence of a potential $A_\mu$ such that

$$F_{\mu\nu} = \partial_\mu A_\nu - \partial_\nu A_\mu \tag{6.98}$$

We find $\alpha_P^{(1)} = 16, \alpha_P^{(2)} = 12, \alpha_P^{(3)} = 7, \alpha_P^{(4)} = 1$ for the vector potential formulation (6.96) and $\alpha_F^{(1)} = \alpha_F^{(2)} = 6, \alpha_F^{(3)} = 4, \alpha_F^{(4)} = 0$ for the field strength formulation (6.97). Theorem 3.9 yields for the number of arbitrary functions in an algebraic representation of a general solution $f_1 = f_2 = -1, f_3 = 5, f_4 = 1$ for (6.96) and $f_1 = f_4 = 0, f_2 = 2, f_3 = 4$ for (6.97). Thus only the field strength formulation admits such a representation of its general solution containing four arbitrary functions of three variables and two arbitrary functions of two variables. If we allow one-dimensional integrals, too, we can find for the potential formulation the following solution of the condition $H_q(r) = T_q(r)$

$$\begin{aligned} f_{1,0} = f_{2,0} = f_{3,0} = f_{4,0} = 1\,, \\ f_{2,-1} = 2,\; f_{3,-1} = 4\,. \end{aligned} \tag{6.99}$$

But this is an obvious consequence of the definition of the field strength. If we consider (6.98) as a system of differential equations for the $A_\mu$ by given $F_{\mu\nu}$, then its general solution has an algebraic representation consisting of the four functions entering algebraically in (6.99). The other six functions are just the arbitrary functions occurring in the general solution of Equation (6.97).

These four additional arbitrary functions reflect of course the gauge freedom (6.90) present in the vector potential but not in the field strength formulation. Indeed Theorem 6.1 yields with $\gamma_1 = 1$ the gauge corrections $g_1 = g_2 = g_3 = g_4 = 1$. Subtracting them leads to the same result as for the field strength formulation. More generally, we find that correcting the Cartan characters of the potential formulation for the gauge symmetry yields exactly the values $\alpha_F^{(k)}$ showing the equivalence of the two formulations.

To demonstrate the use of Lie pseudogroups, we compute now this correction intrinsically. For $d = 1$ the gauge transformation (6.90) are the general solution of the first order differential equation

$$\mathcal{G}_1 : \left\{ \begin{array}{ll} \dfrac{\partial \bar{x}^\mu}{\partial x^\nu} = \delta^\mu_\nu\,, & \dfrac{\partial \bar{x}^\mu}{\partial A_\nu} = 0\,, \\[2ex] \dfrac{\partial \bar{A}_\mu}{\partial x^\nu} = \dfrac{\partial \bar{A}_\nu}{\partial x^\mu}\,, & \dfrac{\partial \bar{A}_\mu}{\partial A_\nu} = \delta^\nu_\mu\,. \end{array} \right. \tag{6.100}$$

(For simplicity, we have allowed rigid translations of the underlying space-time). It is easy to see that this represents an involutive equation with Cartan characters

$$\alpha_G^{(k)} = \left\{ \begin{array}{ll} 0 & k = 5, \ldots, 8\,, \\ 5 - k & k = 1, \ldots, 4\,. \end{array} \right. \tag{6.101}$$

This yields for the Hilbert polynomial of the Lie pseudogroup

$$H_1^{(G)}(r) = 10 + \frac{47}{6}r + 2r^2 + \frac{1}{6}r^3 \tag{6.102}$$

and hence

$$G_2(r) = H_1^{(G)}(r+1) = 20 + \frac{37}{3}r + \frac{5}{2}r^2 + \frac{1}{6}r^3. \tag{6.103}$$

Subtracting this from the Hilbert polynomial of the potential formulation yields the gauge corrected Hilbert polynomial (6.93) which equals the Hilbert polynomial of the field strength formulation.

Using the methods outlined in Section 3.4, one could easily produce more solutions of the diophantine system (3.23). A few of them are discussed in [138]. But in the case of the field strength formulation it is easily possible to produce the general solution of (3.23). We consider only the case $J = \{0, -1\}$, i.e. representations of the solutions containing algebraic functions and one-dimensional integrals. After canceling a few unnecessary integer factors we obtain the following system

$$\begin{aligned} f_{3,0} + f_{3,1} &= 4\,, \\ 2f_{2,0} + 2f_{2,1} + 5f_{3,0} + 3f_{3,1} &= 24\,, \\ f_{1,0} + f_{1,1} + 2f_{2,0} + f_{2,1} + 3f_{3,0} + f_{3,1} &= 16\,. \end{aligned} \tag{6.104}$$

(Here we have already taken into account that the Maxwell Equations have a Cartan genus of 3). All non-negative integer solution of this system are given by

$$\begin{aligned} f_{3,0} &= 4 - f_{3,1}\,, & f_{3,1} &= 0, \dots, 4\,, \\ f_{2,0} &= 2 + f_{3,1} - f_{2,1}\,, & f_{2,1} &= 0, \dots, f_{3,1} + 2\,, \\ f_{1,0} &= f_{2,1} - f_{1,1}\,, & f_{1,1} &= 0, \dots, f_{2,1}\,. \end{aligned} \tag{6.105}$$

One can easily show that there are exactly 80 different solutions. A similar analysis of the vector potential formulation yields more than 30.000 solutions, if we take $J = \{0, -1, -2\}$!

If we are only interested in the total number of arbitrary functions with a given number of arguments, then we obtain

$$\tilde{f}_3 = 4, \quad \tilde{f}_2 = 2 + f_{3,1}, \quad \tilde{f}_1 = f_{2,1}\,. \tag{6.106}$$

Thus there are 25 different sets of values for the $\tilde{f}_k$. For the arbitrary functions of one argument, it does not make much sense to distinguish between different differentiation orders. In any case such a function contributes one arbitrary Taylor coefficient at each order. One only changes the number of arbitrary constants. Adjusting for this, we see that there are essentially only 25 different possibilities for the $f_{k,j}$.

Finally, we give a formally well-posed initial value problem for the field strength formulation. For an easier interpretation we use the electric and the magnetic fields $\mathbf{E}$ and $\mathbf{B}$ and we choose $x^4 = t$. Then one has for each field an evolutionary equation

$$\partial_t \mathbf{E} = \operatorname{curl} \mathbf{B}\,, \quad \partial_t \mathbf{B} = -\operatorname{curl} \mathbf{E} \tag{6.107}$$

and additionally the two constraints

$$\operatorname{div} \mathbf{E} = 0\,, \quad \operatorname{div} \mathbf{B} = 0\,. \tag{6.108}$$

We choose a distinguished spatial direction, say the $x$-direction, and solve the constraints for the corresponding derivatives. Thus there are two principal derivatives belonging to the $x$-components of the fields and they can be prescribed only on the $y$-$z$-plane. All other fields can be given in the whole space.

## 6.8 Gravity

We will treat Einstein's vacuum field equations for an $n$-dimensional space-time

$$R_{\mu\nu}[g_{\rho\sigma}] = 0 \tag{6.109}$$

where $R_{\mu\nu}$ denotes the Ricci tensor. If we take the components of the metric tensor $g_{\rho\sigma}$ as dependent variables, then because of the symmetries of Ricci tensor and metric, respectively, (6.109) represents $n(n+1)/2$ equations for as many dependent variables. Since two-dimensional gravity is trivial, we will assume $n > 2$.

Expanding (6.109) in a general coordinate system in terms of $g_{\mu\nu}$ leads to very complicated expressions. It is nearly impossible to check this way, whether or not the Einstein equations form an involutive system. A closer look at Chapter 2 reveals, however, that all definitions there are local, i.e. they can be applied in a point. We will exploit this fact by choosing a coordinate system which yields in one point $x_0$ of space-time an especially simple form of the metric and its derivatives, namely *locally geodesic coordinates*. It is well-known, that in these coordinates the Christoffel symbols vanish at $x_0$ and $g_{\mu\nu}(x_0) = \eta_{\mu\nu}$, the constant metric of Minkowski space.

In locally geodesic coordinates, the Einstein Equations (6.109) take in $x_0$ the form [145]

$$R_{\mu\nu}(x_0) = \frac{1}{2}\eta^{\rho\sigma}\left[\partial_{\mu\rho}g_{\nu\sigma}(x_0) + \partial_{\nu\sigma}g_{\mu\rho}(x_0) - \partial_{\mu\nu}g_{\sigma\rho}(x_0) - \partial_{\sigma\rho}g_{\mu\nu}(x_0)\right] = 0\,. \tag{6.110}$$

Furthermore the first prolongation is also very simple, as the metricity of the Levi-Civita connection $D_\rho g_{\mu\nu} = 0$ implies that $\partial_\rho g_{\mu\nu}(x_0) = 0$.

$$\partial_\tau R_{\mu\nu}(x_0) = \frac{1}{2}\eta^{\rho\sigma}\left[\partial_{\tau\mu\rho}g_{\nu\sigma}(x_0) + \partial_{\tau\nu\sigma}g_{\mu\rho}(x_0) - \partial_{\tau\mu\nu}g_{\sigma\rho}(x_0) - \partial_{\tau\sigma\rho}g_{\mu\nu}(x_0)\right] = 0\,. \tag{6.111}$$

Note, however, that the higher prolongations become more complicated, as then further derivatives of $g_{\mu\nu}$ not vanishing in $x_0$ must be taken into account.

We will proceed as for the Yang-Mills Equations using the tableau to prove involution of the symbol and $\delta$-regularity of the coordinate system. Because of the symmetry one can assume $\mu \geq \nu$. We distinguish four cases in the choice of the principal derivatives.

(i) $\mu \neq n$ and $\nu \neq n$: The principal derivative is $\partial_{nn}g_{\mu\nu}$.

(ii) $\mu = n$ and $\nu < n-1$: The principal derivative is $\partial_{n,n-1}g_{n-1,\nu}$.

(iii) $\mu = n$ and $\nu = n-1$: The principal derivative is $\partial_{n,n-1}g_{11}$.

(iv) $\mu = n$ and $\nu = n$: The principal derivative is $\partial_{n-1,n-1}g_{11}$.

Besides the last case this can be seen at once. For (iv) several substitutions of other equations are necessary. This yields $\beta^{(n)}_{Ein} = n(n-1)/2$, $\beta^{(n-1)}_{Ein} = n$. The other $\beta^{(k)}_{Ein}$ vanish[3].

The matrix of the tableau of (6.109) is given by

$$\begin{aligned}\mathcal{T}^{\mu\nu}_{\alpha\beta}[\chi] &= 4\frac{\partial R_{\alpha\beta}}{\partial(\partial_{\kappa\lambda}g_{\mu\nu})}\chi_\kappa\chi_\lambda \\ &= \delta^\mu_\beta\chi^\nu\chi_\alpha + \delta^\mu_\alpha\chi^\nu\chi_\beta + \delta^\nu_\beta\chi^\mu\chi_\alpha + \delta^\nu_\alpha\chi^\mu\chi_\beta - 2\eta^{\mu\nu}\chi_\alpha\chi_\beta - \chi^2(\delta^\mu_\alpha\delta^\nu_\beta + \delta^\nu_\alpha\delta^\mu_\beta)\,.\end{aligned} \tag{6.112}$$

[3] If one uses the full Einstein tensor $G_{\mu\nu}$ instead of the Ricci tensor, one obtains the same values for the $\beta^{(k)}_{Ein}$. However, in (iv) another principal derivative must be chosen. A cosmological term does not influence the symbol.

Recalling the symmetries of $R_{\mu\nu}$ and $g_{\mu\nu}$ it is a square matrix with $n(n+1)/2$ rows and columns. $\alpha, \beta$ label the rows, $\mu, \nu$ the columns. The matrix has not full rank, as for any values of $\alpha, \mu, \nu$ the identity

$$\mathcal{T}^{\mu\nu}_{\alpha\beta}[\chi]\chi^{\beta} - \frac{1}{2}\mathcal{T}^{\mu\nu}_{\gamma\beta}[\chi]\eta^{\gamma\beta}\chi_{\alpha} = 0 \tag{6.113}$$

holds.

Thus the maximal rank of $\mathcal{T}[\chi]$ cannot exceed $n(n-1)/2$. Comparing with the above obtained value for $\beta^{(n)}_{Ein}$ shows that locally geodesic coordinates are indeed $\delta$-regular and hence that we have got the correct values. Furthermore, we can show as in the case of the Yang-Mills Equations that the symbol is involutive. The Cartan characters are consequently

$$\alpha^{(n)}_{Ein} = n, \quad \alpha^{(n-1)}_{Ein} = n^2, \quad \alpha^{(n-k)}_{Ein} = \frac{n(n+1)(k+1)}{2}. \tag{6.114}$$

Since $x_0$ was an arbitrary point, these results hold on the entire space-time.

At the level of the differential equations we recover (6.113) in the so-called *contracted Bianchi identities*

$$\partial_\nu(R^\nu_\mu) - \partial_\mu(R^\nu_\nu) = \Gamma^\rho_{\nu\mu}R^\nu_\rho - \Gamma^\nu_{\nu\rho}R^\rho_\mu\,, \tag{6.115}$$

where $\Gamma$ denotes the Christoffel symbols. Here we have fully written out the covariant derivative to see that indeed the lower order terms are algebraically dependent of the field equations. Thus no integrability conditions arise and the Einstein equations form an involutive system. Like the Yang-Mills Equations they are under-determined, as every one-form $\chi$ is characteristic.

To prove the involution of the Einstein Equations it is not necessary to use the identities (6.115). Instead, we can even give a simple indirect proof of them, for it suffices to show that the integrability conditions vanish in locally geodesic coordinates. But this is obvious, as the equations in (6.111) do not contain lower order derivatives. Thus the number of independent equations in (6.111) is given by the rank of the prolonged symbol and Theorem 2.7 proves that there arise no integrability conditions. Note that it suffices to calculate the number of independent equations in a point, because the Einstein Equations describe a manifold in the jet bundle, i.e. the dimension is the same at every point.

Again the under-determinacy stems from a gauge invariance. (6.109) is invariant under arbitrary coordinate changes, i.e. under gauge transformations[4]

$$g_{\mu\nu} = \frac{\partial\bar{x}^\rho}{\partial x^\mu}\frac{\partial\bar{x}^\sigma}{\partial x^\nu}\bar{g}_{\rho\sigma}\,, \tag{6.116}$$

with $\gamma_1 = n$ gauge functions $\bar{x}^\mu$. This yields a gauge corrected strength

$$\bar{Z}^{(1)}_{Ein} = n(n-1)(n-3)\,. \tag{6.117}$$

Analogously to our previous analysis of the Lorentz gauge, we can consider *harmonic coordinates*, i.e. we pose the gauge condition

$$\Box x^\mu = \frac{1}{\sqrt{-g}}\partial_\nu\left(\sqrt{-g}g^{\mu\nu}\right) = 0\,. \tag{6.118}$$

[4] Note that "gauge transformations" have been defined in Section 6.1 purely mathematically. Physically we are dealing here with trivial coordinate transformations. To write gravity as a gauge theory requires the introduction of a frame bundle with $SO(n-1,1)$ as structure group.

In locally geodesic coordinates for $x_0$ this is equivalent to $\partial_\nu g_{\mu\nu}(x_0) = 0$. After prolonging to second order, we get thus $n^2$ additional equations to (6.110). One sees easily that always $n$ of them are in the same class. Hence, the Cartan characters of the combined system are

$$\alpha_H^{(n)} = 0, \quad \alpha_H^{(n-1)} = n(n-1), \quad \alpha_H^{(n-k)} = \frac{n^2(k+1) + n(k-1)}{2}. \tag{6.119}$$

Thus (6.118) provides no complete gauge fixing, because already $\alpha_H^{(n-1)}$ is too big, as we obtain for the highest gauge corrected Cartan characters

$$\bar{\alpha}_{Ein}^{(n)} = 0, \quad \bar{\alpha}_{Ein}^{(n-1)} = n(n-3). \tag{6.120}$$

Specializing to $n = 4$ we obtain for the Hilbert polynomial

$$H_{Ein}(r) = 90 + \frac{184}{3}r + 12r^2 + \frac{2}{3}r^3. \tag{6.121}$$

Gauge correction yields

$$\bar{H}_{Ein}(r) = 10 + 12r + 2r^2. \tag{6.122}$$

This is almost the same result as for the Maxwell Equations in four dimensions. Only the constant term is larger in (6.93). Correspondingly all the gauge corrected Cartan characters are identical with the exception of $\bar{\alpha}_{Ein}^{(1)} = 0$ instead of 6. This implies especially that both equations have the same strength. The general expressions for the strength, (6.91) and (6.117), respectively, show that this is only the case in four dimensions. This perhaps surprising result has lead to some number mysticism [115] that it "explains" why we live in a four-dimensional world. Taking the complete set of Cartan characters into account the argument looses considerably. Note that we have got here an example of an equation where gauge correction leads to a set of Cartan characters which do not satisfy Proposition 3.5. Thus it is not possible to fix the gauge completely using differential equations.

# Chapter 7

# Miscellaneous Applications

*We will analyze various examples of partial differential equations arising in mathematical or physical applications from a formal point of view. These include complete and Jacobian systems, topologically massive models in three dimensions and two-dimensional dilaton-gravity. Finally we will study an example stemming from variational calculus.*

## 7.1 Complete and Jacobian Systems

As a simple application of the formal theory, we consider complete and Jacobian systems. Besides being an illustrative example this class is quite important, as for instance the construction of invariants leads to complete systems (cf. Section 5.2). Furthermore, as we will see later in this section, the integration of a system of finite type can always be reduced to the integration of a complete system. But any complete system can be integrated by solving only ordinary differential equations. The classical treatment of this class of equations stems mainly from Jacobi and Clebsch [31]. Our presentation follows [22, 97]. We will then show how it can be developed from the point of view of formal theory.

We confine ourselves to linear homogeneous first-order systems with $m = 1$ dependent variables. These equations can be written using vector fields on the base manifold $X$:

$$V_\tau u = \sum_{i=1}^{n} a^i_\tau(x) \frac{\partial u}{\partial x^i} = 0\,, \qquad \tau = 1, \ldots, p\,. \tag{7.1}$$

We assume that the $V_\tau$ are linearly independent. System (7.1) is called *complete*, if they form at each point of $X$ a Lie algebra

$$[V_\rho, V_\sigma] = \sum_{\tau=1}^{p} C^\tau_{\rho\sigma}(x) V_\tau\,. \tag{7.2}$$

If a commutator is not a linear combination of the vector fields $V_\tau$, we add it to the system. Then we look again whether the resulting system is complete. After a few iteration this process will lead to a complete system, as there exist only $n$ linear independent vector fields on $X$. Thus we can assume without loss of generality that (7.1) is complete.

Renumbering the independent variables we can always achieve that it is possible to solve the first equation in (7.1) for $\partial u/\partial x^n$, the second one for $\partial u/\partial x^{n-1}$ etc. This leads to an equivalent system generated by vector fields of the form

$$W_\tau = \frac{\partial}{\partial x^{n-\tau+1}} + \sum_{i=1}^{n-p} b^i_\tau(x)\frac{\partial}{\partial x^i}\,. \tag{7.3}$$

These fields obviously satisfy the trivial commutation relations

$$[W_\rho, W_\sigma] = 0\,. \tag{7.4}$$

Systems built with commuting vector fields are called *Jacobian systems.*

**Proposition 7.1** *A complete system can be integrated by solving systems of ordinary differential equations. If it is generated by $p$ independent vector fields on an $n$-dimensional manifold $X$, then it possesses $n-p$ functionally independent solutions $\psi_k(x)$. Its general solution is given by $\Psi(\psi_k)$ with an arbitrary function $\Psi$.*

PROOF. If our system consists of only one equation

$$Vu = \sum_{i=1}^{n} b^i(x)\frac{\partial u}{\partial x^i} = 0\,, \tag{7.5}$$

it is well-known that its general solution can be obtained by solving its *characteristic system* $dx^i/dt = b^i$ and that it is of the form stated in the proposition.

If we have more than one equation, we have seen above that we can always assume without loss of generality that the system is Jacobian. We start with $V_1$ and compute its general solution. Let it be spanned by the $n-1$ functions $\psi_1^j(x)$ for $j = 1,\dots,n-1$. Then we transform the independent coordinates according to

$$\bar{x}^n = x^n, \quad \bar{x}^j = \psi_1^j(x)\,. \tag{7.6}$$

(Due to the special form (7.3) of the vector fields, this provides indeed an invertible change of coordinates). A short calculation shows that $\bar{V}_1 = \partial/\partial\bar{x}^n$. Since commutators remain invariant under coordinate transformations, the $\bar{V}_\tau$ still commute with $\bar{V}_1$. But this implies that their coefficients are independent of $\bar{x}^n = x^n$. Thus there remains a complete system with one equation less in the independent variables $x^1,\dots,x^{n-1}$.

Repeating this process $p$ times leads to the trivial Jacobian system generated by the vector fields $\bar{V}_\tau = \partial/\partial\bar{x}^{n-\tau+1}$. Obviously its general solution is given by $\Psi(\bar{x}^1,\dots,\bar{x}^{n-p})$. Inverting the performed coordinate changes yields the claim. •

For concrete calculations this proof is not very useful. One can achieve the same with much less ordinary differential equations. One defines the characteristic system of the Jacobian system generated by (7.3) as the following system of $n-p$ total differential equations

$$\omega^k = dx^k - \sum_{\tau=n-p+1}^{n} b^k_\tau dx^\tau = 0\,, \quad k = 1,\dots,n-p\,. \tag{7.7}$$

In the language of Appendix A this represents a Pfaffian system and is nothing else then the annihilator of the Lie algebra spanned by the vector fields $W_\tau$. Note that the special solved form of (7.7) is obtained only for Jacobian systems.

With a trick due to Mayer [107] it can be transformed into a system of ordinary differential equations. One chooses a point $x_0 \in X$ and introduces $p$ parameter $\lambda^\tau$ and a new variable $t$ by writing $x^\tau = x_0^\tau + t\lambda^\tau$ for $\tau = n-p+1,\dots,n$. Since now $dx^\tau = \lambda^\tau dt$, we can rewrite the Pfaffian system (7.7) as a system of ordinary differential equations

$$\frac{dx^i}{dt} = \sum_{\tau=n-p+1}^{n} \lambda^\tau b_\tau^i(x^k, x_0^\sigma + t\lambda^\sigma)\,, \quad \begin{cases} i,k = 1,\dots,n-p\,, \\ \sigma = n-p+1,\dots,n\,. \end{cases} \tag{7.8}$$

The solution of this system will be of the form $x^i = \varphi^i(t, \lambda^\tau, x_0^\tau)$. To obtain a solution of the original system of partial differential equations substituting $\lambda^\tau$ by $(x^\tau - x_0^\tau)/t$ must yield an expression independent of $t$. A rather lengthy computation proves that this happens, if and only if the original system is complete.

From the point of view of formal theory, we see that the transformation of a complete system into a Jacobian system consists just of the operations needed to put the symbol in solved form. (7.3) yields at once $\beta_{Jac}^{(n)} = \beta_{Jac}^{(n-1)} = \cdots = \beta_{Jac}^{(n-p+1)} = 1$ and all the lower $\beta_{Jac}^{(k)}$ vanish. To check involution of the symbol we must study what happens if we prolong an equation with respect to a non-multiplicative variable (Proposition 2.9). For the equation $W_\rho u = 0$, the variables $x^n, \dots, x^{n-\rho+2}$ are non-multiplicative; let $x^{n-\sigma+1}$ be one of them. Prolongation with respect to it yields an equation with the leading derivative $p_{n-\sigma+1,n-\rho+1}$. The only way to cancel it is to subtract the equation $D_{n-\rho+1}W_\sigma u = 0$. Similarly one can see that it is possible to eliminate all second-order derivatives and that the left hand side of the resulting equation is just $[W_\rho, W_\sigma]u$. Hence prolongations with respect to non-multiplicative variables do not contribute to the prolonged symbol and according to our criterion (2.19) we have an involutive symbol.

In order to have no integrability conditions we must require that the above obtained commutators are all linear combinations of the equations $W_\tau u = 0$, but this means nothing else than that the vector fields $W_\tau$ form a Lie algebra. Thus any complete system is involutive. This also explains why in differential geometry a system of vector fields is said to be in involution, if they form a Lie algebra at every point.

We obtain for the Cartan characters

$$\alpha_{Jac}^{(n)} = \cdots = \alpha_{Jac}^{(n-p+1)} = 0\,, \quad \alpha_{Jac}^{(n-p)} = \cdots = \alpha_{Jac}^{(1)} = 1\,. \tag{7.9}$$

According to Corollary 3.10 an algebraic representation of the general solution must contain one arbitrary function of $n-p$ arguments in perfect agreement with Proposition 7.1. Note, however, that this corollary does not prove the existence of a solution in contrast to our analysis here!

Applying Definition 2.24 of a characteristic one-form we find that the forms contained in the characteristic system (7.7) build a basis of the characteristic variety $C$ of the Jacobian system generated by (7.3). Indeed the tableau of this system is given by

$$\mathcal{T}_\tau[\chi] = \chi_{n-\tau+1} + \sum_{i=1}^{n-p} b_\tau^i \chi_i\,. \tag{7.10}$$

Its solution space is obviously spanned by the forms $\omega_k$ in (7.7).

One can partially extend these results to non-linear first-order systems using the so-called Jacobi brackets. If $F, G$ are arbitrary functions of $x^i, u, p_i$, then their Jacobi bracket is defined by

$$[F,G] = \sum_{i=1}^{n} \Big[F_{p_i}(G_{x^i} + p_i G_u) - G_{p_i}(F_{x^i} + p_i F_u)\Big]\,. \tag{7.11}$$

It has similar properties as the commutator of vector fields but obeys a different Jacobi identity. One easily sees that in terms of the formal derivative

$$[F,G] = \sum_{i=1}^{n} \Big( F_{p_i} D_i G - G_{p_i} D_i F \Big) . \tag{7.12}$$

A non-linear system $F_\tau(x^i, u, p_i)$ is involutive, if and only if

$$[F_\rho, F_\sigma] = 0 . \tag{7.13}$$

This can be seen in the same way as before in the linear case. We note further that we get exactly the same Cartan characters and hence the same arbitrariness of the general solution as before.

To conclude this section we show how the integration of a finite type equation, i.e. of an equation with a finite-dimensional solution space, can be reduced to the integration of a complete system. Finite type equations have among others two important applications: They can be used to describe foliations of a manifold and to define Bäcklund transformations.

**Proposition 7.2** *Any involutive equation of finite type can be transformed into an equivalent complete system. Thus it can be integrated by solving ordinary differential equations.*

PROOF. By Appendix C.2 any equation can be rewritten as a first-order equation. Thus we can assume that our finite type equation is of order one and has the following form

$$\mathcal{R}_1 : \Big\{ p_i^\alpha = \varphi_i^\alpha(x^j, u^\beta) , \quad \alpha = 1, \ldots, m, \ i = 1, \ldots, n . \tag{7.14}$$

In general, a finite type system can of course also contain lower order equations. Here these would, however, be algebraic equations, which could be solved for some $u^\alpha$ and thus be used to eliminate these leaving a system of the form (7.14).

Let us now consider the following vector fields

$$X_i = \frac{\partial}{\partial x^i} + \sum_{\alpha=1}^{m} \varphi_i^\alpha \frac{\partial}{\partial u^\alpha} . \tag{7.15}$$

One readily calculates

$$[X_i, X_j] = \sum_{\alpha=1}^{m} (D_i \varphi_j^\alpha - D_j \varphi_i^\alpha) \frac{\partial}{\partial u^\alpha} . \tag{7.16}$$

By assumption $\mathcal{R}_1$ is involutive. This implies that the terms in the parentheses must vanish on $\mathcal{R}_1$, as they represent integrability conditions. Hence the vector fields $X_i$ commute.

We claim now that any solution of the Jacobian system $X_i f = 0$ yields a solution of $\mathcal{R}_1$ and vice versa. $X_i f = 0$ represents a system of $n$ equations with the $n+m$ independent variables $x^i$ and $u^\alpha$. According to Proposition 7.1 it has $m$ independent solutions $F^\beta(x^i, u^\alpha)$. Introducing $m$ arbitrary constants $c^\beta$ we solve the equations $F^\beta(x,u) = c^\beta$ for the variables $u^\alpha$ obtaining $u^\alpha = \psi^\alpha(x^i, c^\beta)$. One can see from the proof of Proposition 7.1 that it is indeed always possible to solve for the $u^\alpha$ due to the special form of the vector fields $X_i$. Since

$$\frac{\partial F^\beta}{\partial x^i} + \sum_{\alpha=1}^{m} \frac{\partial F^\beta}{\partial u^\alpha} \frac{\partial \psi^\alpha}{\partial x^i} = 0 , \tag{7.17}$$

and we can conclude that $u^\alpha - \psi^\alpha(x^i, c^\beta)$ is also a solution of the Jacobian system. But this means nothing else then

$$\frac{\partial \psi^\alpha}{\partial x^i} = \varphi_i^\alpha(x^j, \psi^\beta)\,, \tag{7.18}$$

i.e. $\psi^\alpha$ is a solution of $\mathcal{R}_1$.

Conversely, if $u^\alpha = \psi^\alpha(x^i)$ is a solution of $\mathcal{R}_1$, then

$$X_i\Big(u^\alpha - \psi^\alpha(x^j)\Big) = \varphi_i^\alpha(x^j, \psi^\beta) - \frac{\partial \psi^\alpha}{\partial x^i} = 0\,. \tag{7.19}$$

Thus any solution of $\mathcal{R}_1$ yields a solution of $X_i f = 0$. •

This proof stems from Lie [97]. He formulated it directly for systems of arbitrary order and for systems containing lower order equations. This, however, unnecessarily complicates the notation. The basic idea of the proof is the observation that the vector fields $X_i$ generate symmetries of the equation $\mathcal{R}_1$ (cf. Chapter 5). The solutions $F^\beta$ of the Jacobian system are differential invariants of this symmetry group. In the theory of Bäcklund transformations the $X_i$ are used to obtain zero-curvature representations of partial differential equations [16].

## 7.2 Topologically Massive Models in 3 Dimensions

Deser *et al.* [40, 41] proposed the addition of a topologically motivated, gauge invariant term, namely a Chern-Simons form, to the actions of Yang-Mills Theories and general relativity, respectively, in three dimensions (these terms exist only in odd dimensions). This leads to theories with quite distinct properties from their four-dimensional analogues. Another motivation to study such three-dimensional models is the fact that they appear in the high-temperature limit of four-dimensional ones.

We start with Yang-Mills Theory. Compared with Section 6.6 we simplify the notation by suppressing the internal indices corresponding to the gauge group by using a matrix notation. Deser *et al.* studied the action

$$S = \frac{1}{2\kappa^2}\int dx^3\,\mathrm{tr}\,(F_{\mu\nu}F^{\mu\nu}) - \frac{m}{2\kappa^2}\int dx^3\,\epsilon^{\mu\nu\rho}\,\mathrm{tr}\,(F_{\mu\nu}A_\rho - \frac{2}{3}A_\mu A_\nu A_\rho) \tag{7.20}$$

with coupling constants $\kappa, m$. Variation leads to the field equations

$$\mathcal{D}_\mu F^{\mu\nu} + \frac{m}{2}\star F^\nu = \mathcal{D}_\mu F^{\mu\nu} + \frac{m}{2}\epsilon^{\nu\rho\sigma}F_{\rho\sigma} = 0 \tag{7.21}$$

where $\mathcal{D}_\mu$ denotes again the covariant derivative. Using the dual equations and the well-known relation $[\mathcal{D}_\mu, \mathcal{D}_\nu] = F_{\mu\nu}$ one can derive the following equation for the dual field strength

$$(D_\nu D^\nu + m^2)\star F_\mu = \epsilon_{\mu\nu\rho}[\star F^\nu, \star F^\rho]\,, \tag{7.22}$$

which in linear approximation becomes a Klein-Gordon Equation with "mass" $m$.

Comparing (7.21) with the usual Yang-Mills Equations we notice that the additional terms are of lower order. This implies that our analysis of the symbol from Section 6.6 remains still valid and especially that we get the same Cartan characters. Thus to prove involution there remains only to show that there still holds a Noether identity. It is, however, easy to show that due to the definition of $F_{\mu\nu}$

$$\mathcal{D}_\mu \star F^\mu = 0\,. \tag{7.23}$$

Thus from the point of view of a formal analysis there are no changes in the properties of the field equations through the additional terms. Physically, the theories differ nevertheless considerably [41].

General relativity in three dimensions is special in several respects. The Einstein Equations are dynamically trivial: Every Ricci-flat space-time is flat, as Ricci and curvature tensor have the same number of independent components and are equivalent. In the formal analysis one can show that the gauge corrected Cartan characters vanish for $n = 3$. Furthermore the Weyl tensor vanishes. To distinguish conformally flat space-times one must introduce another tensor [48], often called Cotton tensor.

The action proposed by Deser *et al.* is similar to the one used in the Yang-Mills case. One must replace $F$ by the curvature tensor and $A$ by the Levi-Civita connection. This leads to the field equations

$$G^{\mu\nu} + \frac{1}{m\sqrt{g}} C^{\mu\nu} = 0 \tag{7.24}$$

where

$$G^{\mu\nu} = R^{\mu\nu} - \frac{1}{2} g^{\mu\nu} R \tag{7.25}$$

denotes the Einstein tensor constructed from the Ricci tensor $R^{\mu\nu}$ and the curvature scalar $R = R^{\mu}_{\mu}$ and

$$C^{\mu\nu} = \epsilon^{\mu\rho\sigma} \mathcal{D}_{\rho} (R^{\nu}_{\sigma} - \frac{1}{4} \delta^{\nu}_{\sigma} R) \tag{7.26}$$

is the above mentioned Cotton tensor. We note the fundamental difference to the Yang-Mills case, namely that the Cotton tensor depends on third-order derivatives of the metric, thus it is of higher order than the Einstein tensor.

Alternatively one could ask what is the most general tensor constructible of the metric and its derivatives up to third order that has similar properties as the Einstein tensor, i.e. it is (i) a symmetric second-rank tensor, (ii) divergence-free and (iii) trace-less. Lovelock [100] proved that in three dimensions there exists only one tensor satisfying all these conditions and that is the Cotton tensor. This was the motivation for Dereli and Tucker [39] to study (7.24) (with an additional cosmological term). They constructed wave solutions and showed that there exist no static rotationally symmetric solutions.

Opposed to the Yang-Mills case we can no longer use the results of our previous formal analysis of the Einstein Equations, for the symbol is now determined by the Cotton tensor. In a locally geodesic coordinate system it is given by

$$\begin{aligned} C^{\mu\nu} = \tfrac{1}{2}\epsilon^{\mu\nu\sigma}\eta^{\nu\lambda}\Big[\ & \eta^{o\tau}(\partial_{\sigma o\rho} g_{\tau\lambda} - \partial_{\tau o\rho} g_{\sigma\lambda} - \partial_{\sigma\lambda\rho} g_{\tau o} + \partial_{\tau\lambda\rho} g_{\sigma o}) \\ & - \tfrac{1}{4}\delta_{\sigma\lambda}\eta^{o\pi}\eta^{\tau\omega}(\partial_{o\tau\rho} g_{\omega\pi} - \partial_{\omega\tau\rho} g_{o\pi} - \partial_{o\pi\rho} g_{\omega\tau} + \partial_{\omega\pi\rho} g_{o\tau})\Big]\,. \end{aligned} \tag{7.27}$$

Since we have only six equations in six dependent variables, it is easier to write explicitly down the matrix of the tableau instead of performing complicated index manipulations. If we label the columns by $g_{33}, g_{32}, g_{22}, g_{31}, g_{21}$ and $g_{11}$ and drop some integer factors, it has the following form

$$\mathcal{T}[\chi] = \begin{pmatrix} \chi_1\chi_2\chi_3 & \chi_1(\chi_2^2-\chi_3^2) & -\chi_1\chi_2\chi_3 & -\chi_2(\chi_2^2+\chi_3^2) & \chi_3(\chi_2^2+\chi_3^2) & 0 \\ -\chi_3(3\chi_2^2+\chi_1^2) & 2\chi_2(\chi_1^2-3\chi_3^2) & 3\chi_3(\chi_1^2-\chi_3^2) & 2\chi_1(\chi_2^2-\chi_3^2) & -4\chi_1\chi_2\chi_3 & \chi_3(\chi_2^2+\chi_3^2) \\ 3\chi_2(\chi_1^2-\chi_2^2) & 2\chi_3(3\chi_2^2-\chi_1^2) & \chi_2(\chi_1^2-3\chi_3^2) & -4\chi_1\chi_2\chi_3 & 2\chi_1(\chi_3^2-\chi_2^2) & \chi_2(\chi_2^2+\chi_3^2) \\ \chi_1\chi_2\chi_3 & \chi_1(\chi_1^2-\chi_3^2) & 0 & -\chi_2(\chi_1^2+\chi_3^2) & \chi_3(\chi_3^2-\chi_1^2) & \chi_1\chi_2\chi_3 \\ 3\chi_1(\chi_1^2-\chi_2^2) & 4\chi_1\chi_2\chi_3 & \chi_1(\chi_1^2-\chi_3^2) & 2\chi_3(\chi_2^2-3\chi_1^2) & -2\chi_2(\chi_1^2+\chi_3^2) & \chi_1(\chi_2^2+3\chi_3^2) \\ 0 & \chi_1(\chi_1^2-\chi_2^2) & \chi_1\chi_2\chi_3 & \chi_2(\chi_2^2-\chi_1^2) & -\chi_3(\chi_1^2+\chi_2^2) & \chi_1\chi_2\chi_3 \end{pmatrix} . \tag{7.28}$$

Its maximal rank is 3 and can for instance be obtained by choosing $\chi = 2dx^1 + dx^3$. However, the coordinate system is not $\delta$-regular, for all coordinate forms $dx^i$ are systatic (this is not very surprising and basically due to the appearance of the $\epsilon$-tensor in (7.27)). Nevertheless, (7.27) contains only terms of class 2 or 3. Thus without using the 2-tableau we can conclude that $\beta_C^{(3)} = \beta_C^{(2)} = 3$ and $\beta_C^{(1)} = 0$.

Involution of (7.24) comes again from the Bianchi identity. The proof is essentially identical with the one given for the vacuum field equations in Section 6.8. The Bianchi identity ensures that the symbol is involutive and that no integrability conditions arise. Thus for the formal analysis the most important property of the Cotton tensor is that is is divergence-free. The vanishing of the trace implies that without additional fields the curvature scalar vanishes in this modified theory, too.

Although $\beta_C^{(k)} = \beta_{Ein}^{(k)}$, the Cartan characters are different, as we are now dealing with a third-order equation:

$$\alpha_C^{(3)} = 3\,, \quad \alpha_C^{(2)} = 15\,, \quad \alpha_C^{(1)} = 36\,. \tag{7.29}$$

The Hilbert polynomial is given by

$$H_3(r) = 54 + \frac{39}{2}r + \frac{3}{2}r^2\,. \tag{7.30}$$

Of course the theory is still invariant under arbitrary coordinate transformations. Gauge corrections with $\gamma_1 = 3$ gauge functions yield

$$\bar{\alpha}_C^{(3)} = 0\,, \quad \bar{\alpha}_C^{(2)} = 6\,, \quad \bar{\alpha}_C^{(1)} = 18\,. \tag{7.31}$$

The strength of this massive theory is 12, i.e. exactly the value for Einstein and Maxwell Equations, respectively, in four dimensions. Whether this has also some mystical "explanation" is unknown.

## 7.3 Two-dimensional Dilaton-Gravity

The two-dimensional dilaton-gravity has attracted much attention in recent years. There are several reasons for this. The quantization of gravity still represents a major unsolved problem. Two-dimensional models can serve here as an easier accessible test ground than the full four-dimensional theory. One must, however, add a scalar field to obtain reasonable field equations, as in two dimensions the Einstein tensor vanishes and a Ricci flat space has a zero curvature tensor [48].

There are many other reasons to study these models. One obtains for instance a similar type of action in effective string theory [21]. An interesting feature here is the existence of black hole solutions [164]. This explains the interest in the quantum cosmology of dilaton-gravity.

Under fairly general assumptions the most general coordinate invariant lagrangian density containing the metric and a scalar field is given by

$$\mathcal{L} = \sqrt{-g}\left(\frac{1}{2}g^{\mu\nu}\partial_\mu\phi\partial_\nu\phi - V(\phi) + D(\phi)R\right) \tag{7.32}$$

where $R$ denotes as usual the curvature scalar. Using field redefinitions one can, however, considerably simplify this action [5]. Especially it is possible [99] to eliminate the kinetic term for $\phi$ by transforming the metric

$$\bar{g}_{\mu\nu} = \Omega^2(\phi)g_{\mu\nu} \tag{7.33}$$

where $\Omega$ must satisfy the differential equation

$$4D'(\phi)\frac{d\ln\Omega}{d\phi} = 1\,. \tag{7.34}$$

Furthermore we can eliminate $D(\phi)$ by taking it as a new field $\bar{\phi}$ (one must be careful here, if $D$ has critical points [5]). We will therefore restrict our attention to the model described by the action

$$\mathcal{L} = \sqrt{-g}\left[V(\phi) + \phi R\right]. \tag{7.35}$$

Variation with respect to the metric yields the following "Einstein Equations"

$$\mathcal{D}_\mu\mathcal{D}_\nu\phi - g_{\mu\nu}\left[\mathcal{D}^\rho\mathcal{D}_\rho\phi - V(\phi)\right] = 0\,. \tag{7.36}$$

We see at once that if we consider the metric as a kind of external field, this represents a finite type equation for the dilaton field $\phi$. The variation with respect to $\phi$ leads to the additional equation

$$R - V'(\phi) = 0\,. \tag{7.37}$$

The formal analysis of the combined system of field equations shows that it is in involution. An interesting fact hereby is that (7.36) entails more or less (7.37). The integrability conditions of (7.36) require that either (7.37) is satisfied or $\phi$ must be constant. Similar effects are known from other theories coupled to gravity. It would be interesting to study whether this stems from the particular way of coupling or whether it is more accidentally.

The Cartan characters of the full system are

$$\alpha^{(2)}_{DG} = 2\,, \quad \alpha^{(1)}_{DG} = 6\,. \tag{7.38}$$

We must, however, adjust for the invariance under coordinate transformations. Since $\gamma_1 = 2$, the gauge corrected Cartan characters vanish. Thus the reduced phase space of this theory is finite-dimensional. This fact was also proven in [5, 99] using an explicit reduction.

We will now perform a similar reduction to a finite type equation in order to be able to apply the results of Section 7.1. This will allow us to consider ordinary differential equations instead of the field equations. First we exploit the well-known fact that every two-dimensional metric is conformally flat [48] and set

$$g_{\mu\nu} = e^{\lambda}\eta_{\mu\nu}\,. \tag{7.39}$$

With this simpler metric the combined field equations can be written in the following form:

$$\mathcal{R}_2\;:\;\begin{cases} \phi_{tt} - \frac{1}{2}(\phi_t\lambda_t + \phi_x\lambda_x) + \frac{1}{2}e^{\lambda}V(\phi) = 0\,, \\ \phi_{xx} - \frac{1}{2}(\phi_t\lambda_t + \phi_x\lambda_x) - \frac{1}{2}e^{\lambda}V(\phi) = 0\,, \\ \phi_{xt} - \frac{1}{2}(\phi_x\lambda_t + \phi_t\lambda_x) = 0\,, \\ \lambda_{tt} - \lambda_{xx} + e^{\lambda}V'(\phi) = 0\,. \end{cases} \tag{7.40}$$

We will show at the end of this section that the conformal gauge (7.39) can be further enhanced by requiring that $\lambda_x = 0$, i.e. it is always possible to find a coordinate system where $\lambda$ depends only on $t$. It is easy to see that this gauge fixing condition is compatible with (7.40); there do not arise any integrability conditions. If we additionally rewrite (7.40) as a first-order system by introducing new variables, we obtain with obvious notation the following finite type equation

$$\tilde{\mathcal{R}}_1\;:\;\begin{cases} \dot\phi_t = \frac{1}{2}\left[\dot\phi\dot\lambda - e^{\lambda}V(\phi)\right]\,, & \dot\phi_x = \frac{1}{2}\phi'\dot\lambda\,, \\ \phi'_t = \frac{1}{2}\phi'\dot\lambda\,, & \phi'_x = \frac{1}{2}\left[\dot\phi\dot\lambda + e^{\lambda}V(\phi)\right]\,, \\ \phi_t = \dot\phi\,, & \phi_x = \phi'\,, \\ \dot\lambda_t = -e^{\lambda}V'(\phi)\,, & \dot\lambda_x = 0\,, \\ \lambda_t = \dot\lambda\,, & \lambda_x = 0\,. \end{cases} \tag{7.41}$$

Obviously there is no need to introduce $\lambda'$.

Now we construct the two vector fields which define the intermediate Jacobian system. They can easily be read off from (7.41)

$$\begin{aligned} T &= \frac{\partial}{\partial t} + \dot\phi\frac{\partial}{\partial\phi} + \dot\lambda\frac{\partial}{\partial\lambda} + \frac{1}{2}\left[\dot\phi\dot\lambda - e^{\lambda}V(\phi)\right]\frac{\partial}{\partial\dot\phi} + \frac{1}{2}\phi'\dot\lambda\frac{\partial}{\partial\phi'} - e^{\lambda}V'(\phi)\frac{\partial}{\partial\dot\lambda}\,, \\ X &= \frac{\partial}{\partial x} + \phi'\frac{\partial}{\partial\phi} + \frac{1}{2}\phi'\dot\lambda\frac{\partial}{\partial\dot\phi} + \frac{1}{2}\left[\dot\phi\dot\lambda + e^{\lambda}V(\phi)\right]\frac{\partial}{\partial\phi'}\,. \end{aligned} \tag{7.42}$$

We omit the characteristic system and proceed directly to the ordinary differential equations. They are obtained by substituting $t = t_0 + \tau t_1$ and $x = x_0 + \tau x_1$ with fixed but arbitrary constants $t_0, t_1, x_0, x_1$.

$$\begin{aligned}
2\frac{d\dot\phi}{d\tau} &= t_1\left[\dot\phi\dot\lambda - e^\lambda V(\phi)\right] + x_1\phi'\dot\lambda\,, \\
2\frac{d\phi'}{d\tau} &= t_1\phi'\dot\lambda + x_1\left[\dot\phi\dot\lambda + e^\lambda V(\phi)\right]\,, \\
\frac{d\phi}{d\tau} &= t_1\dot\phi + x_1\phi'\,, \\
\frac{d\dot\lambda}{d\tau} &= t_1 e^\lambda V'(\phi)\,, \\
\frac{d\lambda}{d\tau} &= t_1\dot\lambda\,.
\end{aligned} \tag{7.43}$$

(7.43) is a fairly complicated system. It seems hardly feasible to solve it for arbitrary potentials $V(\phi)$. Still it represents some progress compared to the original field equations in so far that we are dealing with ordinary differential equations. This should for instance facilitate quantization, for this is now a problem in quantum mechanics and no longer in quantum field theory. This transformation should also allow for a simpler numerical solution. By choosing specific values for $t_0, x_0$ and adding the equations $dt_1/d\tau = dx_1/d\tau = 0$ one has arrived at a system which can be treated with any of the many standard methods of numerical analysis like Runge-Kutta etc.

Banks and O'Loughlin [5] also presented reductions of the field equations to ordinary differential equations. They are based on the construction of a Killing vector which is orthogonal to the gradient of $\phi$. By using a coordinate system which rectifies this vector field they write a new action in which all fields depend only on the new time variable. Thus the Euler-Lagrange Equations are ordinary differential equations. They call this the homogeneous gauge.

This procedure is somewhat dubious for two reasons. First the Killing vector depends on $\phi$. Thus an explicit construction of the rectifying coordinate system is only possible *after* the field equations (in their original form) have been solved. In other words, it is not possible to recover solutions of the original equations from the solutions of the ordinary differential equations, because the transformation is not known! It depends on $\phi$ given in the *old* coordinates. Furthermore the proof that such a Killing vector always exists needs the original field equations. Thus it is not all clear whether the new and old ones are really equivalent.

In our presentation the ansatz of Banks and O'Loughlin corresponds to the assumption that in the chosen coordinate system $\phi_x = 0$. This implies via the third equation in (7.40) that $\lambda_x = 0$, if $\phi$ is not to be constant. Thus it looks similar to our gauge fixing but is actually a much stronger condition, for (7.41) does not require $\phi_x = 0$. This reflects the above mentioned fact that there is no way back from their ordinary differential equations to the original field equations. Hence they do not obtain the general solution but only a particular one.

By using only the homogeneous gauge and not the conformal one they also get unnecessarily complicated equations. It is straight-forward to show that the metric they use can be brought into conformal form. Now it is possible to obtain a very simple form of the field equations for arbitrary $V(\phi)$, if one uses their "gauge fixing" $\phi_x = 0$.

For this purpose we note that with this condition the first equation of (7.40) becomes $\phi_{tt} - \phi_t\lambda_t = 0$. This can be integrated once and yields

$$\phi_t = Ae^\lambda \tag{7.44}$$

with an integration constant $A$. Substituting this in the second equation of (7.40) leads to

$$A\lambda_t = -V(\phi)\,. \tag{7.45}$$

Differentiating (7.44) allows to eliminate $\lambda$ and we arrive finally at the simple equation

$$A\phi_{tt} + V(\phi)\phi_t = 0\,. \tag{7.46}$$

There is no need to consider the fourth equation of (7.40), as it is an integrability condition and thus automatically satisfied.

To conclude the section we give the proof that it is always possible to assume $\lambda_x = 0$ and that one can indeed construct the necessary coordinate transformation.

**Lemma 7.3** *There exists a coordinate system such that the conformal factor $\lambda$ in (7.39) is independent of $x$.*

PROOF. This proof nicely demonstrates the advantages of the pseudogroup approach using differential equations. We start by analyzing the conditions that a change of coordinates $\bar{x} = f(x,t)$, $\bar{t} = g(x,t)$ respects the conformal form (7.39). It is easy to show that they are

$$f_x f_t = g_x g_t\,, \quad f_x^2 - g_x^2 = g_t^2 - f_t^2\,. \tag{7.47}$$

This represents an involutive system. The condition that the conformal factor is independent of $x$ leads to the additional equation

$$2(f_x f_{xx} - g_x g_{xx})\Lambda + (f_x^2 - g_x^2)\Lambda_x = 0 \tag{7.48}$$

where $\Lambda = e^\lambda$. Hence we must prove that the combined system (7.47,7.48) has a solution for any function $\Lambda$. By a simple analysis of the pivots one can show that the symbol of the system obtained after prolonging (7.47) to second order is involutive, if $f_x g_t \neq f_t g_x$, and that furthermore no integrability conditions arise. But this is exactly the condition for $f, g$ to define an invertible change of coordinates. Thus we can always find a solution $f, g$ and construct from it a coordinate system with the required property. •

## 7.4 A Final Example

In some recent work on variational principles for differential equations Anderson and Pohjanpelto [3] came across the following equation

$$\mathcal{R}_1 : \begin{cases} z^a \dfrac{\partial \lambda}{\partial z^a} + (D+2)\lambda = 0 \\ C^c_{ab} z^b \dfrac{\partial \lambda}{\partial z^c} + C^b_{ab}\lambda = 0\,, \end{cases} \tag{7.49}$$

where all indices run from 1 to $D$, $C^c_{ab}$ are the structure constants of a $D$-dimensional Lie algebra and the unknown $\lambda$ is a function of $z^1, \dots, z^D$. They needed the dimension of the solution space.

Due to the symmetry properties of structure constants the following integrability condition can be derived from the second equation in (7.49)

$$C^c_{ab} z^a z^b \frac{\partial \lambda}{\partial z^c} + C^b_{ab} z^a \lambda = C^b_{ab} z^a \lambda = 0\,. \tag{7.50}$$

Thus if $C^b_{ab} \neq 0$ for some $a$, then there exists only one solution, namely the trivial one $\lambda(z) \equiv 0$. We will assume in the sequel that $C^b_{ab} = 0$ for all $a$. This holds for instance for all semi-simple Lie algebras.

Thus we can take from now on the simpler local representation

$$\mathcal{R}_1 : \begin{cases} z^a \dfrac{\partial \lambda}{\partial z^a} + (D+2)\lambda = 0 \,, \\ C^c_{ab} z^b \dfrac{\partial \lambda}{\partial z^c} = 0 \,. \end{cases} \tag{7.51}$$

It is perhaps worth while noting that this simplification has no real impact on the result. The omitted lower order terms play no role in the analysis of the symbol. They just lead to one additional equation, $\lambda = 0$, which turns a one-dimensional solution space into a zero-dimensional one.

To proceed further we make the assumption that the Lie algebra is semi-simple. The structure theory of semi-simple Lie algebras is well-known [79]. In the Cartan-Weyl basis the structure constants take an especially simple form. Let $r$ be the rank of the algebra. To simplify the notation, we write from now on $i, j, k$ for indices running from 1 to $r$; $l, m, n$ for indices running from $r+1$ to $D$. Indices $a, b, \dots$ run always from 1 to $D$.

We denote by $\omega^{(l)}$ the roots and by $\omega^{(l)}_i$ the covariant components of the root vectors. Since minus a root is always a root, too, we adopt the convention $\omega^{(l)} = -\omega^{(D+r-l+1)}$. Then we get the following structure constants:

$$\begin{gathered} C^k_{ij} = C^l_{ij} = 0 \,, \\ C^j_{il} = 0 \,, \; C^m_{il} = \omega^{(l)}_i \delta^m_l \,, \\ C^i_{mn} = \omega^{(m)i} \delta^{-\omega^{(m)}}_{\omega^{(n)}} \,, \; C^l_{mn} = N_{mn} \delta^{\omega^{(l)}}_{\omega^{(m)}+\omega^{(n)}} \,. \end{gathered} \tag{7.52}$$

Substitution yields the system

$$\mathcal{R}_1 : \begin{cases} \displaystyle\sum_{a=1}^{D} z^a \frac{\partial \lambda}{\partial z^a} + (D+2)\lambda = 0 \,, \\ \displaystyle\sum_{l=r+1}^{D} \omega^{(l)}_i z^l \frac{\partial \lambda}{\partial z^l} = 0 \,, \\ \displaystyle\sum_{i=1}^{D} \left( \omega^{(m)}_i z^i \frac{\partial \lambda}{\partial z^m} + \omega^{(m)i} z^{D+r-m+1} \frac{\partial \lambda}{\partial z^i} \right) + \\ \qquad \displaystyle\sum_{l,n=r+1}^{D} N_{mn} \delta^{\omega^{(l)}}_{\omega^{(m)}+\omega^{(n)}} z^n \frac{\partial \lambda}{\partial z^l} = 0 \,, \end{cases} \tag{7.53}$$

For later use, we note that there exist exactly $r$ linearly independent root vectors. We can assume that they are $\omega^{(r+1)}_i, \dots, \omega^{(2r)}_i$. Since the Killing metric $\kappa_{ab}$ of a semi-simple Lie algebra is non-degenerate, the first $r$ contravariant root vectors are then also linearly independent.

Written as a matrix, the symbol $\mathcal{M}_1$ of $\mathcal{R}_1$ has the following form where the $a$-th column contains the coefficients of $\partial\lambda/\partial z^a$:

$$\mathcal{M}_1 = \left(\begin{array}{ccc|ccc}
z^1 & \cdots & z^r & z^{r+1} & \cdots & z^D \\ \hline
 & & & \omega_1^{(r+1)} z^{r+1} & \cdots & \omega_1^{(D)} z^D \\
 & 0 & & \vdots & & \vdots \\
 & & & \omega_r^{(r+1)} z^{r+1} & \cdots & \omega_r^{(D)} z^D \\ \hline
\omega^{(r+1)1} z^D & \cdots & \omega^{(r+1)r} z^D & -\omega_i^{(r+1)} z^i & & * \\
\vdots & & \vdots & & \ddots & \\
\omega^{(D)1} z^{r+1} & \cdots & \omega^{(D)r} z^d & * & & -\omega_i^{(D)} z^i
\end{array}\right). \tag{7.54}$$

The $*$ stands for the terms $N_{mn}\delta^{\omega^{(l)}}_{\omega^{(m)}+\omega^{(n)}} z^n \frac{\partial\lambda}{\partial z^l}$. Note that no such terms occur on the diagonal. We can write the symbol symbolically as

$$\begin{pmatrix} & z \\ 0 & A \\ B & C \end{pmatrix}. \tag{7.55}$$

$A$ and $B$ have both maximal rank, namely $r$, because, according to our assumption on the root vectors, the first $r$ columns of $A$ and the first $r$ rows of $B$ are linearly independent. As we have already seen during the construction of the integrability condition, we can eliminate the last row of the symbol by adding all equations but the first multiplied with the corresponding variable $z^a$. The remaining rows in $C$ are linearly independent due to the diagonal terms.

We can thus conclude that $\mathcal{M}_1$ has maximal rank, namely $D$, and consequently dimension 0. A vanishing symbol is, however, trivially involutive. Hence, only the question of further integrability conditions remains. A direct proof that none occur seems rather tedious. There exists, however, a much easier way, for Anderson and Pohjanpelto [3] have constructed a solution of $\mathcal{R}_1$.

A system with vanishing symbol has always a finite dimensional solution space, even if it is not involutive, because all prolongations and projections of it will have again vanishing symbols. Thus an upper bound on the dimension of the solution space is given by the dimension of $\mathcal{R}_1$. As we can see from the analysis of the symbol, $D-1$ of the $D$ equations in the upper line of (7.51) are independent. Together with the remaining one this yields $D$ equations in a $(D+1)$-dimensional space. Thus $\dim \mathcal{R}_1 \leq 1$.

On the other hand, one can verify that

$$\lambda = (\kappa_{ab} z^a z^b)^{-\frac{D+2}{2}} \tag{7.56}$$

is a solution. Due to the linearity of the system, all multiples of (7.56) are solutions, too, and we have thus at least a one-dimensional solution space. Together with the result above this implies that the dimension of the solution space is exactly 1 and thus that no integrability conditions can occur. $\mathcal{R}_1$ is consequently involutive.

# Chapter 8

# Computer Algebra

*We will give a short overview of the application of computer algebra to differential equations and a brief introduction into the computer algebra system* AXIOM*. The main part of the chapter will be the description of our* AXIOM *implementation of an environment for symbolic computations in the realm of geometric approaches to differential equations.*

## 8.1 Computer Algebra and Differential Equations

Most casual users of computer algebra systems think that computer algebra and differential equations means basically the design of solution algorithms. But the real situation is fairly different. Although most general purpose systems provide a kind of `solve` command for differential equations, they actually employ mainly well-known techniques and some more or less clever heuristics to choose and apply them. Especially for partial differential equations no solution algorithm treating reasonably general and complicated systems is known, hence most computer algebra systems give up very quickly.

Consequently, computer algebra applications to differential equations tend to work more indirectly. They help to study and understand properties of the solution space. Sometimes they may lead to special solutions. The reason for the use of computer algebra systems lies in the simple fact that most computations with differential equations tend to become soon very tedious. Hence such systems have proved to be an almost indispensable tool. We do not intend to give here a full survey but will briefly indicate the main directions emphasizing differential geometric methods. An excellent survey with a large bibliography oriented more towards algebraic approaches was given by Singer [143].

Almost any constructive method for differential equations has meanwhile been implemented in some computer algebra system. One can, however, distinguish certain approaches which have found most attention. One can classify them coarsely as follows:

- Completion (construction of integrability conditions)
- Symmetry analysis
- Singularity analysis
- Differential ideal theory
- Differential Galois Theory

**Completion.** Any approach that claims to treat general systems of partial differential equations must deal with integrability conditions. The first systematic solution was probably given by the Janet-Riquier Theory (cf. Appendix B) with the introduction of passive systems. Implementations of this ansatz have been undertaken by many authors, see e.g. [124, 135, 155]. The so-called Differential Gröbner Bases [101, 102] can also be seen as an extension of this approach. In geometric theories the notion of a passive system is replaced by involution. Hartley and Tucker [67, 68] implemented the Cartan-Kähler approach using exterior systems (cf. Appendix A). An earlier version of our AXIOM implementation of the formal approach was published in Ref. [132],

**Symmetry analysis.** This approach (cf. Chapter 5) has made the strongest impact on computer algebra applications to differential equations. There exist so many implementations that it is very difficult to keep an overview. We must refer here to a recent survey by Hereman [70] (an updated and extended version will appear in [71]). Most packages just set up the determining system (5.5) for symmetry generating vector fields. A few of the packages also try some heuristics to solve it [69, 134]; usually they can cope, however, only with rather simple examples.

**Singularity analysis.** We must distinguish two approaches: In the Painléve analysis [78] one requires that every solution has at most poles as movable singularities. Movable means here depending on initial data. This method was originally developed for ordinary differential equations and later generalized to partial differential equations [163]. It represents one of the standard tests for (completely) integrable systems. Truncated Painléve expansions are also useful for the construction of Bäcklund transformations, Lax pairs and much more [162]. An implementation for ordinary differential equations was described by Rand and Winternitz [122]. Another direction is given by the construction of formal solutions in the neighborhood of singular points of the equations [46]. It is mainly applied to ordinary differential equations.

**Differential ideal theory.** Here one tries to find a differential extension of algebraic ideal theory. Many of the ideas can already be found in the book of Ritt [128]. The main tools are either Gröbner bases or characteristic sets. Since the ring of differential polynomials is, however, no longer Noetherian, this generalization runs into problems, for algorithms along the lines of the Buchberger algorithm do not terminate in general [24]. As already mentioned above, Mansfield [101, 102] introduced (and implemented) so-called Differential Gröbner Bases, but due to the use of pseudo-reduction they have weaker properties than their algebraic counterpart.

**Differential Galois Theory.** Already Lie was looking for a differential analog of the (algebraic) Galois Theory. What is nowadays called Differential Galois Theory resembles, however, only faintly his ideas. It is mainly based on the work of Kolchin [86] and culminates in the Singer algorithm for computing Liouvillian solutions of linear ordinary differential equations with rational coefficients [142]. An implementation of this algorithm has not been achieved so far. Only the simpler Kovačic algorithm for linear second order ordinary differential equations has been implemented [129]. Pommaret [118] has developed a Differential Galois Theory following Lie's ideas and using the formal theory.

A comparison of the impact made by symmetry analysis and by Differential Galois Theory, respectively, demonstrates clearly the importance of the availability of computer algebra tools. The latter one is a hardly known theory studied only by a few experts. The former one was in the same state for many decades following Lie's original work. One reason was definitely the extremely tedious determination of the symmetry algebra for any only slightly complicated equation. As soon as computer algebra systems came up and were powerful enough, the first packages to set up at least the determining equations were developed. Since then Lie methods belong to the standard tools for treating partial differential equations.

## 8.2 AXIOM

AXIOM [81] was mainly developed at the IBM research center in Yorktown Heights under the name of Scratchpad II. Since 1992 it has been marketed by NAG under the new trade mark AXIOM. We are currently using Version 1.2. It is rather different compared with other general purpose computer algebra systems. Systems like REDUCE, Maple, Mathematica etc. differ not much in their principal structure: they have one basic data type for symbolic expressions and the core of the system consists of the simplifier. The real differences between the systems lay in the simplification strategy, in the power of the advanced packages (e.g. factorization, integration, equation solving) in the system library, in the user interface etc.

AXIOM has a completely different structure. It is inspired by category theory and the theory of abstract data types. Meanwhile there are attempts to incorporate this approach into other systems [60, 62], too. We will describe it briefly in this section in order to give somebody unacquainted with AXIOM a chance to understand some of the following sections. More detailed descriptions of the AXIOM type system and the underlying philosophy can be found in [36, 37, 161].

There is a three level hierarchy: Any *object* is member of one (and only one) *domain* which itself is member of a *category*. The easiest way to understand this hierarchy is to consider a concrete example: The object 5 is a member of the domain `Integer` which is a member of the category `Ring`. In the language of computer science, domains represent *abstract data types*, whereas a category is considered as a *second order type*.

Besides categories and domains there are like in most computer algebra systems *packages*. They do not implement an abstract data type but consist simply of a collection of procedures. Their structure is very similar to a domain, i.e. they first specify the exported operations, then they provide an implementation for them. Unlike a domain, however, a package does not need a representation, as it has no members.

A category specifies which operations are possible in its domains, i.e. it declares their signatures. What it does not contain are any information on the representation of objects. The category `Ring`, for instance, requires that every domain belonging to it must have an operation `+` which takes two ring elements as arguments and returns a ring element. It does not, however, give a concrete implementation of this operation, because it will probably be different in the different domains belonging to `Ring`. The addition of two integers will be provided by the domain `Integer`.

As we will see later a category can also contain implementations of operations. The big difference between a domain and a category is that the former one specifies a *representation* for its objects in terms of other data types. Within the body of the domain constructor this representation type is considered as equivalent to the new data type generated by the domain. This means that if the representation consists e.g. of a record, it is possible to access the individual slots of this record in procedures of the domain. The representation is, however, invisible outside of the domain, i.e. operations from other domains or packages which use objects of this new type cannot see the slots of the record.

A category may be a sub-category of another one. For instance every field is a ring but with additional properties. Consequently, in AXIOM the category `Field` is a refinement of `Ring` by further operations like division, i.e. it *inherits* all specifications of `Ring` and adds new ones. This avoids the tedious duplication of signatures.

Although the system's name is now AXIOM, its weak point is actually the treatment of axioms. A field is not only distinguished from a ring by the fact that it allows division but also by the fact that its multiplication is always commutative. The only way AXIOM can handle such properties is through *attributes*. `Field` has e.g. the attribute `commutative("*")`. But this amounts in practice to not much more than a comment, because the system has no way to check whether or not the implementation of `*` in a domain is commutative. Thus it is more a remainder to the programmer. It is, however, possible to ask whether a given domain has a certain attribute. This is useful, when domains or categories are passed as arguments (see below).

AXIOM's approach to symbolic computation has several advantages. It helps the programmer to ensure the correctness of his computations. Since it is a strongly typed system, every object must have a type, i.e. it belongs to a domain which determines what kind of operations can be performed with it. As long as the semantic of the used domains is sound, only well-defined operations can be called. For instance it is not possible to divide ring elements for the simple reason that no division is defined in the category `Ring`! Of course one can always circumvent this protection mechanism, but it still helps correct programming.

While this point might appear a bit vague and fancy, a real advantage of the object-oriented approach lies in the possibility of generic programming. AXIOM offers two mechanisms for this purpose. The first one is to define procedures "categorically", i.e. one implements a procedure already at the level of a category. Then every domain in the category (and every sub-category) inherits this default implementation; it can, however, still provide its own implementation overriding the default one.

A prototypical example for this technique is the Euclidean algorithm to compute greatest common divisors. AXIOM provides a sub-category of `IntegralDomain` called `EuclideanDomain`. It contains a default implementation for the operation `gcd` using the Euclidean algorithm. Every domain which belongs to `EuclideanDomain` automatically inherits it. Of course this algorithm uses operations which are differently implemented in each of these domains like multiplication. This does not matter, as the category specifies the signature of multiplication and this information suffices for the compiler.

The possibility to pass domains as parameters to another domain (or package or category) yields another way of generic programming. Again AXIOM requires that every parameter is given a type. In the case of a domain this type is the category to which it belongs. This category also specifies all operations possible with objects of the domain. Thus it is only possible to use these operations. This mechanism is for instance used in our implementation of the completion algorithm 2.3 (cf. Section 8.5). The corresponding package takes as argument a domain which represents the class of differential equations considered (e.g. linear equations or polynomial equations). It only knows that this domain lies in the category `JetBundleFunctionCategory`.

Our completion algorithm requires several non-trivial operations like computing the dimension of a submanifold or simplification of a system. The implementation of these varies strongly depending on the kind of equations considered. But this does not matter for the completion procedure which can be written completely independent. Thus one can take full advantage of specific properties of special classes of differential equations and still use the same completion procedure.

A further advantage of AXIOM is the possibility to overload procedure names, i.e. several procedures can have the same name as long as they are distinguished by different signatures. Probably the heaviest overloaded name is `coerce`. This name is always used, if objects are transformed from one representation into another equivalent one. Every domain must contain a `coerce` procedure to the domain `OutputForm`, because this determines the way its objects are displayed in the output.

The implementation of categories, domains or packages requires the use of the compiler. It is still in an experimental stage. The first official release (of a very different new one) will be with the next version of AXIOM. Interactive work with AXIOM is performed using the interpreter. It has sometimes a slightly different syntax. The main difference is, however, that in the interpreter it is no longer necessary to enter an explicit type for every object.

Since this would be very tedious, the interpreter provides a type inference mechanism which tries to derive the types of the objects in the input line. Only if there are ambiguities or the structure of the input is too complicated, the user must declare the types of at least some objects explicitly. The only information this mechanism needs are the signatures (here often called mode maps) of all used operations. (A description of this mechanism can be found in [149]).

Because of the huge size of the library it is not possible to load its complete content at the start of an AXIOM session. If the run time system needs an operation which is not "exposed", as this is called in AXIOM, it is automatically loaded. This autoload scheme does not work with user provided domains and categories. All those that will be used during the calculations must be loaded explicitly using the `)load` command of the interpreter. To facilitate the entering of the often very long type names the AXIOM compiler requires that every category, domain or package is assigned an abbreviation. We will also make use of this and denote everything by its abbreviation as soon as it is introduced.

The remainder of this section contains a short overview of a few domains etc. contained in the AXIOM library which have been used in our implementation. Due to the size of the library one of the biggest difficulties for an AXIOM programmer is to find the ones he needs... Basic data types comprise `Integer` with the subtype `NonNegativeInteger`, `Symbol` and `Boolean`. Rational numbers can be obtained as `Fraction Integer` (the domain constructor `Fraction` takes a ring as argument and returns its quotient field). More complicated data types can be constructed using e.g. `List`, `Vector`, `Matrix` or `Record`.

AXIOM provides a complicated hierarchy of categories for the basic structures used in algebra [38]. We will need only a few of these and most of them have already been mentioned. We will deal almost exclusively with rings, i.e. domains in `Ring`. Sometimes we will have additional structures or properties like a `GcdDomain` or a `PartialDifferentialRing`. The latter one adds partial differentiations with respect to elements of a given set. In our applications this set will always be the domain `Symbol`.

The data type for general expressions is `Expression Integer`. It corresponds roughly to the one basic data type used by most computer algebra systems. It can represent arbitrary expressions like $\sin(x + y) + f(x * y)$ with an unspecified function $f$. All the usual operations like plus, minus, differentiate, etc. can be performed within this domain. Like most other computer algebra systems AXIOM represents an element of it internally as a polynomial. The unknowns are so-called kernels. A kernel can be a simple variable like $x$ but also a more complicated expression like $f(x * y)$. Basically everything that cannot be further simplified is a kernel (of course AXIOM disposes of a domain called `Kernel` for such objects). Kernels like $f(x * y)$ are generated using `BasicOperator`.

Polynomials are probably the most important objects in computer algebra. Hence AXIOM provides many different categories and domains for them. They vary basically in the chosen representation, as for different tasks different ones may be most efficient. `SparseMultivariatePolynomial` uses for instance a recursive representation and `DistributedMultivariatePolynomial` obviously a distributed one.

## 8.3 Implementation I — General Remarks

Most computer algebra applications to differential equations published so far consist of special purpose packages devoted to one specific algorithm. Communication between different packages, e.g. further processing of the output of one algorithm by another one, is then often rather awkward, as every package uses its own data structures and thus complicated conversion procedures must be implemented.

This problem could be avoided by implementing a general purpose environment for geometric computations. Such an environment should comprise basic data structures and procedures for jet bundles and differential equations as they are typically needed. Within this environment it would then be possible to implement packages for different purposes like completion to an involutive system, construction of the symmetry algebra etc.

We have started with the development of such a general environment for computations within the jet bundle formalism. We believe that AXIOM is especially well adapted for such a project because of its unique structure. A first version was implemented by J. Schü in his master's thesis [131]. Together with some minor improvements it was presented in [132]. The current version is mainly distinguished from the earlier one by much more efficient data structures and the introduction of many simplification routines.

In its current state the environment comprises over 6000 lines of AXIOM code organized in 3 categories, 11 domains and 3 packages. There is a natural hierarchy consisting of four layers:

1. Jet bundles, i.e. data structures for local coordinate systems
2. Sections, i.e. functions defined on these charts
3. Higher structures like vector fields, differential forms or a special data structure for differential equations
4. Application packages

The first layer is basically for a better looking user interface. One could argue whether it is really necessary. The only "computations" that happen in it are the construction of the output form for jet variables.

Before we will give a detailed description of the implementation, we present an overview of all the categories and domains provided in the individual layers (see also Fig. 8.1). A list of their abbreviations can be found in Fig. 8.2.

**Layer 1.** The category `JetBundleCategory` specifies jet bundles with a fixed number of independent and dependent variables. The order of the jet bundle is not fixed but passed as argument to the procedures needing it. An instance of this category is given by the domain `JetBundle`. It implements jet bundles with arbitrary names for the independent and dependent variables. The domain `IndexedJetBundle` takes as parameters the numbers of these variables and symbols to construct their names. The domain `JetBundleSymAna` is only needed by the symmetry package (see below).

**Layer 2.** The basic category is `JetBundleFunctionCategory`. There is a subcategory `BaseFunctionCategory` for functions defined only on the base space $X$. Currently four instances are implemented. `JetBundleExpression` is basically an extension of `Expression Integer`. Thus it represents in some sense the most general domain in this category. A similar domain is given by `JetBundleXExpression` which contains only functions of the independent variables. Its main use is to serve as coefficient ring for linear functions realized in `JetBundleLinearFunction`. `JetLazyFunction`

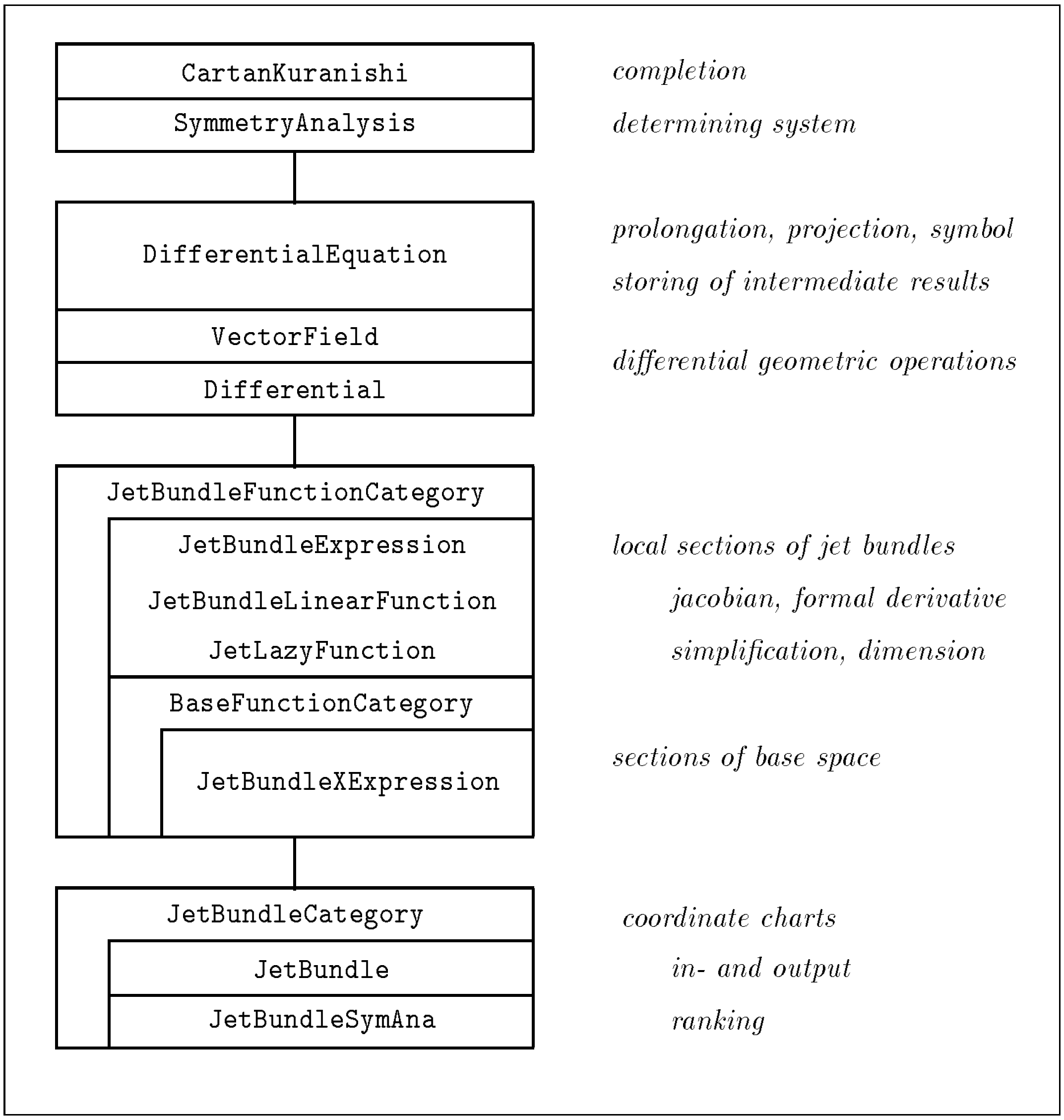


Figure 8.1: Overview of the implemented categories, domains, and packages.

implements a very special instance. It takes another domain of this category as argument and sets a lazy evaluation scheme for partial differentiations on top of it. Its main purpose is to avoid unnecessary calculations when many prolongations are needed.

**Layer 3.** There are two domains `VectorField` and `Differential` for vector fields and one-forms, respectively, implementing the basic differential geometric operations with them. The domain `DifferentialEquation` provides basic operations like prolongation, projection or construction and analysis of symbols and tableaux. Its main purpose is to enhance the efficiency by storing many intermediate results like Jacobians or dimensions.

**Layer 4.** The two application packages implemented so far are `CartanKuranishi` and `SymmetryAnalysis`. The former one contains a completion procedure as described in Section 2.4. Additionally there are some procedures for calculations with Cartan characters, the Hilbert polynomial and for gauge corrections. The symmetry package is still in a very earlier stage. Currently it contains only a primitive procedure to produce determining systems.

Besides these there are two utility packages. `SparseEchelonMatrix` provides a special representation of sparse matrices designed for the fast and efficient computation of row echelon forms. `JetCoordinateTransformation` prolongs coordinate transformations in the base space $\mathcal{E}$ into the jet bundle.

| | |
|---|---|
| `JetBundleCategory` | `JBC` |
| `JetBundle` | `JB` |
| `IndexedJetBundle` | `IJB` |
| `JetBundleSymAna` | `JBSA` |
| `JetBundleFunctionCategory` | `JBFC` |
| `BaseFunctionCategory` | `BFC` |
| `JetBundleExpression` | `JBE` |
| `JetBundleXExpression` | `JBX` |
| `JetBundleLinearFunction` | `JBLF` |
| `JetLazyFunction` | `JLF` |
| `DifferentialEquation` | `DE` |
| `VectorField` | `VF` |
| `Differential` | `DIFF` |
| `CartanKuranishi` | `CK` |
| `SymmetryAnalysis` | `SYMANA` |
| `SparseEchelonMatrix` | `SEM` |
| `JetCoordinateTransformation` | `JCT` |

Figure 8.2: Abbreviations of the new categories, domains, and packages.

Examples of the application of this environment in concrete computations will be given in Sections 8.7 and 8.8. They contain complete examples with in- and output of AXIOM sessions. Tables with most of the exported procedures can be found in Appendix D. They also contain short descriptions of them and can thus be used as a kind of reference manual.

## 8.4 Implementation II — Jet Bundles

This section describes in some detail the implementation of the first two layers. `JetBundleCategory` was mainly introduced to facilitate the passing of jet bundles as arguments to domains and categories of higher layers. Otherwise it would be necessary to repeat each time all the parameters of a jet bundle. Furthermore, it is convenient for some special applications to use another representation than the one provided by the domain `JetBundle`. For instance the notation used in this thesis with indexed variables is of advantage for many large systems; hence there is a domain `IndexedJetBundle`. Most operations are already implemented in `JBC`; `JB` and `IJB` provide essentially only the procedures for the generation and for the output of jet variables.

`IJB` uses as internal representation a record containing the upper and lower index of a jet variable and its type. It knows four different types: Besides the obvious ones `Indep` for the independent variables, `Dep` for the dependent ones and `Deriv` for the derivatives there is a special one called `Const`. It is assigned to the special "jet variable" `1` which is needed for differentiation and for the representation of linear functions (see below).

The upper index labels the different independent and dependent variables; the lower index is needed for derivatives. `IJB` recognizes two different notations for the lower index. Internally always standard multi-index notation is used. For in- and output the user can choose between this and repeated index notation which is the default, as it is usually more convenient for derivatives of low order. The notation can be selected with the procedure `setNotation`.

Jet variables are generated with the procedures `X`, `U` and `P`. For the special cases of only one independent or only one dependent variable, respectively, it is possible to omit the label of the variables. `P` interprets its second argument as a lower index in the currently chosen notation. `Pm` and `Pr` always use multi-index and repeated index notation, respectively.

The implemented ranking is the inverse lexicographic ranking. In addition to `<` and `>` `JBC` provides a procedure `weight` which assigns each jet variable a unique integer such that a higher ranked variable always gets a higher number. Currently there are no provisions for other rankings, although this could be useful for computations as performed in Janet-Riquier Theory. Especially rankings of the type studied by Janet can easily be parameterized by matrices [80]. This mechanism is used e.g. by the `standardform` package of Reid [124].

The `coerce` from `JBC` to `Symbol` consists of the straightforward construction of a symbol with the correct scripts. However, this operation "looses" all information about the jet variable. It is no longer possible to distinguish this symbol from others as a jet variable. Coercion to `Expression Integer (EI)` offers here much more possibilities. The corresponding procedure constructs a `BasicOperator (BOP)` which in turn can easily be coerced to `EI`.

An object of type `BOP` has associated with it a so-called property list which allows to store flags and indicators for properties of the object. We have introduced a new flag `"%jet"` to distinguish jet variables from other kernels. Furthermore we store the weight of each jet variable using the indicator `"%weight"`. This indicator is used by the type `Kernel Expression Integer` to determine an ordering. Thus we can impose our ranking even after coercion to `EI`. This is crucial for the implementation of `JBE` (see below).

`JB` uses exactly the same representation as `IJB`. However, it employs always repeated index notation. `JB` takes two lists of symbols as arguments representing the names of the independent and dependent variables. This is in many applications more natural than indexed variables. For easier input `JB` provides two additional routines with symbols as arguments.

`JBSA` is more or less identical with `JB` but computes its parameters in a special way. It takes as argument another jet bundle. Its independent and dependent variables will be the independent variables of the new jet bundle. The other arguments represent the names used for the dependent variables. The compiler needs such a domain as return type for some procedures in the symmetry package (cf. Section 8.5).

`JetBundleFunctionCategory` provides a default implementation for most operations defined by it. These can be coarsely divided into differential operations like partial or formal derivatives or computation of Jacobians, simplification routines, and some miscellaneous operations like determination of the leading derivative of a function or its effectively occurring jet variables. Furthermore there are the algebraic operations inherited from the category `PartialDifferentialRing Symbol` and some operations are lifted from `JBC` to allow for easier input.

Important among the miscellaneous operations are `jetVariables`, `solveFor`, `subst`, `order`, `leadingDer`, and `dSubst`, as these are much used by the default implementations of the other operations. For the first three of these no default implementation is provided. `jetVariables` returns a list of all effectively in a function occurring jet variables. `solveFor` tries to solve a function for a given jet variable. If that is not possible, it returns `"failed"`. `subst` substitutes a given expression for a jet variable in another given expression.

`order` yields the order as differential equation of a function; it is computed as the order of the leading derivative obtained from `leadingDer`. The default implementation of the latter one in turn calls `jetVariables` and determines the maximum of the resulting list. Whereas `subst` operates purely algebraic, `dSubst` also substitutes for derivatives of the given jet variable. This is done by repeated calls of `subst`.

For partial derivatives one must distinguish between differentiations with respect to objects of the type `Symbol` and differentiations with respect to jet variables, i.e. objects of a domain in `JBC`. The former one is inherited from `PartialDifferentialRing Symbol`. It is of no importance for the formal analysis, as such symbols are considered as parameters. Only differentiations with respect to jet variables are used in formal derivatives and thus in prolongations and similar operations.

Jacobi matrices play an important role in the higher structures discussed in the next section. To compute the formal derivative of a function according to (2.3) requires the knowledge of all its partial derivatives. There are two operations `jacobiMatrix` implemented. In one a second argument containing a list of the effectively occurring jet variables is required. This avoids a call of the procedure `jetVariables`, which can be quite expensive in some instances of `JBFC`.

Similarly, there are several different operations called `formalDiff`. Some are for a single function, some for a system; some take the Jacobi matrix as an additional argument, some compute it. One of them returns a list of the jet variables occurring in the prolonged function to facilitate the determination of its Jacobi matrix. Such "tricks" are fairly important to obtain reasonable run times.

There are two kinds of simplification routines: `simplify`, `simpMod` and `simpOne` use only algebraic operations; `reduceMod` and `autoReduce` apply also differentiations. They serve different purposes. The former ones are used to simplify local representations of differential equations. Hence they can produce an equation of lower order only, if integrability conditions occur. Indeed this is the main task of `simplify`. The latter ones are used to obtain simple generating sets for differential equations. They eliminate for example equations which are just prolongations of others. `JBFC` contains default implementations for all four mentioned simplification routines.

`simplify` has two main tasks. Besides the already mentioned exhibition of integrability conditions it should eliminate functionally dependent equations. This is important for the computation of dimensions (see below). `simpMod` is similar to `reduceMod` (see below) and simplifies a system with respect to another one. `simpOne` tries to simplify individual equations. This entails removal of unnecessary coefficients and signs or reduction of exponents in the case of non-linear equations.

The default implementation of `simplify` uses basically the leading derivatives of the equations. Obviously two equations with different leading derivatives are functionally independent. If several equations with the same leading derivatives occur, `simplify` checks first whether one of them can be solved for it using `solveFor`. If yes it performs the corresponding substitutions in the other equations. If not it tries to analyze further jet variables occurring in the equations. But obviously this will not always suffice to decide functional independence. In this case an error message is issued.

`reduceMod` reduces a system with respect to another one, i.e. it uses the equations of the second system and their prolongations to simplify the first one. `autoReduce` reduces a system with respect to itself. The default implementation depends basically on the effectivity of `solveFor`, as it tries to solve each equation for its leading derivative and to substitute it in the other equations.

As we will see in the next section the power and the efficiency of the simplification routines and here especially of `simplify` are crucial for the performance of our environment in almost any calculation. Thus special care should be applied to them in the implementation of any domain belonging to `JBFC`. It is e.g. very useful, if `simplify` returns the system in such a form that its symbol is already as close as possible to its solved form. This does not only make the handling of the symbol easier, there is a high probability that the simplification of the prolonged system will also be much easier.

There are two operations for determining dimensions: `dimension` takes as arguments a system (i.e. a list of functions) and the order $q$ of the jet bundle of which the differential equation is a submanifold. It computes the dimension of this submanifold. `orderDim` takes the same arguments, but it assumes that all equations are of the same order as the jet bundle and calculates the dimension of the submanifold in $J_q\mathcal{E}/J_{q-1}\mathcal{E}$. In other words, if an empty list is passed as "system", then `dimension` returns $\dim J_q\mathcal{E}$ whereas `orderDim` returns $\dim S_qT^*X \otimes V\mathcal{E}$.

The default implementations of both procedures assume that the system is simplified and contains only functionally independent equations. It then subtracts the number of equations of the appropriate dimension. The reason for the introduction of `orderDim` will become clear in the next section, when we discuss the `dimension` procedure of `DifferentialEquation`.

The category `BaseFunctionCategory` specifies no additional operations. It contains only different default implementations for the procedures generating jet variables and for the coercion of jet variables to `BFC`. They result in an error message, if one attempts to generate a dependent variable or a derivative.

Simplification is especially easy in the case of linear systems. The domain `JetBundle-LinearFunction` represents linear functions as a list of coefficients and a list of jet variables. This explains why `JBC` introduces the special "jet variable" `1`: it is here necessary to allow for a constant term. `simplify` calculates a row echelon form for the system. Since the lists in the chosen representation are always sorted, this puts the symbol at once in solved form. The coefficient domain must belong to `BFC`. Currently the only possibility is `JBX`. One could of course easily implement a special domain for functions with constant coefficients, but this is hardly worth while.

Simplification becomes a much more difficult problem for general expressions as implemented in `JetBundleExpression`. The current implementation assumes a polynomial dependency of the jet variables. It was originally planned to provide a special domain for polynomial equations, but there have been very bizarre problems with AXIOM. For this reason `JBE` takes currently the place of such a domain.

Polynomial equations are probably the most general equations for which there exists a general algorithm to compute dimensions of submanifolds (or better varieties). In the jet bundle the differential equations become algebraic equations. They generate an ideal. Using a Gröbner basis [9] it is easily possible to determine the dimension of the variety defined by it. `dimension` applies the algorithm described by Kredel and Weispfenning [90].

This approach has, however, two serious disadvantages. Since we use the implementation of Gröbner bases provided in the AXIOM library, we cannot track the dependency of the resulting equations of the original equations. We will see in the next section that this is quite important for the domain `DifferentialEquation`. Furthermore the dimension of jet bundles and thus the number of unknowns occurring in the polynomials grows exponentially with the order. The same holds for the number of equations, if several prolongations are needed. But the complexity of Gröbner bases calculations explodes with the number of unknowns and the number of polynomials.

The implementation of `simplify` in JBE tries therefore to avoid the use of the AXIOM `groebner` procedure as much as possible. The main strategy rests on the observation that differential equations are usually sparse, i.e. not every jet variable occurs in every equation. A large part of the simplification can often be done by simply sorting the equations according to their leading derivatives. If there are several equations with the same leading derivative, `simplify` checks whether one of them can be solved for it. In this case this equation can be substituted in the other ones. Only if this is not possible, a Gröbner basis is computed — but only for the equations with the same leading derivative.

One could argue that this approach does not necessarily compute a Gröbner basis for the complete system and that thus the use of the algorithm of Kredel and Weispfenning is not correct. But this is not true. We obtain this way indeed a Gröbner basis but not a reduced one! Obviously we construct a basis with the same leading monomials as a reduced basis. By a well-known characterization theorem for Gröbner basis this implies that we get indeed such basis. In any case the algorithm of Kredel and Weispfenning needs only these leading monomials.

Solving an equation for a given jet variable provides of course also problems, if general expressions are allowed. Currently `solveFor` can handle only the case that the function depends linearly on this variable. But it may be non-linear in other variables. One could of course try to treat some more general cases, but we do not think that this is worth while the effort. For the purpose of simplification this strategy usually suffices, as prolongations lead to quasi-linear equations.

The domain `JetLazyFun` is rather special and still in an experimental state. But it shows the great possibilities offered by a system like AXIOM. The motivation for its development have been the following considerations. If we compare the calculations performed in our completion algorithm 2.3 with other approaches to the construction of integrability conditions, it seems that we perform many unnecessary differentiations.

Most other approaches (take e.g. Reid's standard form [124]) prolong at first sight only those equations where one can suspect from the leading derivatives that cross-derivatives may generate integrability conditions. In our approach we must always prolong *all* equations in order to obtain all independent equations in the jet bundle of the next order. In principle, Proposition 2.9 tells us that for the generation of integrability conditions only the prolongations with respect to non-multiplicative variables are of interest. The other prolongations yield automatically independent equations. Zharkov and Blinkov [165, 166] proposed a new approach to Gröbner bases (of algebraic equations!) based on the completion of an equivalent linear system of partial differential equations to an involutive one. Their algorithm tries to exploit this fact by considering only non-multiplicative prolongations.

A closer analysis reveals, however, that the difference between the two approaches is not as big as it seems. The generation of the potential integrability conditions alone does not suffice. One must also check whether they represent truly new independent equations. This requires simplification and hence reduction by other equations. But now one needs the prolongations with respect to the multiplicative variables, too. Algorithms like the one proposed by Zharkov and Blinkov or like the standard form algorithm of Reid compute these prolongations each time they are needed new. Thus there also occur redundancies.

A priori it is not clear which approach will lead to less unnecessary differentiations.

One could argue that this is only a problem of implementation. One could store each computed prolongation for later reuse. But now the question arises how this can be organized efficiently. `JLF` could be viewed as a kind of synthesis of these two apparently so different approaches. It puts on top of another domain of `JBFC` a lazy evaluation scheme for partial differentiations. This means that the procedure `differentiate` does not really compute a partial derivative but only stores a pointer to the function to be differentiated and the variable with respect to which it is differentiated. Only if later the full expression is needed, the differentiation is actually performed.

Such lazy evaluation schemes have been successfully applied to the implementation of infinite objects like series [19]. There the idea is to compute only as many terms as momentarily necessary, but to be able to calculate more terms if they are needed. In our case the idea is that the most expensive part of a prolongation, namely the many partial derivatives, is not really performed. As in the approach of Zharkov and Blinkov only the prolongations which are absolutely necessary to decide the existence of integrability conditions are computed.

Of course there is a price to be paid. The representation of `JLF` is a complicated three level hierarchy connected by pointers. To avoid as many evaluations as possible information about the leading derivatives and the occurring jet variables are stored. This introduces a considerable overhead. Thus the use of `JLF` is only effective, if one can really omit many differentiations. We will discuss this again in Section 8.7 based on examples.

The guiding principle in the implementation of `JLF` is to use the leading derivative of a function as much as possible. Testing for zero is e.g. always a very important operation: `Ring` specifies a special operation `zero?` for this purpose. Many rings provide a special implementation of it which is more effective than the use of `=`. This is also the case in `JLF` where `zero?` uses the fact that the leading derivative of zero is the special jet variable `1`. Thus only if `leadingDer` yields this value an evaluation is necessary to see whether the function really vanishes.

This makes of course the effective implementation of `leadingDer` especially important. This is reflected in the choice of the representation. The lowest level represents a "lazy derivative". It consists of a record containing a function, a variable and a flag. If the value of the flag is `false`, then the differentiation has not yet been performed. The function is then the one to be differentiated. If the flag's value is `true`, the function represents already the result.

The next level represents a "lazy term", i.e. a coefficient and a pointer to a lazy derivative. In addition there are two flags and a jet variable. One flag indicates again whether or not the term has already been evaluated. In this case only the coefficient is relevant, as it contains the result. The jet variable provides an upper bound for the leading derivative of the term. The second flag indicates whether or not this bound is sharp, i.e. whether it is only a bound or really the leading derivative.

The top level record contains besides a list of pointers to lazy terms two flags, two lists of jet variables and a single jet variable. This single variable represents again an upper bound for the leading derivative of the total expression. One of the flags shows whether it is a sharp bound. The first of the two lists of jet variables represents all variables of which it is known that they occur effectively in the expression. The second one contains those which may occur. The list of lazy terms is considered as a sum. It is ordered by the leading derivatives of the terms. The second flag again indicates whether there are still unevaluated terms present. The representation of `JLF` is a pointer to such a record.

Pointers (in AXIOM realized with the domain `Reference`) are chosen in order to "communicate" the result of an evaluation. This is necessary, because arithmetic operations lead to a kind of proliferation of lazy derivatives or terms. If now such a lazy object is evaluated, the result should be available in all expressions containing it. This can be ensured through the use of pointers.

With this representation it is easy to implement `leadingDer`. It looks first whether the stored leading derivative represents only an upper bound. If not it is returned and no evaluation was performed. Otherwise it starts using the procedure `eval1` to evaluate as many of the lazy terms as necessary to obtain a sharp bound.

We have already mentioned that `zero?` (and similarly `one?`) is based on `leadingDer`. But many procedures implemented categorically in `JBFC` use `zero?`, e.g. `jacobiMatrix` to avoid vanishing entries. This would lead to many unwanted evaluations. We have introduced the attribute `lazyRep` to distinguish domains like `JLF` with a lazy evaluation mechanism. In the case of such a domain `JBFC` tries to avoid procedures like `zero?` which might cause evaluation as much as possible. The same holds for the matrix domain `SEM` discussed in Section 8.6.

As explained above evaluation of a lazy derivative or a lazy term may effect several expressions. Thus an expression might have a more complicated representation than necessary, as some of its lazy terms are already evaluated. The procedure `collect` collects all such terms and leads hence to a simplification of the representation. It causes, however, no evaluation and changes of course not the value of the expression.

The implementation of `simplify` is very close to the default one in `JBFC`. As far as possible only the leading derivatives are used. But if several equations with the same leading derivatives occur, they must be evaluated either to try to solve them or to use the `simplify` procedure of the base domain. Many operations of `JBFC` cannot be implemented without evaluation. Hence they should be avoided as much as possible. They include `subst`, `solveFor` and all reduction procedure.

## 8.5 Implementation III — Higher Structures and Applications

This section describes some details of the implementation of the layers 3 and 4. The core of the third layer is of course the domain `DifferentialEquation`. As already explained in Section 8.3 its main purpose is to enhance the efficiency by storing some intermediate results. The two other domains `VectorField` and `Differential` could also be considered as utilities.

In principle we could use simply a list of functions of some domain in `JBFC` to represent a differential equation locally. The domain `DE` has, however, a more complicated representation. The most important point is that all equations are sorted by order. This allows to keep previous results, especially previous simplifications, almost unchanged by prolongations, as only integrability conditions can affect the lower orders. Such considerations play an important role for an efficient implementation of the completion algorithm outlined in Section 2.4.

The chosen data structure holds much more information than just the equations. The top level is a record of two lists: one contains the effectively occurring orders, the other one for each order a record. This record consists of a list of equations, namely all equations of the given order in the system, the Jacobi matrix of these, and some flags indicating whether the equations of this order are already simplified (`Simp?`), whether they have already been prolonged (`Prolonged?`), or whether their contribution to the dimension of the differential equation (see below) has already been computed (`Dim?`).

Additionally the record contains an integer list (`Deriv`) to avoid the multiple generation of the same equation during prolongations. If an equation was obtained by prolonging another one, this list contains the number of the independent variable with respect to which that equation was prolonged. In the next prolongation the equation will be prolonged only with respect to independent variables with an index greater than or equal to this number.

Because of this list we try to keep track of the equations during simplification. If an equation is a combination of several other equations, then its value in `Deriv` is determined by the minimum of the values of the other equations. This strategy may not be optimal, but it is the best one which can be realized with reasonable effort.

The central operations in `DE` are `prolong`, `project`, `dimension` and `simplify` plus the ones for symbol and tableau. `simplify` is of course based on the corresponding procedure in `JBFC` and assumes that it finds all integrability conditions. It proceeds order by order starting with the equations of highest order. If at some order an integrability condition is detected, it is at once moved to the subsystem of the corresponding order. If algebraic equations occur, `simplify` checks whether they pose conditions on the independent equations. In that case an error message `"inconsistent system"` is issued. As soon as a subsystem has been simplified, the flag `Simp?` is set. If `simplify` is called again, it recognizes this flag and avoids a second, unnecessary simplification.

In the implementation of `prolong` one must take care of both mechanisms to generate integrability conditions: the classical one via elimination of the derivatives of highest order and the prolongation of equations of lower order (cf. Example 2.4). The first one is basically the task of `simplify` applied to the equations obtained by formally differentiating the highest order equations. To avoid unnecessary differentiations of lower order equations (and the subsequent simplification!) the list `Prolonged?` stores for each equation a flag whether or not its prolongation has already been computed.

There exists also a second version of `prolong` requiring two arguments, the differential equation and an order. Only for the equations up to this order it is checked whether their prolongations have already been computed. This version is useful in the completion procedure described in the next section. It allows for the fast prolongation of a system obtained by a projection. In this case only the formal derivatives of lower order equations, namely of the integrability conditions determined during the projection, are needed.

`project` can be implemented very easily, if one assumes that it is applied only to prolonged equations which have already been simplified. This assumption is reasonable within the current context of completion to involution. For other tasks it might be useful to have a more general procedure available. It is, however, not obvious, how it could be realized for arbitrary classes of equations, as it requires then highly non-trivial, non-linear computations. With this assumption `project` amounts to nothing more than eliminating all records belonging to equations of higher order than the target order.

The determination of the dimension of a differential equation is a crucial operation in the completion algorithm 2.3. `DE` takes the following approach: Using the procedure `orderDim` of `JBFC` the contribution of each order is computed separately. It is then stored in the record containing the equations of this order and the flag `Dim?` is set. As long as no integrability conditions of this specific order are appearing this result can be reused, if e.g. the dimension of the prolonged equation must be computed. In the optimal case this requires only the determination of the contribution of the new equations of highest order.

`DE` specifies four operations for symbols: `extractSymbol` extracts it from the Jacobi matrix of the highest order part of the system. `analyseSymbol` returns a record with the values of the $\beta_q^{(k)}$, the total number of multiplicative variables and the rank of the symbol.

prolongSymbol computes directly the prolonged symbol instead of extracting it from the prolonged differential equation. prolongMV takes as argument a record as computed by analyseSymbol and returns the corresponding record for the prolonged symbol. The values are, however, only correct, if the symbol is involutive, as this procedure is based on Lemma 3.6.

Additionally there are two procedures tableau to compute the matrices of tableaux using (2.42) and (2.58), respectively. One takes only one one-form as argument. The other one expects a list of one-forms and can thus be used to compute $k$-tableaux. To enter one-forms the domain Differential must be used. It represents together with the domain VectorField the remainder of the third layer. The implementation of both is somewhat rudimentary, as we hope that some day AXIOM will contain a reasonable environment for differential geometric calculations and then it should be used instead of some special domains like the two mentioned.

Both use an identical representation consisting of two lists. One list contains the coefficients, the other one the basis elements. The domains for both are passed as arguments. The one for the basis elements must be a jet bundle from JBC, the other one a domain belonging to JBFC. The lists are always sorted using the lesser function of JBC.

Besides some elementary procedures to enter one-forms and vector fields, respectively, or to access the coefficients and basis elements we have implemented the usual arithmetic operations and a few geometric operations like commutators or the contraction of a one-form with a vector field. For vector fields we have additionally implemented a procedure prolong which prolongs a field given on the base bundle $\mathcal{E}$ to one acting on a jet bundle of given order using to (5.4). Such a procedure is necessary for symmetry analysis.

The fourth layer contains currently the packages CartanKuranishi and SymmetryAnalysis. CK's central procedure is called complete and completes a given differential equation to an involutive one. Whereas complete only displays some results (depending on the previous setting of the output flags using setOutMode) and returns nothing (or void in the AXIOM language), its second version complete2 returns the involutive equation, the Cartan characters and a few other information. The user can set two further flags controlling reduction and simplification with setRedMode and setSimpMode. They are discussed in more detail in Section 8.7.

complete implements basically the detailed algorithm depicted in Fig. 2.3 on page 23. To achieve a reasonable performance it takes full advantage of the possibilities offered by the implementation of DE. Basically all the computations happen in two local variables: PrevDe contains the current differential equation which is checked for involution; CurDe represents its prolongation which is of course needed for the check.

If the inner loop of our algorithm requires a prolongation, PrevDe gets the previous value of CurDe and CurDe is newly calculated using prolong. Then the necessary information about the symbols are extracted. If the symbol of PrevDe is involutive, involution of the differential equation is decided by comparing dimensions according to Theorem 2.7.

If integrability conditions occur, PrevDe is newly computed from CurDe using project. As explained above prolong puts such conditions directly in the correct place in the representation of DE. Hence the simple implementation of project suffices here. The new value of CurDe could now be computed from PrevDe, but this would be inefficient, as CurDe contains already all needed equations with the exception of the prolongations of the integrability conditions. Here complete makes use of the second form of prolong which computes only these equations.

Besides complete CK contains several procedures for calculations with the Cartan characters and the Hilbert polynomial. Especially it allows the user to compute the $\alpha_q^{(k)}$

from the $\beta_q^{(k)}$ and to calculate $H_q(r)$ from the $\alpha_q^{(k)}$ and vice versa. There is a procedure to evaluate the recursion relation (3.16) for a given differentiation order. Furthermore one can compute gauge corrections.

The implementation of these operations is straightforward from the formulae of Chapter 3. It only demands a procedure `stirling` to compute the modified Stirling numbers introduced in Appendix C.1. They are determined using the recursion relation (C.5). To avoid repeated calculations of the same values, a hash table is set up storing all computed numbers.

`SYMANA` is still fairly primitive and in an experimental stage. It provides essentially one procedure `detSys` to set up the determining system (5.5) for symmetry generators (cf. Chapter 5). There exist different mode maps for this procedure. One can e.g. provide a special ansatz for the symmetry generator or a list of derivatives for which the equations of the system can be solved. The default is the most general ansatz and each equation will be solved for its leading derivative.

If the general ansatz is chosen, `detSys` returns the determining system in another data type, namely as functions over a new jet bundle with the coefficients as dependent variables. In this form the output can directly be converted into a `DifferentialEquation` and further analyzed. An example for this is given in Section 8.8. If one is searching for classical symmetries, it is of advantage to retract the output to a linear system using `linearize`.

In its current form `SYMANA` will have problems to compute the determining equations for a large system. Like all currently available symmetry packages it substitutes the equations of the original system solved for given derivatives in the conditions for the symmetry generators. Then it extracts in one loop the coefficients of all monomials. This approach is very inefficient and yields often many redundant equations. A later version will use a more sophisticated approach setting up the equations step by step and simplifying them at once.

Another project for the next version is to get rid of the restriction to equations which can be solved for some derivatives. If a Gröbner basis of the original system is given, then one can compute a normal form of all conditions and obtain this way independent of any solving the determining system. Note, that we use here standard algebraic Gröbner bases and not differential ones. This suffices, as we work in a jet bundle of fixed order.

`SYMANA` contains also a procedure `ncDetSys` to obtain the determining system for conditional symmetries as they are used in the non-classical method. It simply adds the invariant surface condition corresponding to the given ansatz for the symmetry generator to the original system and then calls `detSys`.

## 8.6 Implementation IV — Utilities

Since it is one of the basic ideas underlying the formal approach to differential equations to use linear algebra instead of differential operations, it is obvious that an efficient treatment of linear systems of equations is important for the performance especially of the `complete` procedure in `CK`. The determination of the $\beta_q^{(k)}$ requires to transform the symbol in row echelon form; some domains in `JBFC` may use the Jacobi matrix in `simplify` or `dimension`, too.

Earlier versions of our environment used the standard `Matrix` domain of AXIOM. However, the matrices typically occurring in our applications have some special properties which can be better exploited by implementing a new domain. The most important one is that they are sparse, often even very sparse. This reflects simply the already mentioned fact that e.g. in a second-order equation usually not all jet variables up to order two are effectively occurring. Often even of the second order derivatives just one or two are present. (2.11) implies furthermore that all prolonged symbols have exactly the same percentage of non-vanishing entries, i.e. only integrability conditions can change the sparsity.

In a recent diploma thesis Berchtold [10] compared different algorithms and data structures to compute row echelon forms and determinants of sparse matrices. He showed that for integer matrices it is of advantage to use a special "sparse" data structure like lists or polynomials for the rows of the matrix. His conclusion that this would not be the case for matrices with polynomial entries is, however, not correct and essentially due to his incorrect implementation.

Most of the matrices studied by Berchtold were still fairly dense compared with the matrices typically appearing as symbols. For such matrices Gaussian elimination amounts essentially to sorting the rows according to the position of the pivot, as the probability that two rows have their pivots in the same column is rather small.

Another result of Berchtold's thesis was that factor-free elimination methods like e.g. the Bareiss algorithm [57] are fairly inefficient. They tend to enlarge the entries considerably. In the case of a square matrix the last entry on the diagonal will contain in the end the complete determinant of the matrix! In our applications this can thoroughly effect the overall performance. Assume the `simplify` procedure of a domain in `JBFC` is based on such an elimination method. If later the resulting equations must be prolonged, these complicated expressions must be partially differentiated several times.

On the other hand divisions like they are used in the standard Gauss algorithm lead to a similar effect, as differentiation of fractions is quite expensive. It is therefore better to stay somewhat in the middle between these two extrema. The so-called primitive factor-free elimination (apparently first introduced by Monagan in Maple) divides each row by its greatest common divisor. This approach avoids collecting factors as in the Bareiss method but still does not generate fractions.

The greatest common divisor of a row can usually be computed fairly efficient, as experience shows that very often already the first two non-vanishing entries determine the final result. Actually there are two possible strategies: The classical one is compute the greatest common divisor iteratively stopping as soon as 1 is reached. Zippel [167] recommends a probabilistic strategy building to random linear combinations of the elements and computing their greatest common divisor. With probability 1 this yields the correct result. But of course one must nevertheless check that one has not obtain a too large result. Which strategy is ultimatively the best is difficult to judge. `SEM` offers both; the user can choose it with the function `setGcdMode`.

These considerations[1] lead to the implementation of the domain `SparseEchelon-Matrix (SEM)`. Its internal data structure is especially designed for the fast computation of row echelon forms. On the other hand hardly any arithmetic operations like addition or multiplication are implemented, as these tend to destroy the sparsity and are furthermore rather inefficient in this special representation.

`SEM` takes two arguments: a domain belonging to `Ring` for the entries and a domain belonging to `OrderedSet` to label the columns. A row is now represented by a record containing two lists: one for the non-vanishing entries, one for their labels. The latter one is always sorted. A matrix consists essentially of a vector of such records.

The central operations of `SEM` are `rowEchelon` and `primitiveRowEchelon`. Both return

[1] I am indebted here to M.B. Monagan who turned my attention to these problems and explained me his and Berchtold's results.

not only the matrix in row echelon form but also the corresponding transformation matrix, the used pivots and the rank of the matrix. `primitiveRowEchelon` is only available, if the domain of the entries belongs to the category `GcdDomain`. Furthermore, the transformation matrix contains then fractions from the divisions by the greatest common divisors.

The implementation of both operations is fairly different from the usual elimination methods. Instead of searching for a pivot in each column, we start by sorting all rows according to their pivots. After each elimination step changed rows are at once moved to their new places, so that the rows remain sorted all the time. The sorting is done using bubble sort. Since we can expect that our matrices are often already close to a row echelon form, this is probably the best choice. Furthermore it allows to keep easily track of the rows which is necessary for the construction of the transformation matrix.

The package `JetCoordinateTransformation` provides two procedures `transform` to prolong coordinate transformations of the base bundle $\mathcal{E}$ into higher order jet bundles. Its parameters are two jet bundles and two vectors (one for the independent and one for the dependent variables) containing expressions for the old coordinates in terms of the new ones. One procedure computes the transformation law for an old jet coordinate; the other one transforms an expression in the old coordinates using the first one and `subst`.

In the current implementation all expressions must be of the type `JBE`. The main reason for this restriction is that this represents by far the most flexible domain, as its representation is based on the domain `Expression Integer`. In it arbitrary transformation laws can be specified. Furthermore for most domains in `JBFC` it is possible to write a `retract` or `coerce` procedure to convert its objects into objects of `JBE` and vice versa.

The implementation of `transform` is straightforward. It makes use of the chain rule. [15] gives a simple recursive formula for the transformation law of a derivative of order $q$ given the one for a derivative of one order less. It requires the inversion of the Jacobi matrix of the transformation of the independent variables. This inverse matrix is computed using an LU decomposition[2]. To enhance the efficiency transformed jet variables are stored in a hash table similar to `stirling` in `CK`.

## 8.7 Examples I — Completion

This and the next section demonstrate the application of the above described environment to concrete examples. Some of them have already been discussed in the previous chapters. The mentioned running times refer to the AXIOM implementation on an IBM RS 6000 Model 530 with 64MB memory.

To start we treat again the classical Example 2.19 of Janet. The upper part of Fig. 8.4 on page 137 shows the necessary input. As first step the different needed domains and packages are constructed and assigned to variables to facilitate the input. We use a jet bundle (`jb`) with 3 independent variables named `x` and one dependent variable named `u`. Derivatives are denoted by `p`. For the equations we use linear functions (`jbl`) whose coefficients can be arbitrary expressions in the independent variables (`jbx`). `de` is the data type used to represent the differential equation; `ck` denotes the package with the completion procedure.

[2] We have implemented for this purpose a package `LUDecomposition` containing three procedures: `LUDecomp` computes a LU decomposition using the algorithm of Crout [121]; `LUSolve` uses such a decomposition to solve a linear system with given right hand side; `LUInverse` determines the inverse for a matrix with given LU decomposition.

The equation is set up in two steps: First the two equations (`eq1,eq2`) of system (2.24)

$$\mathcal{R}_2 : \begin{cases} u_{zz} + y\, u_{xx} = 0\,, \\ u_{yy} = 0\,. \end{cases}$$

are defined. Jet variables are generated with the procedures `X, U, P`. Then a differential equation (`janet`) is generated from this system. The `setOutMode` command controls the amount of output generated and whether or not it should be in TEX. The `setRedMode` command tells `complete` to reduce the found integrability conditions as much as possible. Finally the procedure `complete` from `ck` is called. In the last two lines we have helped the interpreter to locate the used procedures by explicitly stating that they are from the package `ck`.

The output of `complete` is also contained in Fig. 8.4. With the chosen setting of the output flags we get a trace of the dimensions of all intermediate systems and symbols and of all found integrability conditions. The output is directly in TEX. One can easily see that the computation are exactly the same as in our treatment of the Janet example in Section 2.4. It is, however, a non-trivial fact that the program produces the integrability conditions in the simple form shown in Fig. 8.4. Actually many more integrability conditions are produced, for there occur quite a lot of non-multiplicative variables in the higher prolongations. They are, however, not independent. `complete` must invoke several simplification procedures to avoid redundant equations and unnecessary coefficients. In this example it succeeds in reaching the simplest form, because we deal with a linear system. For non-linear equations this is not necessarily the case.

The running time for this example is about 18 sec. It rises to 28 sec, if one takes `JBE` as data type for the equations instead of `JBLF`. There are two main reasons for this striking difference: differentiations are much faster in `JBLF` and simplification of linear systems can be performed more efficient. Although in the end the `simplify` procedure of `JBE` produces the same result as the one in `JBLF`, the latter one finds the necessary steps easier and thus faster.

Due to the many prolongations necessary in this example it also demonstrates very nicely the advantages of our special domain for sparse matrices. Compared with an earlier version of our environment which used the standard `Matrix` domain of AXIOM the mentioned running time improved by more than a factor two. It is easy to see that system (2.24) leads indeed to very sparse matrices for the symbols. Furthermore the procedure `simplify` in `JBLF` computes already a row echelon form of the whole system and thus of course also of the symbol. Hence the call of `rowEchelon` in `analyseSymbol` is in principle redundant. But because of the special way the elimination algorithm is implemented this takes hardly any computing time.

One could think that this example should be an ideal test case for the lazy differentiation scheme implemented in `JLF`. However, the result is disappointing. Instead of accelerating the computation the running time rises to almost 1 min. A closer look solves the mystery. We encounter three times a non-involutive symbol. Each time two prolongations are necessary. Tracing the intermediate systems shows that after the second prolongation all lazy differentiations used in the previous one have been fully evaluated.

The reason is that there occur many non-multiplicative variables in the equations of higher order and deciding whether or not the corresponding prolongations generated integrability conditions requires the evaluation of most of the equations. Thus in this example it is indeed necessary to compute almost all prolongations. Another aspect of this at first sight perhaps surprising performance is that the basic assumption for the development of `JLF`, namely that partial differentiations are expensive, is not correct for linear equations. With the chosen representation of `JBLF` the differentiation with respect to a dependent variable or a derivative requires only the extraction of an element of a list. Since the lists are not too long, this is very fast.

As an example of a non-linear system we treat the Euler Equation of hydrodynamics in a slightly unusual way to obtain an over-determined system.

$$\begin{aligned} \partial_t u^i + \sum_{j=1}^{D} u^j \partial_{x^j} u^i &= 0\,, \qquad i = 1,\ldots,D\,, \\ \sum_{j=1}^{D} \partial_{x^j} u^j &= 0\,, \end{aligned} \tag{8.1}$$

where $D$ denotes the space dimension. In our computer calculation we take the usual value of $D = 3$ and write $x^4$ instead of $t$.

In- and output are shown in Fig. 8.5 on page 138. The input is very similar to the previous example. The main difference is that we use another domain for the equations, namely `JBE`. They can now be arbitrary expressions in the independent and dependent variables. The integrability condition can be obtained by taking the divergence of the first $D$ equations and subtracting the $t$-derivative of the last equation

$$\sum_{i,j=1}^{D} \partial_{x^i} u^j \partial_{x^j} u^i = 0\,. \tag{8.2}$$

Due to the simplification procedures this equation and some of the original equations appear different in the output. As explained in Section 8.4 the `simplify` procedure in `JBE` tries to solve equations for their leading derivatives and to substitute them into the other ones. If this strategy is successful, it leads to kind of normal form of the system which is better adapted for the analysis of the symbol.

The time used for simplification also dominates the running time. Although this example is much smaller than the Janet example in terms of the number of prolongations needed, it takes much longer, namely about 8 min. This time is really surprising, as the simplification does not require the computation of Gröbner bases. Nevertheless, it is frequently necessary to solve equations for their leading derivatives and to substitute the result in other equations. This process is quite time-consuming in `JBE`.

We have used our basic domains for jet bundles in these two examples: in the Janet example `JB`, for the Euler Equation `IJB`. One could see that the input of the equation was slightly more complicated for `JB`, because the interpreter needs some help in order to identify the types correctly. One can of course also use the syntax applied for the Euler Equation in `JB`. But this requires that one knows the used ranking of the variables. The purpose of `JB` is, however, to stay as close as possible to the natural notation.

The two examples differ also in the setting of the reduction and simplification control flags with `setRedMode` and `setSimpMode`. The choice of a good setting requires in some sense to know already a priori what will happen during the completion. In principle one would prefer to avoid the setting of both flags, as they only need computation time.

`setRedMode(1)` affects basically only the output. Without it many spurious integrability conditions and/or equations in the generating system displayed at the end may appear. They are either algebraically dependent of each other or derivatives of some other equations. Reduction tries to detect this. In examples like the Janet example where many prolongations are needed and integrability conditions appear at different orders, this is usually useful. In the case of the Euler Equation one would, however, gain nothing.

Deciding whether `setSimpMode(1)` helps or not is more difficult. As explained in Section 8.5 the individual equations of a system are stored order by order. With the default setting simplification only occurs within the different orders. Equations of lower order are not used. This may lead to more complicated expressions, because some simplifications are overlooked. For instance it might happen in the Janet example that some equations of higher order contain $u_{yy}$, although it is zero. But it turns out that in this example it is more expensive to search for such simplifications than to continue with the unsimplified expressions.

The situation is often different for non-linear systems. Here it is much more important to avoid complicated expressions, as otherwise the standard simplification within one order needs much more time. In the case of the Euler Equation one gains almost a factor of two through simplifying modulo the equations of lower order.

To stress this point we will briefly discuss a non-linear modification of the Janet example. Consider the system

$$\mathcal{R}_2 \,:\, \begin{cases} u_{zz} + v\,u_{xx} &= 0\,, \\ u_{yy} &= 0\,, \\ v_z &= 0\,, \\ v_y - v_x &= 0\,. \end{cases} \tag{8.3}$$

The completion runs completely analogously to the classical Janet example, i.e. the projections occur at the same places and $\mathcal{R}_5^{(2)}$ is involutive. The only difference is that in the second projection two integrability conditions[3] occur, thus there are three:

$$\begin{gathered} 2v_x u_{yxx} + v_{xx}u_{xx} = 0\,, \\ 4v_x^4 u_{xxxx} + v_{xx}(4v_x^2 - vv_{xx})(2v_x u_{xxx} + v_{xx}u_{xx}) = 0\,, \\ 2v_x v_{xxx} - 3v_{xx}^2 = 0\,. \end{gathered} \tag{8.4}$$

It is obvious that here the simplification routines have hard work to do. Indeed the running time for this example is slightly more than one hour. It was not possible to run it without simplification modulo lower order equations. We stopped the computation after about 10 hours!

We showed in Chapter 6 that the Dirac algorithm does nothing else than to complete the equations of motion to involution. Thus we can also use our environment for the analysis of constrained dynamics. Tombal and Moussiaux [154] presented a MACSYMA program for the Dirac algorithm. It can, however, handle only Lagrangians which are quadratic in the velocities. Furthermore they reported that their program was too slow to handle complicated systems like the Maxwell Equations. Our environment is not only capable of treating much more general systems, it proves the involution of the Maxwell Equation in a four-dimensional space-time in about 15sec.

As a simple example we will consider again the motion of a particle in three dimensions whose motion is confined to the 2-sphere of radius 1 (cf. Example 6.5). Setting all masses to 1 we have to analyze the following equations of motion

$$\mathcal{R}_2 \,:\, \begin{cases} \ddot{x} - 2\lambda x = 0\,, \\ \ddot{y} - 2\lambda y = 0\,, \\ \ddot{z} - 2\lambda z = 0\,, \\ x^2 + y^2 + z^2 - 1 = 0\,. \end{cases} \tag{8.5}$$

[3] We count here only the non-trivial conditions. Of course there appear in addition the prolongations of the two first-order equations.

The output of a corresponding AXIOM session is shown in Fig. 8.6 on page 139. $\lambda$ is denoted by $L$. Since we are dealing with ordinary differential equations, there is no need to check for involution of the symbol. The running time is slightly over 30sec with both simplification and reduction flag set.

It might surprise that such a small example consisting only of ordinary differential equations should take twice as long as the Maxwell Equations. Especially if one takes into account that no symbols must be analyzed. But the reason is simply that the latter ones are linear. If we omit setting the simplification flag, i.e. `setSimpMode(0)`, the running time nearly doubles.

A closer analysis of the obtained integrability conditions and the displayed final system reveals some deficiencies in the treatment of non-linear equations. We will discuss them later in more detail. Here we just want to point out that in several equations the constraint was not used for simplification; in the third projection we even obtain two identical equations. Both effects are mainly due to current restrictions in the `solveFor` procedure of `JBE`.

## 8.8 Examples II — Other Applications

One of the classical examples in symmetry analysis is the Heat Equation. We will use `SYMANA` to calculate its determining system for classical Lie symmetries. We omit the further discussion of the symmetries, as it can be found in many places. The output of a corresponding AXIOM session is contained in Fig. 8.7 on page 140.

The first command just shows the ansatz made for the symmetry generators. If `detSys` receives only one argument, it takes automatically this most general ansatz, where every coefficients depends on all variables, i.e. in this example on $x, t, u$. The type of the output of `detSys` depends on the input. If one uses the general ansatz as in Fig. 8.7, one obtains a list of functions of type `JBE` over a new jet bundle in which the coefficients of the ansatz are the dependent variables. Thus the result can be used directly to generate a differential equation and then be further analyzed.

Since we are only looking for classical symmetries, we know that the determining system must be linear. Hence before generating a differential equation the individual equations are retracted to a new type, namely `JBLF`. Then the completion procedure is started. It turns out that we must proceed to third order and that four projections are necessary. The resulting equation can easily be solved explicitly (see e.g. [15, 111]).

Here we are only interested in calculating the size of the symmetry algebra. From the displayed Cartan characters we see that the general solution of the determining system contains two functions of one argument. But this is due to the linearity of the equation: solutions can be superposed. Indeed, we see that one of the equations in the determining system is just the Heat Equation itself.

In general it is difficult to deduce from the knowledge of the Cartan characters and of the dimension of the differential equation the number of arbitrary constants in the general solution, as one must take the lower order part of the arbitrary functions into account. These are usually not independent and thus difficult to count. The only way to do it is to construct a formally well-posed initial value problem. Here we can proceed simpler. The Heat Equation yields a seven-dimensional submanifold in a third-order jet bundle with two independent and one dependent variables. Hence its general solution contains seven arbitrary coefficients of order less than or equal to 3. Since the determining system is 13-dimensional, there remain 6 arbitrary constants for the finite-dimensional part of the symmetry algebra. Thus we find besides the superposition 6 linearly independent symmetry generators.

Setting up the determining system needs about 12 sec. The completion is, however, fairly slow and takes a bit longer than 90 sec. This is mainly due to the prolongation to third order. One should mention that the Heat Equation is in some respect a "lucky" example. Fairly often determining systems turn out to be in not $\delta$-regular coordinate system. Then one cannot simply call `complete` but must start working with the $k$-tableaux.

Finally, we consider an example of a coordinate transformation. It is well-known that the Burgers Equation is related to the Heat Equation via the Cole-Hopf transformation. We will perform this transformation using the AXIOM package JCT starting with the Burgers Equation in potential form. The result can be seen in Fig. 8.3.

```
burgers:jbe1 := P [2] - P [1,1] - P([1])**2

                            2
   (8)  - u    + u  - u
           x,x    t    x
                                 Time: 0.33 (IN) + 0.23 (EV) + 0.80 (OT) = 1.36 sec
transform(burgers)$jct

        - v    + v
           y,y    s
   (9)  -----------
             v
                                 Time: 0.22 (IN) + 2.02 (EV) + 1.54 (OT) = 3.78 sec
transform(burgers)$jct

         - v    + v
            y,y    s
   (10)  -----------
              v
                                 Time: 0.01 (IN) + 0.35 (EV) + 0.05 (OT) = 0.41 sec
```

Figure 8.3: Cole-Hopf transformation of the Burgers Equation.

The timing of the first transformation also contains the time needed for the precomputation of an inverse Jacobi matrix. Since its result is stored, subsequent transformations will be faster. In this example this effect is neclectable, as the independent variables remain unchanged and the mentioned Jacobi matrix measures their dependence of the old independent variables. JCT further keeps as hash table of already treated jet variables. Hence the second call with the same argument is much faster.

## 8.9 Outlook and Discussion

It should be clear from the discussion so far, that this environment is by far not yet finished. There are permanent changes and improvements. Most changes are needed in the domains for non-linear equations. The simplification and reduction procedures implemented in JBE are still too inefficient for more complicated equations. This is partly due to its current role as a substitute for a special domain for polynomial equations. Although the procedures are mainly designed for the application to polynomial equations, there had to be made concessions for more general expressions.

The most important point is reduction. In a polynomial domain one could make full use of the auto reduction algorithms proposed already by Ritt [128]. Currently `JBE` inherits the default implementation of `autoReduce` in `JBFC` which tries to solve equations for their leading derivatives and to substitute the results in the other equations. This can be very time consuming for complicated equations. Similar remarks can be made for simplification.

This effect is clearly visible in the example of the Euler Equation. Although the completion algorithm terminates after just one iteration, the computing time is much higher than in the Janet examples with its many prolongations and projections. This is especially remarkable, as a closer trace reveals that no Gröbner basis had to be calculated. If one looks at the second equation of the final result in Fig. 8.5 on page 138, one remarks that it has become considerably more complicated. This stems from “simplification”!

Even worse things are happening in the completion of the constrained system (8.5) where not only the constraint was several times not used for apparently easy simplifications and in one projection two identical integrability conditions were shown. The first effect will disappear as soon as a truly polynomial domain will be implemented. The current simplification and reduction routines rely heavily on the ability of `solveFor`. It in turn cannot treat products, if several factors contain jet variables. Since it is possible that one factor might be zero, `solveFor` returns in such cases `"failed"` to omit wrong results in the simplification.

Another point which will have to be addressed in further versions is regularity. As soon as we treat non-linear equations ranks and characters may differ on different points of a differential equation (cf. Example 2.2). If this happens, case distinctions are necessary. It is probably not efficient to treat automatically all occurring cases, as branching can occur in each step of the completion. But the output of a procedure like `complete` should at least show where problems can arise. Currently it follows just one branch without specifying the conditions.

To detect non-regularity one must inspect pivots. If a prolonged equation is not a manifold, then its Jacobi matrix has different ranks on different parts of it. The determination of a row echelon form, say for a symbol, makes usually assumptions that certain expressions, namely the used pivots, are non-vanishing. The procedures for row echelon forms in `SEM` always return the pivots; however, the higher level domains and packages like `DE` or `CK` make currently no use of this information.

Sit [144] developed an algorithm which determines all necessary case distinctions in determining the rank of a symbolic matrix. It is based on the primary decomposition of algebraic varieties. He argued that an analysis based on pivots may lead to many unnecessary branches. While this is surely correct from a theoretical point of view, his method is probably too expensive to be applied automatically in an already expensive completion algorithm. Imagine all matrices occurring in the completion of the modified Janet example in Section 8.7 would be analyzed this way!

Besides simplification and reduction $\delta$-regularity provides the most trouble. Currently `complete` in `CartanKuranishi` can handle only equations given in a $\delta$-regular coordinate system. Otherwise it will prolong and prolong and prolong without ever finding an involutive symbol. In principle one could implement the method present in Section 2.7 using $k$-tableaux. We have refrained from this approach, because it becomes quickly computationally very demanding.

Computing row echelon forms of symbolic matrices is in general very expensive. In computer algebra a matrix is considered as large, if it has more than, say, 20 rows and columns (of course this depends on the complexity of the entries). For the symbol we could partially remedy this by using a special domain for sparse matrices. This allows to handle efficiently considerably larger matrices. In the Janet example one encounters e.g. matrices with more than 60 rows. The tableaux are, however, always dense. One reaches quickly machine limits, if several prolongations are needed or the system is large.

Often one can detect already by inspection that a coordinate system is not $\delta$-regular. One possible strategy is to use the tableaux to find a non-systatic basis of $T^*X$ and an associated coordinate system. Then one rewrites the equation in this system and starts the completion algorithm. Of course it can always happen that the coordinate system will become again singular after some projections, but often this approach succeeds.

Our symmetry package is the first one developed in AXIOM. The REDUCE package SPDE [134] was translated [136] to its predecessor system Scratchpad II. It became, however, soon obsolete as new versions of Scratchpad II and later AXIOM appeared. `SYMANA` is still in a very early and experimental stage. The evaluation of the condition for symmetry generators is primitive compared with more sophisticated approaches, as they are e.g. described in [29].

Nevertheless, it demonstrates clearly the advantages of implementing a general environment: the basic procedure for setting up the determining equations consists of about 15 lines! Everything else is essentially provided by other domains. The many simplification and reduction routines originally implemented to support the completion procedure in `CK` are very useful to simplify determining systems. By using different domains in `JBFC` different simplification strategies can be applied.

We believe that, although there exist meanwhile so many symmetry packages, it is still possible to make significant progress in this field, even if one restricts ones attention to classical Lie symmetries (the extensions to more general symmetries are fairly straightforward anyway). We have already mentioned the possibility to handle equations which cannot be solved for some derivatives with Gröbner bases. One will have to wait whether it will be possible to provide an efficient implementation of this approach.

Another possibility for improvements is to make use of theoretical results on the form of admissible symmetry generators. Bluman [13] showed that at least for semi-linear equations one can simplify the ansatz for the generators considerably, because the coefficients' dependency of the dependent variable is very often trivial. This can save considerable computation time. First the prolongation of the generator is simpler and will lead to fewer terms, hence the resulting determining system is simpler. A lot of the work done by the packages to solve determining systems is just to rediscover through lengthy computations these simplifications!

```
jb:=JB(['x,'y,'z],['u])
jbx:=JBX jb
jbl:=JBLF(jb,jbx)
de:=DE(jb,jbl)
ck:=CK(jb,jbl)

eq1 := D('u,['z,'z])$jb::jbl + 'y::jb::jbl * D('u,['x,'x])$jb::jbl
eq2 := D('u,['y,'y])$jb::jbl
janet:de := generate [eq1,eq2]

setOutMode(14)$ck
setRedMode(1)$ck
complete janet
```

---

Symbol $M_2$ not involutive! Dimension: 4

Symbol $M_3$ involutive! Dimension: 4

Equation $R_3$ not involutive! Dimension: 12

======= 1. Projection =======

Integrability condition(s)

$$u_{y,\,x,\,x} = 0$$

=============================

Symbol $M_3^{(1)}$ not involutive! Dimension: 3

Symbol $M_4^{(1)}$ involutive! Dimension: 2

Equation $R_4^{(1)}$ not involutive! Dimension: 13

======= 2. Projection =======

Integrability condition(s)

$$u_{x,\,x,\,x,\,x} = 0$$

=============================

Symbol $M_4^{(2)}$ not involutive! Dimension: 1

Symbol $M_5^{(2)}$ involutive! Dimension: 0

*************** Final Result ***************

Equation $R_5^{(2)}$ involutive!

System without prolonged equations. Dimension: 12

$$\begin{gathered} u_{x,\,x,\,x,\,x} = 0, \\ u_{y,\,x,\,x} = 0, \\ u_{z,\,z} + y\; u_{x,\,x} = 0, \\ u_{y,\,y} = 0 \end{gathered}$$

System of finite type.

---

Figure 8.4: In- and output for the Janet example (2.19).

```
jb:=IJB('x,'u,'p,4,3)
jbe:=JBE jb
de:=DE(jb,jbe)
ck:=CK(jb,jbe)

eq1:jbe := P(1,[4]) + U(1)*P(1,[1]) + U(2)*P(1,[2]) + U(3)*P(1,[3])
eq2:jbe := P(2,[4]) + U(1)*P(2,[1]) + U(2)*P(2,[2]) + U(3)*P(2,[3])
eq3:jbe := P(3,[4]) + U(1)*P(3,[1]) + U(2)*P(3,[2]) + U(3)*P(3,[3])
eq4:jbe := P(1,[1]) + P(2,[2]) + P(3,[3])
euler:de := generate [eq1,eq2,eq3,eq4]

setOutMode(14)$ck
setSimpMode(1)$ck
complete euler
```

---

Symbol $M_1$ involutive! Dimension: 8

Equation $R_1$ not involutive! Dimension: 11

======= 1. Projection =======

Integrability condition(s)

$$2\ p_2^3\ p_3^2 + 2\ p_1^3\ p_3^1 + 2\ {p_2^2}^2 + 2\ p_1^1\ p_2^2 + 2\ p_1^2\ p_2^1 + 2\ {p_1^1}^2 = 0$$

============================

Symbol $M_1^{(1)}$ involutive! Dimension: 7

*************** Final Result ***************

Equation $R_1^{(1)}$ involutive!

System without prolonged equations. Dimension: 14

$$\begin{gathered}
p_4^3 + u^2\ p_2^3 - u^3\ p_2^2 + u^1\ p_1^3 - u^3\ p_1^1 = 0,\\
p_4^2 - u^3\ p_1^3\ p_3^1 + \left(u^2\ p_2^2 + u^1\ p_1^2\right)\ p_2^3 - u^3\ {p_2^2}^2 - u^3\ p_1^1\ p_2^2 - u^3\ p_1^2\ p_2^1 - u^3\ {p_1^1}^2 = 0,\\
p_4^1 + u^3\ p_3^1 + u^2\ p_2^1 + u^1\ p_1^1 = 0,\\
p_3^3 + p_2^2 + p_1^1 = 0,\\
p_2^3\ p_3^2 + p_1^3\ p_3^1 + {p_2^2}^2 + p_1^1\ p_2^2 + p_1^2\ p_2^1 + {p_1^1}^2 = 0
\end{gathered}$$

Cartan characters: 3, 3, 1, 0

---

Figure 8.5: In- and output for the Euler Equation (8.1).

Equation $R_2$ not involutive! Dimension: 8

======= 1. Projection =======

Integrability condition(s)

$$2\ z\ z_t + 2\ y\ y_t + 2\ x\ x_t = 0$$

==============================

Equation $R_2^{(1)}$ not involutive! Dimension: 7

======= 2. Projection =======

Integrability condition(s)

$$\left(2\ z^2 + 2\ y^2\right)\ {y_t}^2 + 4\ x\ y\ x_t\ y_t + \left(2\ z^2 + 2\ x^2\right)\ {x_t}^2 + 4\ L\ z^4 + \left(4\ L\ y^2 + 4\ L\ x^2\right)\ z^2 = 0$$

==============================

Equation $R_2^{(2)}$ not involutive! Dimension: 6

======= 3. Projection =======

Integrability condition(s)

$$\begin{gathered} 4\ L_t\ z^2 + \left(4\ y^2 + 4\ x^2\right)\ L_t = 0, \\ 4\ L_t\ z^2 + \left(4\ y^2 + 4\ x^2\right)\ L_t = 0 \end{gathered}$$

==============================

Equation $R_2^{(3)}$ not involutive! Dimension: 5

======= 4. Projection =======

Integrability condition(s)

$$\left(4\ z^2 + 4\ y^2 + 4\ x^2\right)\ L_{t,\,t} = 0$$

==============================

************** Final Result **************

Equation $R_2^{(4)}$ involutive!

System without prolonged equations. Dimension: 4

$$\begin{gathered} y_{t,\,t} - 2\ L\ y = 0, \\ x_{t,\,t} - 2\ L\ x = 0, \\ \left(4\ z^2 + 4\ y^2 + 4\ x^2\right)\ L_{t,\,t} = 0, \\ z\ z_t + y\ y_t + x\ x_t = 0, \\ \left(2\ z^2 + 2\ y^2\right)\ {y_t}^2 + 4\ x\ y\ x_t\ y_t + \left(2\ z^2 + 2\ x^2\right)\ {x_t}^2 + 4\ L\ z^4 + \left(4\ L\ y^2 + 4\ L\ x^2\right)\ z^2 = 0, \\ \left(-z^2 - y^2\right)\ {y_t}^2 - 2\ x\ y\ x_t\ y_t + \left(-z^2 - x^2\right)\ {x_t}^2 - 2\ L\ z^4 + \left(-2\ L\ y^2 - 2\ L\ x^2\right)\ z^2 = 0, \\ 4\ L_t\ z^2 + \left(4\ y^2 + 4\ x^2\right)\ L_t = 0, \\ 4\ L_t\ z^2 + \left(4\ y^2 + 4\ x^2\right)\ L_t = 0, \\ z^2 + y^2 + x^2 - 1 = 0 \end{gathered}$$

System of finite type.

Figure 8.6: Output for the constrained system (8.5).

```
v:vf := ansatz()$sym

   tau D  + xi D  + eta D
        t       x        u

ds := detSys([eq])$sym

   [- 2tau , - 2tau , - xi   , - tau   , eta    - 2xi   , - 2tau    - 2xi ,
          u         x      u,u       u,u     u,u      u,x        u,x      u
    2eta    - xi    + xi , - tau    - 2xi  + tau , eta    - eta ]
        u,x     x,x     t       x,x      x      t     x,x      t

lds:List jbl2 := [retract(eq)  for eq in ds]
r2:de := generate lds
setOutMode(4)$ck
setRedMode(1)$ck
complete r2

   *************** Final Result ***************

             (4)
   Equation R     involutive!
             3
   System without prolonged equations. Dimension: 13

       tau     = 0
          t,t,t
        eta   = 0
           u,u
            1
    eta    + - xi = 0
       u,x   2   t
     eta    - eta = 0
        x,x      t
           1
   eta    + - tau   = 0
      u,t   4    t,t
         xi   = 0
           t,t
          xi = 0
            u
         tau = 0
            u
           1
     xi  - - tau = 0
       x   2    t
         tau = 0
            x

   Cartan characters: 2,0,0
```

Figure 8.7: Determining system for the Heat Equation.

# Chapter 9

# Conclusions

*It is too bad that it has to be mathematics...*

*R.P. Feynman: The Character of Physical Law*

Feynman devoted one of his BBC lectures on the character of physical laws to the relation between mathematics and physics. His point of view was that mathematics represents the language of physics. It provides methods to reason about the physical laws and to deduce further statements from them. This thesis should be understood in this spirit. We have presented a mathematical machinery centered around the notion of involution. Then we begun to show how it can be used to obtain physical results.

As already indicated in the introduction it was one of our main goals to demonstrate that the formal theory of partial differential equations provides useful tools for physics. We hope that we have succeeded in this point, even though we restricted our attention mainly to initial value problems[1]. Although the theory is surely non-trivial, it allows for a clear and unifying treatment of many problems.

We believe that especially the treatment of systems with constraints can benefit from the formal theory. First of all it yields a rigorous foundation for the Dirac algorithm to construct the constraints. Completion to involution is just the mathematical formulation of Dirac's starting point, namely to check whether or not the equations of motion are consistent. We have also seen that it yields a unifying framework for Lagrangian and Hamiltonian formulation as well as for the symplectic formalism.

The next steps would be to obtain an intrinsic formulation for the Legendre transformation and an intrinsic characterization of first and second class constraints. One can say that basically the Legendre transformation uses a different embedding of $J_2\mathcal{E}$ into $J_1(J_1\mathcal{E})$ than the one we employ in Appendix C.2. The distinction between first and second class constraints should be visible in projections of the equations of motion to lower order jet bundles.

We have further seen that in the case of field theories the Dirac algorithm must use a kind of involution analysis as soon as there are constraints of lower class which require to take spatial derivatives into account. But then it is conceptually much clearer to start from the very beginning with such an analysis instead of keeping an artificial distinction into temporal prolongations which are done using Poisson brackets and spatial one which are performed in the usual way. One must also keep always in mind that the involution analysis can as well be applied to the Euler-Lagrange Equations, i.e. in a covariant formalism.

[1] Many other applications can be found in the books of Pommaret [117, 118, 119].

We did not attack the fundamental problem of quantizing such systems. One approach could be based on our novel treatment of initial value problems. Even if many different initial value problems might be possible for a given system, each one will lead to an unconstrained parameterization of the state space. It is, however, yet unclear, how this representation could be used to define canonical coordinates and/or canonical commutation relations. Already at classical level one deals now with functions having domains of definition with different dimensions. Thus one needs a new way to introduce Poisson brackets.

From a physical point of view Chapter 3 could be titled “What is a degree of freedom?”. For finite-dimensional systems, or in the formal language for equations of finite type, the answer is fairly simple. The general solution depends only on a finite number of constants and this number is completely independent of any chosen representation. In some canonical coordinate system one can identify these constants with initial positions and velocities. Thus one would say that halving the number of constants yields the number of degrees of freedom.

In field theories the situation becomes much more complicated. Usually one says that the constraints “determine” some of the fields and halving the number of remaining fields yields the number of degrees of freedom. We have used here quotation marks, because the constraints are typically differential equations and thus can not completely fix any fields. As already discussed in Chapter 4 the effect of the constraints is that the initial data for some fields cannot be prescribed on submanifolds of codimension 1 but only on lower-dimensional submanifolds. Hence there remains some arbitrariness. It is thus unclear what could be a good definition of degrees of freedom for field theories.

In some sense this problem is closely related to the above mentioned problem of quantization. If one could find a reasonable way to introduce canonical variables, then this would probably entail a reasonable definition of degrees of freedom. But it is still doubtful in the light of the here presented results that one will obtain a *single* number. It will probably be some kind of sequence like the Cartan characters.

Independent of the physical implications one could ask what is the best way to compare the arbitrariness of the general solution of two different differential equations. We have seen many different possibilities: Cartan characters, Hilbert polynomial, index of generality plus Cartan genus, strength and the number of arbitrary functions. In some sense they are of course all equivalent, as each can be used to compute the others (or at least the relevant part of them): one always ends up with the index of generality and the Cartan genus. The only exception is the strength; this was already discussed in Section 6.2.

If a differential equation has more independent variables than another one, it also has more Cartan characters. Relevant is, however, only the highest non-vanishing one and its index, i.e. the index of generality and the Cartan genus. Similarly one can compare the Hilbert polynomials. But of importance is then only its degree and leading coefficients, i.e. again more or less the index of generality and the Cartan genus.

The number of arbitrary functions depends on the choice of a representation for the general solution. The only way it can be used reasonably is to rewrite every equation as a first-order one and to compare then the values for an algebraic representation. Relevant is then of course only the number of functions with the maximal number of arguments — which is again the index of generality and the Cartan genus.

All these tools are intrinsically defined, i.e. they are invariant under coordinate transformations and independent of any specific representation chosen for the submanifold $\mathcal{R}_q$. But one can make more drastic transformations like e.g. the reduction of an equation to a first-order one. This means that one represents the differential equation in another jet bundle! The Cartan characters are not necessarily invariant under such operations but the Hilbert polynomial (adjusted for the new order taken as basis), the index of generality and the Cartan genus.

Our analysis of symmetries and reduction methods in Chapter 5 is still far from satisfactory. The only problems that can be considered as solved are the determination of the size of symmetry groups and the loss of generality during a reduction. The former one is a trivial spin-off of Chapter 3; the latter one omits the real problem. Of course one wants to obtain as general a solution as possible, but the really important point is to find a reduction such that one can actually solve the reduced equations.

Although we have proposed a way how formal theory could help here, we have failed to produce a new reduction for some equation of interest like e.g. the Korteweg-de Vries Equation. Our attempts run aground, because the resulting conditions for a suitable differential constraint turned out to be intractable. Probably this approach is feasible only in combination with some other ideas.

The dilaton-gravity studied in Section 7.3 belongs to a fairly interesting class of systems, namely those with a finite-dimensional reduced phase-space. Since these systems occupy a kind of intermediate position between field theory and point mechanics, they provide an ideal test ground to investigate methods for quantization. By choosing a gauge fixing it is always possible to reduce questions of quantum field theory to quantum mechanical ones.

While this is at least sometimes exploited in concrete examples, it has apparently not yet been noticed that these systems could also be quite useful to study problems of complete integrability. So far this term has a precise definition only for finite-dimensional systems. For infinite-dimensional ones many criteria based on many different approaches have been proposed, but no coherent picture has emerged. Systems of finite type could help to clarify this unsatisfactory situation.

As some of the examples in Sections 8.7 and 8.8 showed the use of a computer algebra system is more or less inevitable for geometric computations. No matter whether one wants to complete a differential equation to an involutive one or to compute its symmetry algebra, as soon as one gets beyond second or perhaps third order the computational effort is so high that it takes at least days to produce a result. Even if somebody possesses the necessary stamina, hand calculations contain usually more errors than computer results (if a program is thoroughly tested).

As already mentioned in Section 8.9, $\delta$-regularity provides the biggest problem for the automatic completion to involution. Our introduction of the $k$-tableaux allows surely for a more efficient treatment of it than the brute force approach with a general coordinate system. It is, however, still very time-consuming because of the large dense matrices which must be handled. In hand calculations one can often find ways to avoid applying higher tableaux (see e.g. our treatment of three-dimensional gravity in Section 7.2), but a computer algebra system is probably not able to detect such shortcuts.

A big but often overlooked problem in computer algebra nowadays is the fact that all available systems are closed. Thus much work is duplicated in order to make certain algorithms available in several systems. The time spent for this could be used much better for further research. Especially in symmetry analysis one can see this effect very clearly. The number of available packages is difficult to assess. But many are just translations of others from one computer algebra system into another one. On the whole they all use exactly the same methods and their capabilities hardly differ. The main differences are in the quality of the implementation and thus the performance.

A similar problem occurs, if one wants to treat the output of one algorithm with another one. This can be quite useful as we have seen in the example of the symmetry analysis of the Heat Equation. In fact a determining system in "raw" form is usually not a pretty sight. It contains many dependent or unnecessarily complicated equations. In order to extract information from it one must treat it further. But then one finds the same effect: probably the algorithm one needs is implemented by somebody else in some other computer algebra system!

Some people try to counter this effect by developing interfaces, i.e. programs which transform the output of one package in one computer algebra system into a form which can be read by another package in another system. But this approach is feasible only for a small number of involved packages. Otherwise one ends up with a pile of interfaces. Another problem is that this ansatz requires detailed knowledge of the data structures of each package involved.

Often different packages by different authors written for the same computer algebra system use different data structures. Especially for derivatives, or more generally jet variables, one finds often different approaches. This starts already with the question whether one should use multi-index or repeated index notation. Here systems like AXIOM are advantageous. Due to their modular structure every author can use his favorite notation and still share code with others (at least in theory), if there is an agreement about the needed functionality, i.e. if they specify together a category.

We hope that in the future some other authors will also contribute their methods to the environment we have started in this thesis. In principle this is easily possible, if they can accept the "standards" given by the two implemented categories `JetBundleCategory` and `JetBundleFunctionCategory`. Whether they will like the domain `DifferentialEquation` is another question, because it is really adapted to the requirements of the completion procedure in `CartanKuranishi`. For other applications many of the internally computed and stored data like Jacobi matrices are often not necessary and only slow down the computations. Even our own symmetry package is not based on it!

There exist many possible directions for future extensions of the environment. We mentioned already possible (and probably necessary) improvements of `SymmetryAnalysis`. It can also be generalized to further classes of symmetries like generalized symmetries. Of interest would also be a package for the Painlevé analysis of partial differential equations. Currently there does not seem to exist a package for this in any computer algebra system. Speaking of (completely) integrable systems it could be useful to make a connection between this environment and another program of us for handling pseudo differential operators [139] developed independently of this thesis.

Still very primitive are the provisions for differential geometric calculations. The two existing domains for vector fields and one-forms contain just the necessary operations for symmetry analysis and for the determination of tableaux. One can easily imagine implementing more involved operations in that direction, e.g. for computations with the variational bi-complex [111]. Very useful would be programs for determining the Spencer cohomology or some of the differential sequences frequently applied in formal theory.

The formal theory represents of course not the only way to deal with general systems of partial differential equations. We have already mentioned several times the Cartan-Kähler and Janet-Riquier Theory, respectively. There exists furthermore a dual version of the exterior system approach based on vector fields developed by Vessiot [158]. It is the least known of these theories. The Janet-Riquier Theory can in some sense be considered as superseded by the formal approach which provides an intrinsic treatment. Its specific techniques are, however, quite useful in concrete calculations. Our simplification and reduction routines for general expressions are for instance closely related to Reid's standard form algorithm.

It is probably inevitable to comment on whether or not the formal approach to involution working directly with partial differential equations is "better" than the Cartan-Kähler approach using exterior systems, as this becomes sometimes an almost religious question. The easiest answer is to cite [63] which proves rigorously the equivalence of both approaches. Thus none can claim to be more powerful than the other. Indeed one can often see that preference of one stems from personal taste and not actual advantages. Of course there exist problems which can be handled better with one approach but one cannot give a general judgment.

The exterior approach has surely advantages, if one deals with coordinate free problems. An example of such a problem set up in a frame bundle was given by Hartley *et al.* [68], namely finding a time-like curve in a Minkowski space-time satisfying the Euler-Lagrange equations for a Lagrangian depending on the acceleration and the torsion of the curve. Exterior calculus is also advantageous in maintaining manifest covariance.

On the other hand many problems lead to partial differential equations and not to exterior systems. There are in general many ways to rewrite a partial differential equation as an exterior system. To use the pull-back of the contact system is just one and not necessarily the best. The Korteweg-de Vries Equation is often transformed into an ideal generated by two-forms. Here one must ask why transform at all, if it is possible to do the same calculations directly with the original partial differential equation. The jet bundle formalism also provides an intrinsic differential geometric framework with invariant results.

From a computational point of view using partial differential equations has one advantage: the transformation into an exterior system leads usually to considerably more equations (consider e.g. the case of a single, first-order equation). Since both approaches face the same fundamental problem of combinatorial explosion as soon as prolongations are necessary, this effect is not to be underestimated even when using computer algebra.

# 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 $\Lambda T^*M$ 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 $\omega$ 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* $\Omega$ 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 $\Omega$ does not contain 0-forms. Otherwise, we take as base manifold the submanifold defined by them. Restricting all forms in $\Omega$ 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 $\Omega$, 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 $\Omega$ 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 $\Omega$ 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 $\Omega$, i.e. $\mathrm{d}I(\Omega) \subset I(\Omega)$. $\alpha_k, \beta_k$ 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 $\Omega$, 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 $\Omega$ 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 $\omega$ is independent of $x^k$ if and only if $i_k\omega = 0$ (i.e. $\omega$ 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 $\Omega$ assuming as always that $\Omega$ 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 $\omega_j$ 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 $\omega_j$ 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 $b_i^\ell$ 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 $\beta_j^i$ 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 $\Omega$, if*

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

*$v$ is then called an* isovector *of $\Omega$.*

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 $\Omega$, 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 $\Omega$, 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 $\Omega$ 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 $\Omega$ 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 $\mathcal{P} = \{\omega_1, \ldots, \omega_r\}$, 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 $k = 1, \ldots, r$, 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 $z^1, \ldots, z^N$ 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 $\frac{\partial}{\partial z^{r+1}}, \ldots, \frac{\partial}{\partial z^N}$ 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 $Y_{r+1}, \ldots, Y_N$ form a basis of $\mathcal{P}^\perp$. Then there exists locally a coordinate system $y^1, \ldots, y^N$ on $M$ such that $\frac{\partial}{\partial y^1}, \ldots, \frac{\partial}{\partial y^r}, Y_{r+1}, \ldots, Y_N$ 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

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

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 $\mathrm{d}z^1, \dots, \mathrm{d}z^r$ 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 $\Omega$ 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 $\Omega$. A set of vector fields $\{Z_i\}$, $i = 1, \ldots, g$ such that $E_k$ is generated by $Z_1, \ldots, Z_k$ 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 $\Omega$ and $z_0 \in \mathcal{S}_k$. Assume $\{E_i\}_{i=1,\ldots,k+1}$ 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 $(x^1, \ldots, x^{k+1}, z^1, \ldots, z^{N-k-1})$ such that $\mathcal{S}_k$ is given by the equations $x^{k+1} = 0$, $z^j = \phi^j(x^1, \ldots, x^k)$ 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 $z^j = \Phi^j(x^1, \ldots, x^{k+1})$. 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

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

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 $Z_1, \ldots, Z_k, Z$ 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

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

If we set $\Phi^j = F^j$ for $j > r_k$ where $F^j(x^1, \ldots, x^{k+1})$ 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

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

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 $\Omega$ we must estimate the number of parameters a regular sequence depends on. If $Z_1, \ldots, Z_k$ 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 $x^1, \ldots, x^k$. 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 $\Omega$ 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 $\chi_1, \dots, \chi_n$ 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 $i = 1, \dots, n$. 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 $z^1, \dots, z^n$ as "independent variables", for $\mathcal{S}$ can be described by equations of the form $z^i = f^i(z^1, \dots, z^n)$ for $i = n+1, \dots, N$. 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 $\chi_i$, if there is no $v \in E$ such that

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

We call the exterior system $\Omega$ 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

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

(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

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

where the coefficients $A_k^i$ 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 $\Omega$ 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 $A_k^i$ 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

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

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

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

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

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

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 $p^\alpha_\mu$ 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 $p^\alpha_\nu$ 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 $p^\alpha_\mu$, i.e. belonging to the same dependent variable $u^\alpha$. We introduce furthermore the set $\bar{\mathcal{L}}^\alpha_>$ containing all derivatives $p^\alpha_\mu$ 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 $\delta$-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 $p^\alpha_\mu$ 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 $p^\alpha_\nu$ with $\nu_j = \mu_j$ for $j > i$ would have to be identical with $p^\alpha_\mu$. 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 $\nu$ 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 $\delta$-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 $\delta$-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 $\delta$-regularity makes its hidden appearance even in this algebraic approach to differential equations. We conjecture that if a system is

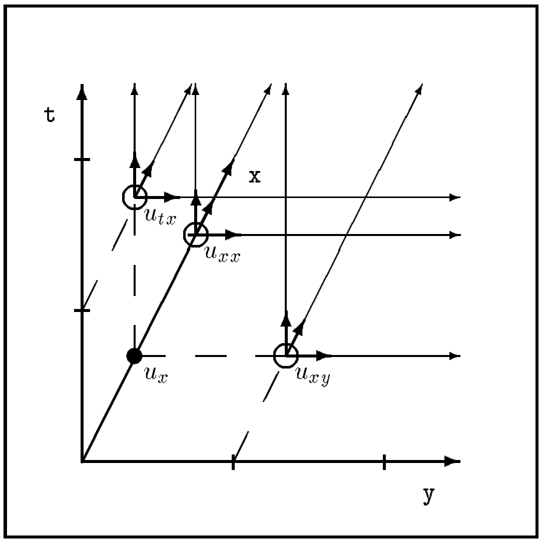


Figure B.2: Reid diagram for equations (B.1,B.2).

passive with respect to a ranking that corresponds to a coordinate system that is not $\delta$-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 $\delta$-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 $0 \leq |\mu| \leq q$. 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

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

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 $\iota_{q+1}$ 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

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

$\tilde{\Phi}^\tau$ is not uniquely defined, as there are in general several possibilities to express a derivative $p^\alpha_\mu$ 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 $\delta$-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 $\sigma$ 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 $\gamma_l$. 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:<br>0 $\rightarrow$ no output<br>1 $\rightarrow$ result is displayed<br>2 $\rightarrow$ Cartan characters are displayed<br>3 $\rightarrow$ integrability conditions are traced<br>4 $\rightarrow$ intermediate dimensions are traced<br>5 $\rightarrow$ intermediate systems are traced<br>6 $\rightarrow$ intermediate symbols are traced<br>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`.

# Bibliography


[1] M. Abramowitz and I. Stegun, editors. *Handbook of Mathematical Functions.* Dover, New York, 1965.

[2] A.M. Ampère. Considérations Générales sur les Intégrales des Équations aux Différentielles Partielles. *J. École Polytéchnique*, 10:549–611, 1815.

[3] I.M. Anderson and J. Pohjanpelto. Variational principles for differential equations with symmetries and conservation laws I. $2^{nd}$ order scalar equations. *Math. Ann.*, 299:191–222, 1994.

[4] D.J. Arrigo, P. Broadbridge, and J.M. Hill. Nonclassical symmetry solutions and the methods of Bluman-Cole and Clarkson-Kruskal. *J. Math. Phys.*, 34:4692–4703, 1993.

[5] T. Banks and M. O'Loughlin. Two-dimensional quantum gravity in Minkowski space. *Nucl. Phys. B*, 362:649–664, 1991.

[6] J. Barcelos-Neto and N.R.F. Braga. Symplectic analysis of a Dirac constrained theory. *J. Math. Phys.*, 35:3497–3503, 1994.

[7] J. Barcelos-Neto and C. Wotzasek. Faddeev-Jackiw quantization and constraints. *Int. J. Mod. Phys. A*, 7:4981–5003, 1992.

[8] J. Barcelos-Neto and C. Wotzasek. Symplectic quantization of constrained systems. *Mod. Phys. Lett. A*, 7:1737–1747, 1992.

[9] Th. Becker and V. Weispfenning. *Gröbner Bases.* Springer-Verlag, New York, 1993.

[10] I. Berchtold. Sparse matrix determinants over integral domains. Master's thesis, ETH Zürich, 1993.

[11] I.N. Bernstein and B.I. Rosenfeld. Homogeneous spaces of infinite Lie algebras and characteristic classes of foliations. *Uspekhi Mat. Nauk*, 28:103–138, 1973.

[12] M. Bernstein and J. Sonnenschein. Comment on the quantization of chiral bosons. *Phys. Rev. Lett.*, 60:1772, 1988.

[13] G.W. Bluman. Simplifying the form of Lie groups admitted by a given differential equation. *J. Math. Anal. Appl.*, 145:52–62, 1990.

[14] G.W. Bluman and J.D. Cole. The general similarity solution of the heat equation. *J. Math. Mech.*, 18:1025–1042, 1969.

[15] G.W. Bluman and S. Kumei. *Symmetries and Differential Equations.* Applied Mathematical Sciences 81. Springer-Verlag, New York, 1989.

[16] F. Brandt. Bäcklund transformations and zero-curvature representations of systems of partial differential equations. *J. Math. Phys.*, 35:2463–2484, 1994.

[17] R.L. Bryant, S.S. Chern, R.B. Gardner, H.L. Goldschmidt, and P.A. Griffiths. *Exterior Differential Systems.* Mathematical Sciences Research Institute Publications 18. Springer-Verlag, New York, 1991.

[18] B. Buchberger and R. Loos. Algebraic simplification. In B. Buchberger, G.E. Collins, and R. Loos, editors, *Computer Algebra — Symbolic and Algebraic Computation.* Springer-Verlag, Berlin, 1982.

[19] W.H. Burge and S.M. Watt. Infinite structures in Scratchpad II. In J.H. Davenport, editor, *Proc. EUROCAL '87*, Lecture Notes in Computer Science 378, pages 138–148. Springer-Verlag, Berlin 1987.

[20] R. Burman. On the strengths of field equations. *Czech. J. Phys. B*, 27:113–116, 1977.

[21] C.G. Callan, S.B. Giddings, J.A. Harvey, and A. Strominger. Evanescent black holes. *Phys. Rev. D*, 45:R1005–R1009, 1992.

[22] C. Carathéodory. *Variationsrechnung und Partielle Differentialgleichungen erster Ordnung, Band I.* Teubner, Leipzig, 1956.

[23] J. Carminati, J.S. Devitt, and G.J. Fee. Isogroups of differential equations using algebraic computing. *J. Symb. Comp.*, 14:103–120, 1992.

[24] G. Carrà Ferro. Gröbner bases and differential algebra. In L. Huguet and A. Poli, editors, *AAECC-5*, Lecture Notes in Computer Science 350, pages 129–140, Berlin, 1987. Springer-Verlag.

[25] G. Carrà Ferro and S.V. Duzhin. Differential-algebraic and differential-geometric approach to the study of involutive symbols. Preprint, University of Catania, Italy, 1992.

[26] E. Cartan. *Les Systèmes Différentielles Extérieurs et leurs Applications Géométriques.* Hermann, Paris, 1945.

[27] E. Cartan. Sur la Théorie des Systèmes en Involution et ses Applications à la Relativité. In *Oeuvres Complètes d'Elie Cartan, Partie* II, *Vol. 2*, pages 1199–1230. Gauthier-Villars, Paris, 1953. (Original: Bull. Soc. Math. France 59 (1931) 88–118).

[28] E. Cartan and A. Einstein. *Lettres sur la Parallélisme Absolue 1929–1932,* edited by R. Debever. Palais des Académies, Bruxelles, 1979.

[29] B. Champagne, W. Hereman, and P. Winternitz. The computer calculation of Lie point symmetries of large systems of differential equations. *Comp. Phys. Comm.*, 66:319–340, 1991.

[30] P.A. Clarkson and M.D. Kruskal. New similarity reductions of the Boussinesq equation. *J. Math. Phys.*, 30:2201–2212, 1989.

[31] A. Clebsch. Über die simultane Integration linearer partieller Differentialgleichungen. *J. Reine Angew. Math.*, 65:257–268, 1866.

[32] M.E.V. Costa and H.O. Girotti. Comment on "Self-dual fields as charge-density solitons". *Phys. Rev. Lett.*, 60:1771, 1988.

[33] R. Courant and D. Hilbert. *Methods of Mathematical Physics*, volume 2. Interscience Publishers, New York, 1962.

[34] D. Cox, J. Little, and D. O'Shea. *Ideals, Varieties, and Algorithms.* Undergraduate Texts in Mathematics. Springer-Verlag, New York, 1992.

[35] C. Cronström and M. Noga. First-order Lagrangians and the Hamiltonian formalism. Preprint TH-7317/94, CERN, 1994.

[36] J.H. Davenport. The AXIOM system. AXIOM Technical Report 3, NAG, 1992.

[37] J.H. Davenport. How does one program in the AXIOM system. AXIOM Technical Report 4, NAG, 1992.

[38] J.H. Davenport and B.M. Trager. Scratchpad's view of algebra I: Basic commutative algebra. In A.M. Miola, editor, *Proc. DISCO '90*, Lecture Notes in Computer Science 429, pages 40–54. Springer-Verlag, Berlin 1990.

[39] T. Dereli and R.W. Tucker. Gravitational interactions in 2+1 dimensions. *Class. Quant. Grav.*, 5:951–959, 1988.

[40] S. Deser, R. Jackiw, and S. Templeton. Three-dimensional massive gauge theories. *Phys. Rev. Lett.*, 48:975–978, 1982.

[41] S. Deser, R. Jackiw, and S. Templeton. Topologically massive gauge theories. *Ann. Phys.*, 140:372–411, 1982.

[42] J. Dieudonné. *Treatise on Analysis*, volume IV. Academic Press, London, 1974.

[43] P.A.M. Dirac. Generalized Hamiltonian dynamics. *Can. J. Math.*, 2:129–148, 1950.

[44] P.A.M. Dirac. Generalized Hamiltonian dynamics. *Proc. Roy. Soc. A*, 246:326–332, 1958.

[45] P.A.M. Dirac. *Lectures on Quantum Mechanics.* Belfer Graduate School Monograph Series 3. Yeshiva University, New York, 1964.

[46] J. Della Dora and E. Tournier. Formal solutions of differential equations in the neighbourhood of singular points. In P.S. Wang, editor, *Proc. SYMSAC*, pages 25–29. ACM, New York 1981.

[47] A. Einstein. *The Meaning of Relativity.* Princeton University Press, Princeton, 5th edition, 1955.

[48] L.P. Eisenhart. *Riemannian Geometry.* Princeton University Press, Princeton, 1926.

[49] P.G. Estévez. The direct method and the singular manifold method for the Fitzhugh-Nagumo equation. *Phys. Lett. A*, 171:259–261, 1992.

[50] P.G. Estévez. Non-classical symmetries and the singular manifold method: the Burgers and the Burgers-Huxley equations. *J. Phys. A*, 27:2113–2127, 1994.

[51] L. Faddeev and R. Jackiw. Hamiltonian reduction of unconstrained and constrained systems. *Phys. Rev. Lett.*, 60:1692–1694, 1988.

[52] N.K. Falck and A.C. Hirshfeld. Dirac-bracket quantization of a constrained nonlinear system: The rigid rotator. *Eur. J. Phys.*, 4:5–9, 1983.

[53] A. Finzi. Sur les Systèmes d'Équations aux Derivées Partielles qui, comme les Systèmes Normaux, Comportent Autant d'Équations que de Fonctions Inconnues. *Proc. Kon. Neder. Akad. v. Wetenschappen*, 50:136–142, 143–150, 288–297, 351–356, 1947.

[54] R. Floreanini and R. Jackiw. Self-dual fields as charge-density solitons. *Phys. Rev. Lett.*, 59:1873–1876, 1987.

[55] V.A. Galaktionov. On new exact blow-up solutions for nonlinear heat conduction with source and application. *Diff. Int. Eqs.*, 3:863–874, 1990.

[56] R.B. Gardner. The Cauchy problem for Pfaffian systems. *Comm. Pure Appl. Math.*, 22:587–596, 1969.

[57] K.O. Geddes, S.R. Czapor, and G. Labahn. *Algorithms for Computer Algebra.* Kluwer Academic Publishers, Dordrecht, 1992.

[58] H.L. Goldschmidt. Integrability criteria for systems of non-linear partial differential equations. *J. Diff. Geom.*, 1:269–307, 1969.

[59] M. Golubitsky and V. Guillemin. *Stable Mappings and Their Singularities.* Graduate Texts in Mathematics 14. Springer-Verlag, New York, 1973.

[60] K. Gottheil. Axioms, categories and domains. *mathPAD*, 4:24–29, 1994.

[61] J. Govaerts. Hamiltonian reduction of first-order action. *Int. J. Mod. Phys. A*, 5:3625–3640, 1990.

[62] D. Gruntz and M.B. Monagan. Introduction to GAUSS. *MapleTech*, 9:23–35, 1993.

[63] V.W. Guillemin and S. Sternberg. An algebraic model of transitive differential geometry. *Bull. Amer. Math. Soc.*, 70:16–47, 1964.

[64] W. Haack and W. Wendland. *Vorlesungen über Partielle und Pfaffsche Differentialgleichungen.* Birkhäuser, Basel, 1969.

[65] A. Hanson, T. Regge, and C. Teitelboim. *Constrained Hamiltonian Systems.* Academia Nazionale dei Lincei, Rome, 1976.

[66] B.K. Harrison and F.B. Estabrook. Geometric approach to invariance groups and solutions of partial differential systems. *J. Math. Phys.*, 12:653–666, 1971.

[67] D. Hartley and R.W. Tucker. A constructive implementation of the Cartan-Kähler theory of exterior differential systems. *J. Symb. Comp.*, 12:655–667, 1991.

[68] D. Hartley, R.W. Tucker, and P.A. Tuckey. Constrained dynamics and exterior differential systems. *J. Phys. A*, 24:5253–5265, 1991.

[69] A.K. Head. Lie, a PC program for Lie analysis of differential equations. *Comp. Phys. Comm.*, 77:241–248, 1993.

[70] W. Hereman. Review of symbolic software for the computation of Lie symmetries of differential equations. *Euromath Bull.*, 2:45–82, 1994.

[71] W. Hereman. Symbolic software for Lie symmetry analysis. In N.H. Ibragimov, editor, *CRC Handbook of Lie Group Analysis of Differential Equations, Volume 3: New Trends in Theoretical Development and Computational Methods*, chapter 13. CRC Press, Boca Raton, Florida, to appear (1995).

[72] R. Hermann. *The Geometry of Non-Linear Differential Equations, Bäcklund Transformations, and Solitons, Part A.* Interdisciplinary Mathematics XII. Math Sci Press, Brookline, 1976.

[73] D. Hilbert. Über die Theorie der algebraischen Formen. *Math. Ann.*, 36:473–534, 1890.

[74] C. Hoenselaers. The strength of a system of differential equations. *Prog. Theor. Phys.*, 58:1185–1190, 1977.

[75] S.A. Hojman and L.C. Shepley. No Lagrangian? No quantization. *J. Math. Phys.*, 32:142–146, 1991.

[76] S.A. Hojman and L.F. Urrutia. On the inverse problem of the calculus of variations. *J. Math. Phys.*, 22:1896–1903, 1981.

[77] E. Holmgren. Über Systeme von linearen, partiellen Differentialgleichungen. *Ofversigt af kongl. Vetenskapsakad. Förh.*, 58:91–103, 1901.

[78] E.L. Ince. *Ordinary Differential Equations.* Dover, New York, 1956.

[79] N. Jacobson. *Lie Algebras.* Dover, New York, 1962.

[80] M. Janet. Sur les Systèmes d'Équations aux Dérivées Partielles. *J. Math. Pure Appl.*, 3:65–151, 1920.

[81] D. Jenks and R.S. Sutor. *AXIOM – The Scientific Computation System.* Springer-Verlag, New York, 1992.

[82] F. John. *Partial Differential Equations.* Applied Mathematical Sciences 1. Springer-Verlag, New York, 1982.

[83] M. Johnston, C. Rogers, W. Schief, and W.M. Seiler. On moving pseudo-spherical surfaces: A Bianchi system and its formal analysis. *Lie Groups Appl.*, to appear, 1994.

[84] E. Kähler. *Einführung in die Theorie der Systeme von Differentialgleichungen.* Teubner, Leipzig, 1934.

[85] I. Kaplansky. *An Introduction to Differential Algebra.* Hermann, Paris, 1957.

[86] E.R. Kolchin. *Differential Algebra and Algebraic Groups.* Academic Press, New York, 1973.

[87] M.V. Korovina. Cauchy problems for overdetermined systems of linear differential equations. *Sov. Math. Dokl.*, 40:469–471, 1990.

[88] M.V. Korovina. Some results related to the Cauchy problem for overdetermined systems of linear differential equations. *Diff. Eq.*, 26:67–76, 1990.

[89] I.S. Krasilshik, V.V. Lychagin, and A.M. Vinogradov. *Geometry of Jet Spaces and Nonlinear Partial Differential Equations.* Gordon&Breach, New York, 1986.

[90] H. Kredel and V. Weispfenning. Computing dimension and independent sets for polynomial ideals. *J. Symb. Comp.*, 6:231–247, 1988.

[91] U. Kulshreshtha. Hamiltonian formulation of a theory with constraints. *J. Math. Phys.*, 33:633, 1992.

[92] M. Kuranishi. On E. Cartan's prolongation theorem of exterior differential systems. *Amer. J. Math.*, 79:1–47, 1957.

[93] M. Kuranishi. *Lectures on Involutive Systems of Partial Differential Equations.* Publicações da Sociedade de Mathemática de São Paulo, São Paolo, 1967.

[94] S. Lang. *Algebra.* Addison-Wesley, Reading, 1984.

[95] D. Levi and P. Winternitz. Non-classical symmetry reduction: Example of the Boussinesq equation. *J. Phys. A: Math. Gen.*, 22:2915–2924, 1989.

[96] H. Lewy. An example of a smooth linear partial differential equation without solution. *Ann. Math.*, 66:155–158, 1957.

[97] S. Lie and F. Engel. *Theorie der Transformationsgruppen*, volume 1. Teubner, Leipzig, 1888.

[98] S. Lie and G. Sheffers. *Vorlesungen über Continuierliche Gruppen.* Teubner, Leipzig, 1893.

[99] D. Louis-Martinez, J. Gegenberg, and G. Kunstatter. Exact Dirac quantization of all 2D dilaton gravity theories. *Phys. Lett. B*, 321:193–198, 1994.

[100] D. Lovelock. Divergence-free third order concomitants of the metric tensor in three dimensions. In H. Rund and W.F. Forbes, editors, *Topics in Differential Geometry.* Academic Press, New York, 1976.

[101] E. Mansfield. *Differential Gröbner Bases.* PhD thesis, Macquarie University, Sydney, 1991.

[102] E. Mansfield and E.D. Fackerell. Differential Gröbner bases. Preprint 92-108, Macquarie University, Sydney, 1992.

[103] K.H. Mariwalla. Applications of the concept of strength of a system of partial differential equations. *J. Math. Phys.*, 15:468–473, 1974.

[104] Y. Matsushima. Sur les algèbres de Lie linéaires semi-involutives. In *Colloque de Topologie de Strasbourg*, page 17. Université de Strasbourg, 1954–55.

[105] N.F.J. Matthews. On the strength of Maxwell's equations. *J. Math. Phys.*, 28:810–814, 1987.

[106] N.F.J. Matthews. The strength of Einstein's equations. *Gen. Rel. Grav.*, 24:17–33, 1992.

[107] A. Mayer. Über unbeschränkt integrable Systeme von linearen totalen Differentialgleichungen. *Math. Ann.*, 5:448–470, 1872.

[108] S.V. Meleshko. Differential constraints and one-parameter Lie-Bäcklund transformation groups. *Sov. Math. Dokl.*, 28:37–41, 1983.

[109] P. Molino. Exterior differential systems and partial differential equations. Lecture notes, Mathematisch Instituut, Universiteit van Amsterdam, 1982.

[110] G.L. Murphy. On the strength of Einstein's unified field equations. *Int. J. Theor. Phys.*, 18:323–327, 1979.

[111] P.J. Olver. *Applications of Lie Groups to Differential Equations.* Graduate Texts in Mathematics 107. Springer-Verlag, New York, 1986.

[112] P.J. Olver. Direct reduction and differential constraints. *Proc. Roy. Soc. Ser. A*, 444:509–523, 1994.

[113] P.J. Olver and P. Rosenau. The construction of special solutions to partial differential equations. *Phys. Lett.*, 114A:107–112, 1986.

[114] P.J. Olver and P. Rosenau. Group-invariant solutions of differential equations. *SIAM J. Appl. Math.*, 47:263–278, 1987.

[115] R. Penney. On the dimensionality of the real world. *J. Math. Phys.*, 6:1607–1611, 1965.

[116] I.G. Petrovsky. *Lectures on Partial Differential Equations.* Interscience Publishers, New York, 1954.

[117] J.F. Pommaret. *Systems of Partial Differential Equations and Lie Pseudogroups.* Gordon&Breach, London, 1978.

[118] J.F. Pommaret. *Differential Galois Theory.* Gordon&Breach, London, 1983.

[119] J.F. Pommaret. *Lie Pseudogroups and Mechanics.* Gordon&Breach, London, 1988.

[120] J.F. Pommaret and A. Haddak. Effective methods for systems of algebraic partial differential equations. In T. Mora and C. Traverso, editors, *Proc. MEGA '90*, pages 411–426. Birkhäuser, Boston 1991.

[121] W.H. Preuss, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C.* Cambridge University Press, Cambridge, 2nd edition, 1992.

[122] D.W. Rand and P. Winternitz. ODEPAINLEVE — a Macsyma package for Painlevé analysis of ordinary differential equations. *Comp. Phys. Comm.*, 42:359–383, 1986.

[123] J. Rauch. *Partial Differential Equations.* Graduate Texts in Mathematics 128. Springer-Verlag, New York, 1991.

[124] G.J. Reid. Algorithms for reducing a system of PDEs to standard form, determining the dimension of its solution space and calculating its Taylor series solution. *Eur. J. Appl. Math.*, 2:293–318, 1991.

[125] G.J. Reid. Finding abstract Lie symmetry algebras of differential equations without integrating determining equations. *Eur. J. Appl. Math.*, 2:319–340, 1991.

[126] G.J. Reid, I.G. Lisle, A. Boulton, and A.D. Wittkopf. Algorithmic determination of commutation relations for Lie symmetry algebras of PDEs. In B.M. Trager and D. Lazard, editors, *Proc. ISSAC '92*, pages 63–68. ACM, New York 1992.

[127] C. Riquier. *Les Systèmes d'Équations aux Derivées Partielles.* Gauthier-Villars, Paris, 1910.

[128] J.F. Ritt. *Differential Algebra.* Dover, New York, 1966. (Original: AMS Colloquium Publications, Vol. XXXIII, 1950).

[129] B.D. Saunders. An implementation of Kovačic's algorithm for solving second order linear homogeneous differential equations. In P.S. Wang, editor, *Proc. SYMSAC*, pages 105–108. ACM, New York 1981.

[130] D.J. Saunders. *The Geometry of Jet Bundles.* London Mathematical Society Lecture Notes Series 142. Cambridge University Press, Cambridge, 1989.

[131] J. Schü. Implementierung des Cartan-Kuranishi-Theorems in AXIOM. Master's thesis, Institut für Algorithmen und Kognitive Systeme, Universität Karlsruhe, 1992.

[132] J. Schü, W.M. Seiler, and J. Calmet. Algorithmic methods for Lie pseudogroups. In N. Ibragimov, M. Torrisi, and A. Valenti, editors, *Proc. Modern Group Analysis: Advanced Analytical and Computational Methods in Mathematical Physics*, pages 337–344, Acireale (Italy), 1992. Kluwer, Dordrecht 1993.

[133] B.F. Schutz. On the strength of a system of partial differential equations. *J. Math. Phys.*, 16:855–856, 1975.

[134] F. Schwarz. Automatically determining symmetries of partial differential equations. *Comp.*, 34:91–106, 1985.

[135] F. Schwarz. Symmetries and involution systems: Some experiments in computer algebra. In M. Ablowitz, B. Fuchssteiner, and M. Kruskal, editors, *Nonlinear Evolution Equations, Solitons and the Inverse Scattering Transform*, Oberwolfach, 1986. World Science Press, Singapore 1987.

[136] F. Schwarz. Programming with abstract data types: The symmetry package SPDE in Scratchpad. In R. Janßen, editor, *Trends in Computer Algebra*, Lecture Notes in Computer Science 296. Springer-Verlag, Heidelberg, 1987.

[137] W.M. Seiler. Arbitrariness of the general solution and symmetries. *Acta Appl. Math.*, to appear, 1994. (Special Issue Proc. Algebraic and Geometric Structures in Differential Equations, Twente 1993, J. Krasilshik and P.H.M. Kersten, eds.).

[138] W.M. Seiler. On the arbitrariness of the general solution of an involutive partial differential equation. *J. Math. Phys.*, 35:486–498, 1994.

[139] W.M. Seiler. Pseudo differential operators and integrable systems in AXIOM. *Comp. Phys. Comm.*, 79:329–340, 1994.

[140] W.M. Seiler, P.J. Vassilou, and C. Rogers. Formal analysis of the Cauchy problem for a system associated with the (2+1)-dimensional Krichever-Novikov equation. Submitted.

[141] J.A. Shih. *Sur la Saturation, la Stabilité des Systèmes d'Équations aux Derivées Partielles et le Calcul Formel.* PhD thesis, Université Denis Diderot, Paris VII, 1994.

[142] M.F. Singer. Liouvillian solutions of $n$-th order homogeneous linear differential equations. *Am. J. Math.*, 103:661–682, 1981.

[143] M.F. Singer. Formal solutions of differential equations. *J. Symb. Comp.*, 10:59–94, 1990.

[144] W.Y. Sit. An algorithm for solving parametric linear systems. *J. Symb. Comp.*, 13:353–394, 1992.

[145] H. Stephani. *Allgemeine Relativitätstheorie.* Deutscher Verlag der Wissenschaften, Berlin, 1980.

[146] H. Stephani. *Differential Equations: Their Solution Using Symmetries.* Cambridge University Press, Cambridge, 1989.

[147] M. Sué. Involutive systems of differential equations: Einstein's strength versus Cartan's degré d'arbitraire. *J. Math. Phys.*, 32:392–399, 1991.

[148] K. Sundermeyer. *Constrained Dynamics.* Lecture Notes in Physics 169. Springer-Verlag, New York, 1982.

[149] R.S. Sutor and R.D. Jenks. The type inference and coercion facilities in the Scratchpad II interpreter. *SIGPLAN Notices*, 22:7–14, 1987.

[150] J.W. Sweeny. The D-Neumann problem. *Acta Math.*, 120:223–251, 1968.

[151] J.M. Thomas. Riquier's existence theorems. *Ann. Math.*, 30:285–321, 1929.

[152] J.M. Thomas. Riquier's existence theorems. *Ann. Math.*, 35:306–311, 1934.

[153] J.M. Thomas. *Systems and Roots.* W. Byrd Press, 1962.

[154] Ph. Tombal and A. Moussiaux. MACSYMA computation of the Dirac-Bergmann algorithm for Hamiltonian systems with constraints. *J. Symb. Comp.*, 1:419–421, 1985.

[155] V.L. Topunov. Reducing systems of linear differential equations to a passive form. *Acta Appl. Math.*, 16:191–206, 1989.

[156] A. Tresse. Sur les invariants différentiels des groupes de transformations. *Acta Math.*, 18:1–88, 1894.

[157] T. Tsujishita. Formal geometry of systems of differential equations. *Sugaku Exp.*, 3:25–73, 1990.

[158] E. Vessiot. Sur une Théorie Nouvelle des Problèmes Généraux d'Integration. *Bull. Soc. Math. Fr.*, 52:336–395, 1924.

[159] E. von Weber. Partielle Differentialgleichungen. In *Enzyklopädie der mathematischen Wissenschaften, Vol. II, Part 1.1*, chapter A5, pages 294–399. 1900.

[160] F. Warner. *Foundations of Differentiable Manifolds and Lie Groups.* Scott, Foreman, Glenview, London, 1971.

[161] S.M. Watt, R.D. Jenks, R.S. Sutor, and B.M. Trager. The Scratchpad II type system: Domains and subdomains. In A.M. Miola, editor, *Computing Tools for Scientific Problem Solving.* Academic Press, New York, 1990.

[162] J. Weiss. The Painlevé property for partial differential equations II: Bäcklund transformations, Lax pairs, and the Schwarzian derivative. *J. Math. Phys.*, 24:1405–1413, 1983.

[163] J. Weiss, M. Tabor, and G. Carnevale. The Painlevé property for partial differential equations. *J. Math. Phys.*, 24:522–526, 1983.

[164] E. Witten. String theory and black holes. *Phys. Rev. D*, 44:314–324, 1991.

[165] A.Yu. Zharkov and Yu.A. Blinkov. Involution approach to solving systems of algebraic equations. In G. Jacob, N.E. Oussous, and S. Steinberg, editors, *Proc. Int. IMACS Symp. Symbolic Computation*, pages 11–17, Lille, 1993.

[166] A.Yu. Zharkov and Yu.A. Blinkov. Involutive bases of zero-dimensional ideals. *J. Symb. Comp.*, to appear, 1994.

[167] R. Zippel. *Effective Polynomial Computation.* Kluwer, Dordrecht, 1993.

# Glossary

| | |
|---|---|
| $\left[{k \atop \ell}\right]$ | number of Taylor coefficients of order $\ell$ of a function with $k$ arguments |
| $\alpha_q^{(k)}$ | $k$-th Cartan character of the differential equation $\mathcal{R}_q$ |
| $\bar{\alpha}_q^{(k)}$ | gauge corrected Cartan character |
| $A_\mu^a$ | vector potential of a Yang-Mills theory |
| $\beta_q^{(k)}$ | number of equations of class $k$ in symbol $\mathcal{M}_q$ |
| $\bar{\beta}_q^{(k)}$ | gauge corrected value of $\beta_q^{(k)}$ |
| $C_{bc}^a$ | structure constants of a Lie algebra |
| $C^{\mu\nu}$ | Cotton tensor |
| $C_P$ | characteristic variety at point $P \in \mathcal{R}_q$ |
| $\mathcal{C}^\infty$ | smooth functions |
| $\mathcal{C}^\omega$ | analytic functions |
| $\mathrm{d}$ | exterior derivative |
| $D_i$ | formal derivative with respect to $x^i$ |
| $\mathcal{D}_\mu$ | covariant derivative with respect to $x^\mu$ |
| $\eta_{\mu\nu}$ | metric tensor of Minkowski space |
| $E_k$ | $k$-dimensional integral element |
| $f_{j,k}$ | number of functions of differentiation order $j$ with $k$ arguments in the general solution |
| $F_{\mu\nu}^a$ | field strength of a Yang-Mills theory |
| $\mathcal{F}^\omega$ | local analytic functions on $J_\infty\mathcal{E}$ |
| $\Gamma_{loc}(\mathcal{E})$ | sheaf of local sections of $\mathcal{E}$ |
| $\Gamma_{\mu\nu}^\rho$ | Christoffel symbols |
| $g_{\mu\nu}$ | metric tensor of space-time |
| $G_{\mu\nu}$ | Einstein tensor |
| $H(E_k)$ | polar system of the integral element $E_k$ |
| $H_q(r)$ | Hilbert polynomial of the symbol $\mathcal{M}_q$ |
| $\bar{H}_q(r)$ | gauge corrected Hilbert polynomial |
| $i_v$ | interior derivative |
| $I_q$ | ideal of the analytic functions vanishing on $\mathcal{R}_q$ |
| $j_q(\sigma)$ | $q$-fold prolongation of section $\sigma \in \Gamma_{loc}(\mathcal{E})$ |
| $J_q\mathcal{E}$ | jet bundle of order $q$ |
| $\mathcal{L}_v$ | Lie derivative |
| $\mathcal{L}_>^\alpha$ | leading derivatives with respect to the ranking $>$ of a system of differential equations belonging to the dependent variable $u^\alpha$ |
| $\Lambda^p T^* M$ | $p$-forms on $M$ |
| $\mathcal{M}_q$ | symbol of the differential equation $\mathcal{R}_q$ |
| $N^*\Sigma$ | co-normal variety of submanifold $\Sigma$ |

| | |
|---|---|
| $\hat{q}(m, n, q)$ | upper bound for the number of prolongations needed to render the symbol $\mathcal{M}_q$ involutive |
| $\mathcal{P}^\perp$ | annihilator of $\mathcal{P}$ |
| $\mathcal{P}_q(\sigma)$ | linearization of $\mathcal{R}_q$ at the section $\sigma$ |
| $R_{\mu\nu}$ | Ricci tensor |
| $\mathcal{R}_q$ | differential equation of order $q$ |
| $\mathcal{R}_{q+r}$ | $r$-fold prolongation of $\mathcal{R}_q$ |
| $\mathcal{R}_{q+r}^{(s)}$ | $s$-fold projection of $(r+s)$-fold prolongation of $\mathcal{R}_q$ |
| $s_k^{(n)}(q)$ | modified Stirling number |
| $S_n V$ | $n$-fold symmetric product of vector space $V$ |
| $\mathcal{S}_k$ | $k$-dimensional integral manifold |
| $\sigma_\chi$ | tableau |
| $\sigma_{[\chi^{(k)},\dots,\chi^{(n)}]}$ | $k$-tableau |
| $\mathcal{T}[\chi]$ | matrix of the tableau $\sigma_\chi$ |
| $TM$ | tangent bundle of manifold $M$ |
| $T^*M$ | cotangent bundle of manifold $M$ |
| $V\mathcal{E}$ | vertical bundle of $\mathcal{E}$ |
| $Z_q^{(0)}$ | compatibility coefficient of $\mathcal{R}_q$ |
| $\bar{Z}_q^{(0)}$ | gauge corrected compatibility coefficient |
| $Z_q^{(1)}$ | strength of $\mathcal{R}_q$ |
| $\bar{Z}_q^{(1)}$ | gauge corrected strength |

# Index