6. Completion to Involution

I wish to God these calculations had been executed by steam.

Charles Babbage

We have seen in the previous chapters that involutive equations possess many attractive features. So the question naturally arises what one should do, if one faces a non-involutive equation. The answer is that one should *complete* it to an equivalent involutive one. We will show that this is, in principle, always possible and discuss algorithms to perform this task. However, algorithms in the strict sense of computer science can be obtained only for differential equations with at most polynomial non-linearities.

Even in the cases where a completion algorithm exists, it tends to be rather expensive and tedious. This is not surprising, if one recalls the one-to-one correspondence between linear partial differential equations in one dependent variable with constant coefficients and polynomials (we simply identify the derivative $\partial^{|\mu|}u/\partial x^{\mu}$ with the term x^{μ}). Completing a system of such equations to involution is equivalent to determining a Gröbner basis for the corresponding polynomial ideal. But this operation is known to possess a doubly exponential worst case complexity (see Appendix B.2). As a consequence, computer algebra is an important tool for concrete computations with general systems of differential equations. Most of our calculations were performed with *MuPAD*.[1] This computer algebra system contains implementations of many of the algorithms discussed here [16].

We start with the fairly simple case of ordinary differential equations. Here completion takes the very intuitive geometric form of imposing tangency conditions. Because of its simplicity it has been discovered and rediscovered

[1] See the web site www.mupad.de for further information.

many times in different fields. For partial differential equations, completion becomes considerably more involved. We describe in Section 6.2 a general geometric method as a constructive proof of the Cartan–Kuranishi Theorem.

Rendering the Cartan–Kuranishi completion algorithmic requires some work. We first go back to the algebraic theory introduced in Chapter 2 and develop in Section 6.4 an algorithm for determining involutive bases. Unfortunately, the termination of this algorithm (which is equivalent to the existence of a basis) is a non-trivial question and requires the introduction of some technicalities which are the topic of the preceeding section.

Section 6.5 combines for linear differential equations the geometric and the algebraic approach into a hybrid algorithm providing a concrete realisation of the Cartan–Kuranishi method (and thus yielding intrinsic results) based on the more efficient algebraic algorithms.

Finally, we discuss the application of the completion theory in the numerical analysis of general systems of differential equations. In Section 6.6, we show how some of the index concepts proposed for differential algebraic equations can be intrinsically formulated within the formal theory. This leads to definitions that can be directly applied to partial differential equations, too. Section 6.7 studies the relation between the index of a partial differential equation and the differential algebraic equations obtained from it by a semi-discretisation. Somewhat surprisingly, it turns out that obstructions to involution become integrability conditions so that again involution and not mere formal integrability is relevant.

6.1 Ordinary Differential Equations

We first restrict our attention to *autonomous* ordinary differential equations where the independent variable x does not occur explicitly. This is only for the simplicity of the presentation and to make the theory more intuitive. The general case will be treated at the end of the section.

Autonomous systems are traditionally described within the tangent bundle (see Appendix C.1). So let $\mathcal{U}$ be an m-dimensional manifold. The right hand side of an ordinary differential equations in the solved form

$$\mathbf{u}' = \boldsymbol{\phi}(\mathbf{u}) \tag{6.1}$$

is interpreted as a vector field $X \in \mathfrak{X}(\mathcal{U})$. Systems of the more general form

$$\boldsymbol{\Phi}\left(\mathbf{u}, \mathbf{u}'\right) = 0 \ , \tag{6.2}$$

where the number of equations may differ from $\dim \mathcal{U}$, appear in many applications and are known in the literature under different names. A classical one is *implicit equation*; in numerical analysis the name *differential algebraic equation* has become popular; in engineering applications one often speaks of a *descriptor form* in contrast to the *state space form* (6.1). Geometrically, such a system corresponds (under some regularity assumptions) to a submanifold $\mathcal{S}^{(0)} \subseteq T\mathcal{U}$.

Assuming that $\mathbf{\Phi}$ is a smooth function and that the rank of the Jacobian $\partial\mathbf{\Phi}/\partial\mathbf{u}'$ is constant, we transform (6.2) into *semi-explicit form*: we split the vector $\mathbf{u}$ into two parts $\mathbf{v}$ and $\mathbf{w}$ (in general of different dimensions) such that the derivatives $\mathbf{w}'$ do not appear in (6.2); by the Implicit Function Theorem we can solve for the derivative $\mathbf{v}'$ and obtain

$$\mathbf{v}' = \boldsymbol{\phi}(\mathbf{v}, \mathbf{w}) \, , \tag{6.3a}$$

$$0 = \boldsymbol{\psi}(\mathbf{v}, \mathbf{w}) \, . \tag{6.3b}$$

This also explains the name *differential algebraic equation*; in semi-explicit form such a system comprises a mixture of differential and algebraic equations. The algebraic equations are often called *constraints*. Note that so far we do not assume that we are dealing with an involutive equation. Hence in contrast to Section 5.1, we may *not* call (6.3) Cartan normal form; the necessary conditions (5.2) are generally not satisfied.

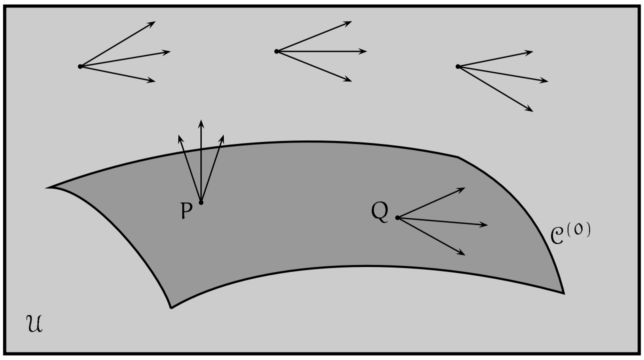

Fig. 6.1. Vector field pencil and submanifold

Thus we are not given a simple vector field on a manifold but a whole *pencil* of vector fields on $\mathcal{U}$ and a *submanifold* $\mathcal{C}^{(0)} = \tau_{\mathcal{U}}(\mathcal{S}^{(0)}) \subseteq \mathcal{U}$, the *constraint manifold* obtained via the tangent bundle projection (see Figure 6.1). In the semi-explicit form, the submanifold is described by the constraint equations (6.3b) and we have a pencil $\mathcal{P}^{(0)}$ of vector fields, as (6.3a) does not uniquely determine all components of a vector field (the $\mathbf{w}$ components remain unknown).

Example 6.1.1. As a trivial example we may consider the following linear system in semi-explicit form defined over $\mathcal{U} = \mathbb{R}^3$ with $\mathbf{u} = (u, v, w)$:

$$v' = u - 2\,, \quad w' = -v\,, \quad v = 3\,. \tag{6.4}$$

Here the constraint manifold $\mathcal{C}^{(0)}$ is the plane $v = 3$ and the pencil $\mathcal{P}^{(0)}$ contains all vector fields $X_a \in \mathfrak{X}(\mathcal{U})$ of the form $X_a(\mathbf{u}) = \bigl(a(\mathbf{u}), u - 2, -v\bigr)^t$ where $a \in \mathcal{F}(\mathcal{U})$ is an arbitrary function. $\triangleleft$

A solution of the differential algebraic equation must define an integral curve to one of the vector fields in the pencil $\mathcal{P}^{(0)}$ and this curve must lie in the constraint manifold $\mathcal{C}^{(0)}$. But this is only possible, if the vector field is tangential to $\mathcal{C}^{(0)}$. As shown in Figure 6.1, two possibilities arise. At some points $P \in \mathcal{C}^{(0)}$ all vectors contained in $\mathcal{P}^{(0)}$ are transversal to $\mathcal{C}^{(0)}$; such points can never lie on a curve defined by a solution of the differential algebraic equation and must be eliminated. This leads to a smaller constraint manifold $\mathcal{C}^{(1)} \subseteq \mathcal{C}^{(0)}$. At the remaining points Q the pencil $\mathcal{P}^{(0)}$ contains at least one vector tangential to $\mathcal{C}^{(0)}$. Here all the transversal vectors must be eliminated of the pencil and we obtain a shrunk pencil $\mathcal{P}^{(1)}$.

If $\mathcal{C}^{(1)}$ is a proper submanifold of $\mathcal{C}^{(0)}$, we must again check the tangency, as vectors tangential to $\mathcal{C}^{(0)}$ are not necessarily tangential to $\mathcal{C}^{(1)}$. Thus we obtain a decreasing sequence of submanifolds $\mathcal{C}^{(0)} \supset \mathcal{C}^{(1)} \supset \cdots \supset \mathcal{C}^{(k)}$ and accompanying vector field pencils $\mathcal{P}^{(0)} \supseteq \mathcal{P}^{(1)} \supseteq \cdots \supseteq \mathcal{P}^{(k)}$. The process stops as soon as $\mathcal{C}^{(i)} = \mathcal{C}^{(i+1)}$; depending on whether or not $\mathcal{P}^{(i)} = \mathcal{P}^{(i+1)}$ we have $k = i$ or $k = i + 1$. At this point we have obtained a submanifold $\mathcal{C}^{(k)} \subseteq \mathcal{U}$ and a pencil $\mathcal{P}^{(k)}$ of vector fields tangential to it.

In a more constructive manner we describe this process by two sequences of submanifolds $\mathcal{C}^{(i)} \subseteq \mathcal{U}$ and $\mathcal{S}^{(i)} \subseteq T\mathcal{U}$. The constraint manifold $\mathcal{C}^{(i)}$ is always obtained via the tangent bundle projection: $\mathcal{C}^{(i)} = \tau_{\mathcal{U}}(\mathcal{S}^{(i)})$. The new differential equation $\mathcal{S}^{(i)}$ arises as the intersection of the old one with the tangent space of its constraint manifold: $\mathcal{S}^{(i+1)} = \mathcal{S}^{(i)} \cap T\mathcal{C}^{(i)}$. The pencils $\mathcal{P}^{(i)}$ are uniquely defined only on the constraint manifolds $\mathcal{C}^{(i)}$; there they simply span the submanifolds $\mathcal{S}^{(i)}$. Off the submanifolds $\mathcal{C}^{(i)}$ the vector fields contained in $\mathcal{P}^{(i)}$ may be continued arbitrarily.

In most applications the final pencil consists of a single vector field (see Figure 6.2); otherwise we are dealing with an underdetermined system. Such a field is called an *underlying vector field.* Note again that it is uniquely determined only *on* the constraint manifold $\mathcal{C}^{(k)}$. If we add to an underlying vector field any field vanishing on $\mathcal{C}^{(k)}$, we obtain another underlying vector field. It is well possible that $\mathcal{C}^{(k)}$ is an empty submanifold. This simply means that the original system was *inconsistent*; it does not possess any solution, as the conditions it poses cannot be fulfilled simultaneously.

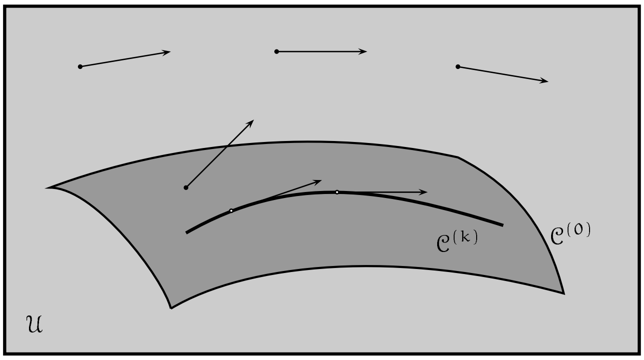

Fig. 6.2. Completed system

Proposition 5.1.1 may now be interpreted as follows: the final constraint manifold $\mathcal{C}^{(k)}$ is an *invariant manifold* for any vector field contained in the final pencil $\mathcal{P}^{(k)}$. Indeed, in dynamical systems theory an invariant manifold for a vector field has the property that any solution for initial data on it stays within the manifold for all times. And this property was our guiding principle in the discussion above.

Example 6.1.2. We continue Example 6.1.1. As the constraint manifold $\mathcal{C}^{(0)}$ is parallel to the u-w plane, a vector field is tangential to it, if and only if its v-component vanishes on $\mathcal{C}^{(0)}$. As the parameter a does not appear in this component, the tangency cannot be achieved by shrinking the pencil. Instead we have to eliminate all points $P \in \mathcal{C}^{(0)}$ which are not simultaneously on the plane $u = 2$. Thus the manifold $\mathcal{C}^{(1)}$ is the straight line $u = 2$, $v = 3$.

As the constraint manifold has become smaller, we have to check the tangency again. Vector fields tangential to $\mathcal{C}^{(1)}$ must have a u-component vanishing on $\mathcal{C}^{(1)}$. This can be achieved by shrinking the pencil to all vector fields X_a where the function a vanishes on $\mathcal{C}^{(1)}$. The constraint manifold remains unchanged, so that the completion stops here.

A simple underlying vector field is $X_1(\mathbf{u}) = (0, 0, -3)^t$. However, the field $X_2(\mathbf{u}) = (2-u, v-3, u-v-2)^t$ is also a valid choice, as X_1 and X_2 coincide on the constraint manifold $\mathcal{C}_1$. $\triangleleft$

The extension of these results to *non-autonomous systems* requires essentially not more than replacing everywhere the tangent by the jet bundle. Thus we assume now that we are given a fibred manifold $\pi : \mathcal{E} \to \mathcal{X}$ over a one-dimensional base space $\mathcal{X}$; for simplicity we take $\mathcal{X} = \mathbb{R}$. Our starting point is a differential equation $\mathcal{R}_1 \subseteq J_1\mathcal{E}$. Instead of the sequence $\mathcal{S}^{(i)}$ we consider the sequence $\mathcal{R}_1^{(i)} \subseteq J_1\mathcal{E}$ constructed by $\mathcal{R}_1^{(i+1)} = \pi_1^2(\mathcal{R}_2^{(i)})$. The completion stops as soon as $\mathcal{R}_1^{(i+1)} = \mathcal{R}_1^{(i)}$.

At first sight, this looks very different compared to the completion process for autonomous systems. The definition of $\mathcal{R}_1^{(i+1)}$ requires as first step a prolongation to second order and then we project back to first order. In the autonomous case, we reversed the order of the operations: first we projected to the constraint manifold, then we "prolonged" by considering its tangent space. But for *ordinary* differential equations we have the following result.

Proposition 6.1.3. *Let $\mathcal{R}_q \subseteq J_q\mathcal{E}$ be an ordinary differential equation. Then $\mathcal{R}_q^{(1)} = \mathcal{R}_q \cap J_1\mathcal{R}_{q-1}^{(1)}$ (here we consider $J_1\mathcal{R}_{q-1}^{(1)}$ as a submanifold of $J_q\mathcal{E}$).*

Proof. For notational simplicity we give the proof for the case of a first-order equation: $q = 1$. The extension to the general case is trivial. Let a local representation of the differential equation $\mathcal{R}_1$ be

$$\mathcal{R}_1 : \begin{cases} \boldsymbol{v}' = \boldsymbol{\phi}(x, \boldsymbol{v}, \boldsymbol{w}) , \\ 0 = \boldsymbol{\psi}(x, \boldsymbol{v}, \boldsymbol{w}) . \end{cases} \tag{6.5}$$

This is the obvious generalisation of the semi-explicit form (6.3) to non-autonomous systems. We need a local description of $\mathcal{R}_1^{(1)}$. Because of the semi-explicit form it is easy to see that the prolongation of the equations containing $\boldsymbol{v}'$ leads to second-order equations and each equation is solved for a different derivative. Thus no cancellations of second-order derivatives are possible and these equations drop out when we project back to first order.

A local representation of $\mathcal{R}_1^{(1)}$ is obtained by adding to the equations already contained in (6.5) the equations

$$\frac{\partial \boldsymbol{\psi}}{\partial x} + \frac{\partial \boldsymbol{\psi}}{\partial \boldsymbol{v}}\boldsymbol{v}' + \frac{\partial \boldsymbol{\psi}}{\partial \boldsymbol{w}}\boldsymbol{w}' = 0 \tag{6.6}$$

obtained by prolonging the algebraic equations in (6.5). Of course, in general this is a redundant description, as these new equations are not necessarily independent of the original ones. But this is of no importance here.

It is trivial to see that the same equations also define a local representation of $\mathcal{R}_1 \cap J_1\mathcal{R}_0^{(1)}$. The constraint manifold $\mathcal{R}_0^{(1)}$ is locally described by the algebraic equations in (6.5) and we must simply add their prolongations (6.6) to the original equations for the intersection. □

Hence we may reverse the order of prolongation and projection for ordinary differential equations. There remains to show that the final differential equation $\mathcal{R}_1^{(k)}$ obtained in our completion process is indeed an involutive equation. Obviously, any smooth solution of the original equation $\mathcal{R}_1$ is also a solution of $\mathcal{R}_1^{(k)}$, as it automatically satisfies all the integrability conditions added during the completion.

Corollary 6.1.4. *The final differential equation $\mathcal{R}_1^{(k)}$ is involutive.*

Proof. As we are dealing with ordinary differential equations, involution is equivalent to formal integrability. By the previous proposition and our construction it is obvious that $\mathcal{R}_1^{(k)} = \mathcal{R}_1^{(k+\ell)}$ for any $\ell > 0$. But this means that it is a formally integrable equation. □

A closer analysis of (6.6) clarifies the two different effects described in the autonomous case: shrinking of either the constraint manifold or the vector field pencil. The derivatives v' in (6.6) can be eliminated using the differential equations in (6.5). Thus the rank of the Jacobian $\partial\psi/\partial w$ determines how many new differential equations, i. e. integrability conditions of order one, appear. Such new differential equations correspond to a shrinking of the pencil. If the Jacobian does not possess maximal rank, we can derive some algebraic equations from (6.6). If these are independent of the constraints in (6.5), we have found further integrability conditions leading to a smaller constraint manifold.

Obviously, the completion process always terminates, as the proof of Proposition 6.1.3 shows that $\dim \mathcal{R}_1^{(i+1)} < \dim \mathcal{R}_1^{(i)}$ until the process stops (we have additional equations). As $J_1\mathcal{E}$ is a finite-dimensional manifold, we cannot have an infinite decreasing sequence. One should note that the assumption of a regular equation is crucial for this proof. It is equivalent to the assumption of constant rank of all appearing Jacobians which is necessary to allow us at each step the distinction into algebraic and differential equations.

6.2 The Cartan–Kuranishi Completion

In the previous section we mainly worked in the tangent bundle, as this makes the completion process for (autonomous) ordinary differential equations very transparent and intuitive. Now we derive in an intrinsic manner a rather general completion method that can – in principle – handle any *regular* system of partial differential equations. However, as we will see, some of the required operations cannot always be performed effectively.

Recall that a differential equation $\mathcal{R}_q$ is regular, if all its prolongations and subsequent projections lead to fibred submanifolds. Furthermore, we assume that the indices of all symbols are constant. Obviously, it is not easy to check in advance whether a given differential equation satisfies this condition. Usually, it can only be verified during the completion. A trivial exception are linear equations.

Theorem 6.2.1 (Cartan–Kuranishi). *Let $\mathcal{R}_q \subseteq J_q\mathcal{E}$ be a regular differential equation. Then two integers $r, s \geq 0$ exist such that $\mathcal{R}_{q+r}^{(s)}$ is involutive.*

Proof. We give a constructive proof leading directly to the promised general completion method. It is based on Algorithm 6.1. If we can show that it always terminates with an involutive equation of the form $\mathcal{R}^{(s)}_{q+r}$, we are done. Thus there are two points to prove: the correctness and the termination.

Algorithm 6.1 Cartan–Kuranishi Completion

Input: regular differential equation $\mathcal{R}_q \subseteq J_q\mathcal{E}$
Output: equivalent involutive equation $\mathcal{R}^{(s)}_{q+r}$

$r \leftarrow 0$; $s \leftarrow 0$
repeat
  while $\mathcal{M}^{(s)}_{q+r}$ not involutive **do**
    $r \leftarrow r + 1$
  end_while
  $intConds \leftarrow \mathcal{R}^{(s+1)}_{q+r} \subsetneq \mathcal{R}^{(s)}_{q+r}$
  if $intConds$ **then**
    $s \leftarrow s + 1$
  end_if
until $\neg intConds$
return $\mathcal{R}^{(s)}_{q+r}$

The *correctness* is simple. Assume that the algorithm terminates. At the end of the inner loop the symbol $\mathcal{M}^{(s)}_{q+r}$ is involutive. If *intConds* is false, then $\mathcal{R}^{(s+1)}_{q+r} = \mathcal{R}^{(s)}_{q+r}$. Thus all conditions of Theorem 3.2.3 are satisfied and the equation $\mathcal{R}^{(s)}_{q+r}$ is involutive.

For the *termination* we must consider the two loops in Algorithm 6.1. In local coordinates at a given point $\rho \in \mathcal{R}^{(s)}_{q+r}$ the inner one corresponds by definition of an involutive symbol to the determination of a Pommaret basis. We will prove later in this chapter that such a basis always exists and thus the loop always terminates provided our coordinates are δ-regular (Theorem 6.4.6 together with the results of Section 2.4).

The termination of the outer loop follows from a Noetherian argument. At the end of the sth iteration of the outer loop we arrive at the equation $\mathcal{R}^{(s)}_{q+r_s}$ where r_s is the total number of iterations through the inner loop performed so far. If $\rho \in \mathcal{R}^{(s)}_{q+r_s}$ is an arbitrary fixed point, then for $0 \leq \bar{s} \leq s$ the points $\rho^{(\bar{s})} = \pi^{q+r_s}_{q+r_{\bar{s}}}(\rho)$ lie in $\mathcal{R}^{(\bar{s})}_{q+r_{\bar{s}}}$. As discussed in Section 3.1, the equations defining the symbol space $(\mathcal{M}^{(\bar{s})}_{q+r_{\bar{s}}})_{\rho^{(\bar{s})}}$ generate the symbol module[2] $\mathcal{M}^{\mathcal{P}}_{\bar{s}}$, a submodule of a free polynomial module over $\mathcal{P} = \Bbbk[x^1, \dots, x^n]$. Obviously, we have $\mathcal{M}^{\mathcal{P}}_0 \subseteq \langle \mathcal{M}^{\mathcal{P}}_0, \mathcal{M}^{\mathcal{P}}_1 \rangle \subseteq \cdots \subseteq \langle \mathcal{M}^{\mathcal{P}}_0, \mathcal{M}^{\mathcal{P}}_1, \dots, \mathcal{M}^{\mathcal{P}}_s \rangle$; in fact, each integrability condition added in the outer loop yields a new module generator.

[2] Here the subscript does not refer to a homogeneous component but labels different modules.

Thus we have constructed an ascending chain of submodules of a free polynomial module. As polynomial modules are Noetherian, this chain must become stationary for some value of s. But this implies that in the iteration of the outer loop corresponding to this value *no* integrability condition was found. Hence the algorithm terminates at this iteration. □

If we apply Algorithm 6.1 to a concrete differential equation, we alternate prolongations and projections, thus, strictly speaking, we work with differential equations $((((\mathcal{R}_{q+r_1})^{(1)})_{+r_2})^{(1)})\cdots)_{+r_s}$. However, we only project equations with an involutive symbol. Thus, by Part (ii) of Proposition 3.1.10, these equations coincide with $\mathcal{R}^{(s)}_{q+r_1+\cdots+r_s}$.

The involutive equation $\mathcal{R}^{(s)}_{q+r}$ is *equivalent* to the original equation $\mathcal{R}_q$ in the sense that both equations possess the same formal solution space. In fact, both equations have even the same smooth solutions. Differences may show up only for solutions of $\mathcal{R}_q$ contained in $C^k(\Omega)$ with $q \leq k < q + r$. They cannot be strong solutions of $\mathcal{R}^{(s)}_{q+r}$, as they are lacking the necessary differentiability. But we may expect that they are at least weak solutions.

Example 6.2.2. We demonstrate the working of Algorithm 6.1 for a classical linear second-order equation due to Janet:

$$\mathcal{R}_2 : \begin{cases} u_{zz} + yu_{xx} = 0 \,, \\ u_{yy} = 0 \,. \end{cases} \tag{6.7}$$

Thus we have one dependent variable and three independent variable. Obviously, the system is overdetermined.

According to our algorithm we prolong until the symbol becomes involutive. One readily checks that the indices of the symbol $\mathcal{M}_2$ are $\beta_2^{(3)} = \beta_2^{(2)} = 1$ and $\beta_2^{(1)} = 0$. The rank of the symbol matrix M_3 is 6 (the non-multiplicative prolongation $u_{yyz} = 0$ is necessary to generate $\mathcal{M}_3$), thus $\mathcal{M}_2$ is not involutive and we must prolong. In a lengthy computation one shows that $\mathcal{M}_3$ is involutive and the inner loop terminates with $r = 1$. Now we must check for integrability conditions. One finds $\dim \mathcal{R}_3^{(1)} = 11 < \dim \mathcal{R}_3 = 12$. Thus $\mathcal{R}_3$ is not involutive, as one condition is hidden. Indeed,

$$\partial_{yy}(u_{zz} + yu_{xx}) - \partial_{zz}u_{yy} - y\partial_{xx}u_{yy} = u_{xxy} = 0 \,. \tag{6.8}$$

Thus at the end of the first iteration of the outer loop $r = s = 1$ and we continue with the analysis of $\mathcal{R}_3^{(1)}$. The calculations are now becoming more and more tedious, as the number of equations needed in local representations of either the differential equations or their symbols rapidly grows. So we can only sketch the progress of the computation.

$\mathcal{M}_3^{(1)}$ is again not involutive, since setting up $\mathcal{M}_4^{(1)}$ requires the non-multiplicative prolongation $u_{xxyz} = 0$. The symbol $\mathcal{M}_4^{(1)}$ is involutive and at the end of the inner loop $r = 2$. But again we find an integrability condition,

$$\partial_{xxy}(u_{zz} + yu_{xx}) - y\partial_{xx}u_{xxy} - \partial_{zz}u_{xxy} = u_{xxxx} = 0\ , \tag{6.9}$$

and $\mathcal{R}_4^{(1)}$ is not involutive. Thus now $s = 2$ and we continue with $\mathcal{R}_4^{(2)}$.

Again, the symbol $\mathcal{M}_4^{(2)}$ is not involutive because of the non-multiplicative prolongation $u_{xxxxz} = 0$. Thus we prolong once more in order to obtain $\mathcal{R}_5^{(2)}$. Finally, the algorithm terminates with $r = 3$ and $s = 2$, as the symbol $\mathcal{M}_5^{(2)}$ is involutive and no integrability conditions appear during the prolongation so that $\mathcal{R}_5^{(2)}$ is involutive. Thus, $\mathcal{R}_4^{(2)}$ is another example of an equation which is formally integrable but *not* involutive. The problem is that we can detect this only in hindsight, after the completion has terminated.

In fact, the involution of $\mathcal{M}_5^{(2)}$ is trivial, as it vanishes. Thus we are dealing here with an equation of finite type. Its solution space is twelve-dimensional as $\dim \mathcal{R}_5^{(2)} = 12$. Indeed, a direct integration of the completed equation yields the solution

$$\begin{aligned} u(x,y,z) =& \frac{1}{3}a_1x^3z - a_2xyz^2 + a_3x^3 + a_2x^2z + a_4xyz - a_1yz^2 + \\ & a_5x^2 + a_6xy + a_7xz + a_8yz + a_9x + a_{10}y + a_{11}z + a_{12}\ . \end{aligned} \tag{6.10}$$

As one can see the straightforward application of Algorithm 6.1 becomes quite tedious even for such small equations. In our particular example one needs a total of 132 prolongations in order to get local representations for all appearing equations. Furthermore, one must compute row echelon forms of symbol matrices and Jacobians; the largest is an 86×84 matrix. We will show later that by a clever organisation of the computations one can completely avoid the determination of row echelon forms and reduce the number of prolongations by about 80%. ◁

Algorithm 6.1 does not really represent an algorithm but only a "method", i. e. it is not clear whether all steps can be performed effectively. This concerns in particular the test for inequality in Line /6/. To decide whether or not a symbol is involutive requires basically only some linear algebra and is thus fully algorithmic. However, even there it may happen for non-linear systems that not all necessary operations with the coefficients of the symbol can be performed effectively.

The test in Line /6/ requires to decide whether certain equations are algebraically independent. Depending on the type of the equation this can be highly non-trivial. For *linear* equations it is again only a matter of some linear algebra. For *polynomial* equations it can – at least in principle – be done using Gröbner bases (see Appendix B.2), although the computations suffer potentially from a very high complexity. For arbitrary non-linear equations no general criterion is known. Depending on the class of functions considered the situation may become even worse. While analytic functions are either independent or not, smooth functions may be independent in some domains and dependent in others.

Finally, we compare Algorithm 6.1 with the geometric completion procedure for ordinary differential equations introduced in the last section. As any ordinary differential equation has an involutive symbol, the inner loop becomes superfluous. The algorithm reduces to a simple *prolong–project*, *prolong–project*, ... But we have seen in the proof of Proposition 6.1.3 that it is possible to swap the operations and use *project–prolong* cycles. The result is the same, but the modified algorithm is computationally much more efficient, as much less equations must be prolonged. But one should stress again that this modification is allowed only for *ordinary* differential equations, as only there one may omit the analysis of the symbol.

6.3 Continuous and Constructive Divisions

In Chapter 2 we introduced involutive bases. But we postponed the question how they can actually be constructed. Unfortunately, for arbitrary involutive division no satisfying solution is known so far. One can follow a brute force approach, namely performing a breadth first search through the tree of all possible completions. Obviously, it terminates only, if a finite basis exists. But for divisions satisfying some additional properties one can design a fairly efficient completion algorithm.

The first problem in constructing an involutive completion of a finite subset $\mathcal{N} \subset \mathbb{N}_0^n$ for a division L is to check whether $\mathcal{N}$ is already involutive. The trouble is that we do not know a priori where obstructions to involution might lie. As these multi indices must somehow be related to the non-multiplicative indices of the elements of $\mathcal{N}$, the multi indices $\nu + 1_j$ with $\nu \in \mathcal{N}$ and $j \in \bar{N}_{L,\mathcal{N}}(\nu)$ are a natural first guess.

Definition 6.3.1. *The finite set $\mathcal{N} \subset \mathbb{N}_0^n$ is* locally involutive *for the involutive division L, if $\nu + 1_j \in \langle\mathcal{N}\rangle_L$ for every non-multiplicative index $j \in \bar{N}_{L,\mathcal{N}}(\nu)$ of every multi index $\nu \in \mathcal{N}$.*

While involution obviously implies local involution, the converse does not hold, as the following rather bizarre example of a globally defined division due to Gerdt and Blinkov [56] demonstrates.

Example 6.3.2. We consider $\mathbb{N}^3$. The involutive division L is defined by the following assignment of multiplicative indices. With the exception of four multi indices all elements of $\mathbb{N}^3$ do not have any multiplicative indices. These four exceptions are: $N_L([0,0,0]) = \{1,2,3\}$, $N_L([1,0,0]) = \{1,3\}$, $N_L([0,1,0]) = \{1,2\}$, and $N_L([0,0,1]) = \{2,3\}$. It is straightforward to show that this indeed defines an involutive division.

Now we take the set $\mathcal{N} = \{[1,0,0],[0,1,0],[0,0,1]\}$. It is not involutive, as the multi index $[1,1,1] \in \langle\mathcal{N}\rangle$ is not contained in the involutive span $\langle\mathcal{N}\rangle_L$. But it is locally involutive, as the three multi indices $[1,1,0]$, $[0,1,1]$ and $[1,0,1]$ obtained by taking the elements of $\mathcal{N}$ and adding their respective non-multiplicative index are contained in $\langle\mathcal{N}\rangle_L$. $\triangleleft$

Definition 6.3.3. *Let* L *be an involutive division and* $\mathcal{N} \subset \mathbb{N}_0^n$ *a finite set. Let furthermore* $(\nu^{(1)},\dots,\nu^{(t)})$ *be a finite sequence of elements of* $\mathcal{N}$ *where every multi index* $\nu^{(k)}$ *with* $k < t$ *has a non-multiplicative index* $j_k \in \bar{N}_{L,\mathcal{N}}(\nu^{(k)})$ *such that* $\nu^{(k+1)} |_{L,\mathcal{N}} \nu^{(k)} + 1_{j_k}$. *The division* L *is called* continuous, *if any such sequence consists only of distinct elements, i. e. if* $\nu^{(k)} \neq \nu^{(\ell)}$ *for all* $k \neq \ell$.

It is easy to see that the division of Example 6.3.2 is not continuous. And indeed for this special class of divisions local involution always implies involution. This makes these divisions very convenient for concrete computations: if obstructions to involution exist, some of them must be of the form $\nu + 1_j$ with $\nu \in \mathcal{N}$ and index $j \in \bar{N}_{L,\mathcal{N}}(\nu)$ and all others are “multiples” of these.

Proposition 6.3.4. *For a continuous division* L, *any locally involutive set* $\mathcal{N} \subset \mathbb{N}_0^n$ *is involutive.*

Proof. Let the set Σ contain those obstructions to involution that are of minimal length. We claim that for a continuous division L all multi indices $\sigma \in \Sigma$ are of the form $\nu + 1_j$ with $\nu \in \mathcal{N}$ and $j \in \bar{N}_{L,\mathcal{N}}(\nu)$. This immediately implies our proposition: since for a locally involutive set all such multi indices are contained in $\langle\mathcal{N}\rangle_L$, we must have $\Sigma = \emptyset$ and thus $\langle\mathcal{N}\rangle = \langle\mathcal{N}\rangle_L$.

In order to prove our claim, we choose a $\sigma \in \Sigma$ for which no $\nu \in \mathcal{N}$ exists with $\sigma = \nu + 1_j$. We collect in $\mathcal{N}_\sigma$ all divisors $\nu \in \mathcal{N}$ of σ of maximal length. Let $\nu^{(1)}$ be an element of $\mathcal{N}_\sigma$; by assumption the multi index $\mu^{(1)} = \sigma - \nu^{(1)}$ satisfies $|\mu^{(1)}| > 1$ and at least one non-multiplicative index $j_i \in \bar{N}_{L,\mathcal{N}}(\nu^{(1)})$ exists with $\mu^{(1)}_{j_1} > 0$. By the definition of Σ we have $\nu^{(1)} + 1_{j_1} \in \langle\mathcal{N}\rangle_L$. Thus a multi index $\nu^{(2)} \in \mathcal{N}$ exists with $\nu^{(2)} |_{L,\mathcal{N}} \nu^{(1)} + 1_{j_1}$. This implies $\nu^{(2)} \mid \sigma$ and we set $\mu^{(2)} = \sigma - \nu^{(2)}$. By the definition of the set $\mathcal{N}_\sigma$ we have $|\nu^{(2)}| \leq |\nu^{(1)}|$. Hence $\nu^{(2)} + 1_j \in \langle\mathcal{N}\rangle_L$ for all j.

Choose a non-multiplicative index $j_2 \in \bar{N}_{L,\mathcal{N}}(\nu^{(2)})$ with $\mu^{(2)}_{j_2} > 0$. Such an index exists as otherwise $\sigma \in \langle\mathcal{N}\rangle_L$. By the same arguments as above, a multi index $\nu^{(3)} \in \mathcal{N}$ exists with $\nu^{(3)} \mid_{L,\mathcal{N}} \nu^{(2)} + 1_{j_2}$ and $|\nu^{(3)}| \leq |\nu^{(2)}|$. We can iterate this process and produce an infinite sequence $(\nu^{(1)}, \nu^{(2)}, \dots)$ where each multi index $\nu^{(i)} \in \mathcal{N}$ and $\nu^{(i+1)} \mid_{L,\mathcal{N}} \nu^{(i)} + 1_{j_i}$ with $j_i \in \bar{N}_{L,\mathcal{N}}(\nu^{(i)})$. As $\mathcal{N}$ is a finite set, the elements of the sequence cannot be all different. This contradicts our assumption that L is a continuous division: by taking a sufficiently large part of this sequence we obtain a finite sequence with all properties mentioned in Definition 6.3.3 but containing some identical elements. Hence a multi index $\nu \in \mathcal{N}$ must exist such that $\sigma = \nu + 1_j$. □

Continuity is not only sufficient for the equivalence of local involution and involution but also useful for proving other properties. The main point is that it excludes the existence of infinite reduction chains. The two involutive divisions introduced in Examples 2.1.2 and 2.1.5 are both continuous.

Lemma 6.3.5. *The Janet and the Pommaret division are continuous.*

Proof. Let $\mathcal{N} \subseteq \mathbb{N}_0^n$ be a finite set and $(\nu^{(i)}, \dots, \nu^{(t)})$ a finite sequence where $\nu^{(i+1)} \mid_{L,\mathcal{N}} \nu^{(i)} + 1_j$ with $j \in \bar{N}_{L,\mathcal{N}}(\nu^{(i)})$ for $1 \leq i < t$.

We claim that for $L = J$, the Janet division, $\nu^{(i+1)} \succ_{\text{lex}} \nu^{(i)}$ implying that the sequence is strictly ascending and cannot contain any identical entries. Set $k = \max\{i \mid \mu_i \neq \nu_i\}$. Then $j \leq k$, as otherwise $j \in N_{J,\mathcal{N}}(\nu^{(i+1)})$ entails $j \in N_{J,\mathcal{N}}(\nu^{(i)})$ contradicting our assumption that j is non-multiplicative for the multi index $\nu^{(i)}$. But $j < k$ is also not possible, as then $\nu_k^{(i+1)} < \nu_k^{(i)}$ and so k cannot be multiplicative for $\nu^{(i+1)}$. There remains as only possibility $j = k$. In this case we must have $\nu_j^{(i+1)} = \nu_j^{(i)} + 1$ (otherwise j could not be multiplicative for $\nu^{(i+1)}$). Thus we conclude that $\nu^{(i+1)} \succ_{\text{lex}} \nu^{(i)}$ and the Janet division is continuous.

The proof for the case $L = P$, the Pommaret division, is slightly more subtle.[3] The condition $j \in \bar{N}_P(\nu^{(i)})$ implies that $\operatorname{cls}(\nu^{(i)} + 1_j) = \operatorname{cls}\nu^{(i)}$ and if $\nu^{(i+1)} \mid_P \nu^{(i)} + 1_j$, then $\operatorname{cls}\nu^{(i+1)} \geq \operatorname{cls}\nu^{(i)}$, i. e. the class of the elements of the sequence is monotonously increasing. If $\operatorname{cls}\nu^{(i+1)} = \operatorname{cls}\nu^{(i)} = k$, then the involutive divisibility requires that $\nu_k^{(i+1)} \leq \nu_k^{(i)}$, i. e. among the elements of the sequence of the same class the corresponding entry is monotonously decreasing. And if finally $\nu_k^{(i+1)} = \nu_k^{(i)}$, then we must have $\nu^{(i+1)} = \nu^{(i)} + 1_j$, i. e. the length of the elements is strictly increasing. Hence all elements of the sequence are different and the Pommaret division is continuous. □

[3] It is tempting to tackle the lemma for the Pommaret division in the same manner as for the Janet division using $\prec_{\text{revlex}}$ instead of $\prec_{\text{lex}}$; in fact, such a "proof" is contained in [56]. Unfortunately, it is not correct, as $\prec_{\text{revlex}}$ is not a term order (see Example B.2.1): if $\nu^{(i+1)} = \nu^{(i)} + 1_j$, then $\nu^{(i+1)} \prec_{\text{revlex}} \nu^{(i)}$ although the latter multi index is a divisor of the former one! Thus the sequences considered in Definition 6.3.3 are in general not strictly ascending with respect to $\prec_{\text{revlex}}$.

Definition 6.3.6. *Let $\mathcal{N} \subset \mathbb{N}_0^n$ be a finite set of multi indices and choose a multi index $\nu \in \mathcal{N}$ and a non-multiplicative index $j \in \bar{N}_{L,\mathcal{N}}(\nu)$ such that*

(i) $\nu + 1_j \notin \langle \mathcal{N} \rangle_L$.
(ii) *If there exists $\mu \in \mathcal{N}$ and $k \in \bar{N}_{L,\mathcal{N}}(\nu)$ such that $\mu + 1_k \mid \nu + 1_j$ but $\mu + 1_k \neq \nu + 1_j$, then $\mu + 1_k \in \langle \mathcal{N} \rangle_L$.*

The division L is constructive, *if for any such set $\mathcal{N}$ and any such multi index $\nu + 1_j$ no multi index $\rho \in \langle \mathcal{N} \rangle_L$ with $\nu + 1_j \in \mathcal{C}_{L,\mathcal{N}\cup\{\rho\}}(\rho)$ exists.*

In words, constructivity can roughly be explained as follows. The conditions imposed on ν and j ensure a kind of minimality: no proper divisor of $\nu + 1_j$ is of the form $\mu + 1_k$ for a $\mu \in \mathcal{N}$ and not contained in the involutive span $\langle \mathcal{N} \rangle_L$. The conclusion implies that it is useless to add multi indices to $\mathcal{N}$ that lie in some involutive cone, as none of them can be an involutive divisor of $\nu + 1_j$. An efficient completion algorithm for a constructive division should consider only non-multiplicative indices. Because of the restriction to multi indices of the form $\nu + 1_j$, constructivity is really important only for divisions which are also continuous.

Lemma 6.3.7. *Any globally defined division (and thus the Pommaret division) is constructive. The Janet division is constructive, too.*

Proof. For a globally defined division the proof is very simple. For any multi index $\rho \in \langle \mathcal{N} \rangle_L$ there exists a multi index $\mu \in \mathcal{N}$ such that $\rho \in \mathcal{C}_L(\mu)$. As for a globally defined division the multiplicative indices are independent of the reference set, we must have by the definition of an involutive division that $\mathcal{C}_L(\rho) \subseteq \mathcal{C}_L(\mu)$. Hence adding such a multi index to $\mathcal{N}$ cannot change the involutive span and if $\nu + 1_j \notin \langle \mathcal{N} \rangle_L$, then also $\nu + 1_j \notin \langle \mathcal{N} \cup \{\rho\} \rangle_L$. But this implies the constructivity of L.

The proof of the constructivity of the Janet division is more involved. The basic idea is to show that if it was not constructive, it also could not be continuous. Let $\mathcal{N}$, ν, j be as described in Definition 6.3.6. Assume for a contradiction that a multi index $\rho \in \langle \mathcal{N} \rangle_J$ exists with $\nu + 1_j \in \mathcal{C}_{J,\mathcal{N}\cup\{\rho\}}(\rho)$. We write $\rho = \nu^{(1)} + \mu$ for a multi index $\nu^{(1)} \in \mathcal{N}$ with $\rho \in \mathcal{C}_{J,\mathcal{N}}(\nu^{(1)})$. As $\nu + 1_j \notin \langle \mathcal{N} \rangle_J$, we must have $|\mu| > 0$. Set $\lambda = \nu + 1_j - \rho$ and let m, l be the maximal indices such that $\mu_m > 0$ and $\lambda_l > 0$, respectively.

We claim that $j > \max\{m, l\}$. Indeed, if $j \leq m$, then $\nu_m^{(1)} < \nu_m$ and, by definition of the Janet division, this implies that $m \notin N_{J,\mathcal{N}}(\nu^{(1)})$, a contradiction. Similarly, we cannot have $j < l$, as then $l \notin N_{J,\mathcal{N}\cup\{\rho\}}(\rho)$. Finally, $j = l$ is not possible. As we know already that $j > m$, we have in this case that $\rho_i = \nu_i^{(1)} = \nu_i$ for all $i > j$ and $\rho_j \leq \nu_j$. Hence $j \in \bar{N}_{J,\mathcal{N}\cup\{\rho\}}(\nu)$ and this implies furthermore $j \in \bar{N}_{J,\mathcal{N}\cup\{\rho\}}(\rho)$, a contradiction.

Now we are going to construct a sequence as in Definition 6.3.3 of a constructive division. Choose an index j_1 with $\lambda_{j_1} > 0$ and $j_1 \in \bar{N}_{J,\mathcal{N}}(\nu^{(1)})$. Such an index exists, as otherwise $\nu + 1_j \in \mathcal{C}_{J,\mathcal{N}}(\nu^{(1)}) \subset \langle \mathcal{N} \rangle_J$. So we may write $\nu + 1_j = (\nu^{(1)} + 1_{j_1}) + \mu + \lambda - 1_{j_1}$. Because of $|\mu| > 0$, the multi index $\nu^{(1)} + 1_{j_1}$ is a proper divisor of $\nu + 1_j$ and according to our assumptions a $\nu^{(2)} \in \mathcal{N}$ exists with $\nu^{(1)} + 1_{j_1} \in \mathcal{C}_{J,\mathcal{N}}(\nu^{(2)})$.

By the same arguments as above an index $j_2 \in \bar{N}_{J,\mathcal{N}}(\nu^{(2)})$ must exist with $(\mu + \lambda - 1_{j_1})_{j_2} > 0$ and a further multi index $\nu^{(3)} \in \mathcal{N}$ such that $\nu^{(2)} + 1_{j_2} \in \mathcal{C}_{J,\mathcal{N}}(\nu^{(3)})$. Thus we can iterate and produce an infinite sequence $(\nu^{(1)}, \nu^{(2)}, \dots)$ where $\nu^{(i+1)} |_{J,\mathcal{N}} \nu^{(i)} + 1_{j_i}$ with $j_i \in \bar{N}_{J,\mathcal{N}}(\nu^{(i)})$. By the continuity of the Janet division all members of the sequence must be different. But every multi index $\nu^{(i)}$ is a divisor of $\nu + 1_j$, so obviously only finitely many of them can be different. Thus the sequence must terminate which only happens, if $\nu + 1_j \in \mathcal{C}_{J,\mathcal{N}}(\nu^{(i)})$ for some i contradicting our assumptions. □

6.4 Computation of Involutive Bases

We present now an algorithm for determining involutive bases of left ideals in a polynomial algebra of solvable type $(\mathcal{P}, \star, \prec)$. Note that this includes simultaneously as special cases the completion of polynomial ideals and of linear systems of differential equations. As mentioned above, for arbitrary involutive divisions, nobody has so far been able to find a reasonable algorithm. But if we assume that the division is continuous and constructive, then a very simple algorithm exists. As in Chapter 2 we start with the "monomial" case, i. e. we consider the problem how we can complete a given finite set $\mathcal{N} \subset \mathbb{N}_0^n$ of multi indices to an involutive set with respect to some involutive division L, and extend later to finite polynomial sets $\mathcal{F} \subset \mathcal{P}$.

Algorithm 6.2 Completion in $(\mathbb{N}_0^n, +)$

Input: a finite set $\mathcal{N} \subset \mathbb{N}_0^n$, an involutive division L, a term order $\prec$
Output: an involutive completion $\bar{\mathcal{N}}$ of $\mathcal{N}$
/1/ $\bar{\mathcal{N}} \leftarrow \mathcal{N}$
/2/ **repeat**
/3/ $\quad \mathcal{S} \leftarrow \{\nu + 1_j \mid \nu \in \bar{\mathcal{N}},\ j \in \bar{N}_{L,\bar{\mathcal{N}}}(\nu),\ \nu + 1_j \notin \langle \bar{\mathcal{N}} \rangle_L\}$
/4/ $\quad \bar{\mathcal{N}} \leftarrow \bar{\mathcal{N}} \cup \{\min_\prec \mathcal{S}\}$
/5/ **until** $\mathcal{S} = \emptyset$
/6/ **return** $\bar{\mathcal{N}}$

The strategy behind Algorithm 6.2 is fairly natural given the results of the last section. In Line /3/ it collects in a set $\mathcal{S}$ all "minimal" obstructions to involution. For a continuous division L, the set $\mathcal{N}$ is involutive, if and only if $\mathcal{S} = \emptyset$. Furthermore, for a constructive division L it does not make sense to add elements of $\langle\mathcal{N}\rangle_L$ to $\mathcal{N}$ in order to complete. Thus we add in Line /4/ an element of $\mathcal{S}$. This element is chosen with the help of a term order.

Proposition 6.4.1. *Let the finite set $\mathcal{N} \subset \mathbb{N}_0^n$ possess a finite involutive completion with respect to the continuous and constructive division L. Then Algorithm 6.2 terminates with an involutive completion $\bar{\mathcal{N}}$ of $\mathcal{N}$.*

Proof. If Algorithm 6.2 terminates, its correctness is obvious. The criterion for its termination, $\mathcal{S} = \emptyset$, is equivalent to local involution of $\bar{\mathcal{N}}$. By Proposition 6.3.4, local involution implies for a continuous division involution. Thus the result $\bar{\mathcal{N}}$ is an involutive completion of $\mathcal{N}$, as by construction $\mathcal{N} \subseteq \bar{\mathcal{N}} \subset \langle\mathcal{N}\rangle$.

If the input set $\mathcal{N}$ is already involutive, Algorithm 6.2 leaves it unchanged and thus obviously terminates. Let us assume that $\bar{\mathcal{N}}$ is not yet involutive. In the first iteration of the `repeat` loop a multi index of the form $\nu + 1_j$ is added to $\mathcal{N}$. It is not contained in $\langle\mathcal{N}\rangle_L$ and minimal with respect to $\prec$ among all such non-multiplicative "multiples" of multi indices in $\mathcal{N}$. If $\mathcal{N}_L$ is an arbitrary involutive completion of $\mathcal{N}$, it must contain a multi index $\mu \notin \mathcal{N}$ such that $\mu \mid_{L,\mathcal{N}_L} \nu + 1_j$. We claim that $\mu = \nu + 1_j$.

Assume that $\mu \neq \nu + 1_j$. Since $\mathcal{N}_L \subset \langle\mathcal{N}\rangle$, μ must lie in the cone of some multi index $\nu^{(1)} \in \mathcal{N}$. We will show that, because of the continuity of L, $\mu \in \langle\mathcal{N}\rangle_L$ contradicting the constructivity of L. If $\nu^{(1)} \mid_{L,\mathcal{N}} \mu$, we are done. Otherwise we write $\mu = \nu^{(1)} + \rho^{(1)}$ for some multi index $\rho^{(1)} \in \mathbb{N}_0^n$. A non-multiplicative index $j_1 \in \bar{N}_{L,\mathcal{N}}(\nu^{(1)})$ with $\rho^{(1)}_{j_1} > 0$ must exist. Consider the multi index $\nu^{(1)} + 1_{j_1}$. Because of $\nu^{(1)} + 1_{j_1} \mid \mu$ and $\mu \mid \nu + 1_j$ the inequality $\nu^{(1)} + 1_{j_1} \prec (\nu + 1_j)$ holds. As $\nu + 1_j$ is the minimal element of $\mathcal{S}$ with respect to $\prec$, we must have $\nu^{(1)} + 1_{j_1} \in \langle\mathcal{N}\rangle_L$. Thus a multi index $\nu^{(2)} \in \mathcal{N}$ exists such that $\nu^{(2)} \mid_{L,\mathcal{N}} \nu^{(1)} + 1_{j_1}$.

By iteration of this argument we obtain a sequence $(\nu^{(1)}, \nu^{(2)}, \dots)$ where each element $\nu^{(i)}$ is a divisor of μ and where $\nu^{(i+1)} \mid_{L,\mathcal{N}} \nu^{(i)} + 1_{j_i}$ with a non-multiplicative index $j_i \in \bar{N}_{L,\mathcal{N}}(\nu^{(i)})$. This sequence cannot become infinite for a continuous division, as μ has only finitely many different divisors and all the multi indices $\nu^{(i)}$ in arbitrary finite pieces of the sequence must be different. But the sequence can only stop, if some $\nu^{(i)} \in \mathcal{N}$ exists such that $\nu^{(i)} \mid_{L,\mathcal{N}} \mu$ and hence $\mu \in \langle\mathcal{N}\rangle_L$.

Thus *any* involutive completion $\mathcal{N}_L$ of the set $\mathcal{N}$ must contain the multi index $\nu + 1_j$. In the next iteration of the `repeat` loop, Algorithm 6.2 treats the enlarged set $\mathcal{N}_1 = \mathcal{N} \cup \{\nu + 1_j\}$. As any involutive completion $\mathcal{N}_L$ of $\mathcal{N}$ is also an involutive completion of $\mathcal{N}_1$, we may apply the same argument again.

As a completion $\mathcal{N}_L$ is by definition a finite set, we must reach after a finite number k of iterations an involutive basis $\mathcal{N}_k$ of $\langle\mathcal{N}\rangle$. □

Note the crucial difference between this result and Theorem B.2.13 on the termination of the Buchberger algorithm for the construction of Gröbner bases. In the latter case, we can show the termination for arbitrary input, i. e. the theorem provides a constructive proof for the existence of such a basis. Here we are only able to prove the termination under the assumption that a finite involutive basis exists; the existence has to be shown separately. For example, Lemma 2.1.14 guarantees us that any monoid ideal possesses a finite Janet basis.

An interesting aspect of the proof above is that it shows that our completion algorithm is largely independent of the term order $\prec$. It affects only the order in which multi indices are added but not which or how many multi indices are added during the completion. We may even use different term orders at each step, as for the proof it is only important that we always add a multi index that is minimal in $\mathcal{S}$ with respect to *some* term order.

Corollary 6.4.2. *If Algorithm 6.2 terminates, its output* $\tilde{\mathcal{N}}$ *is independent of the chosen term order* $\prec$*. This holds even, if the term order is changed in every iteration of the* `repeat` *loop. Furthermore, if* $\mathcal{N}_L$ *is any involutive completion of* $\mathcal{N}$ *with respect to the division* L*, then* $\tilde{\mathcal{N}} \subseteq \mathcal{N}_L$*.*

Proof. Consider the set $\mathcal{L}(\mathcal{N})$ of all involutive completions of $\mathcal{N}$ with respect to the division L and define

$$\tilde{\mathcal{N}} = \bigcap_{\mathcal{N}_L \in \mathcal{L}(\mathcal{N})} \mathcal{N}_L \,. \tag{6.11}$$

We claim that Algorithm 6.2 determines this set $\tilde{\mathcal{N}}$ independent of the used term order. Obviously, this implies our corollary.

In our proof of Proposition 6.4.1, we have shown that all the multi indices added in the course of Algorithm 6.2 are contained in *any* involutive completion of $\mathcal{N}$. Thus all these multi indices are elements of $\tilde{\mathcal{N}}$. As our algorithm terminates with an involutive completion, its output is an element of $\mathcal{L}(\mathcal{N})$, too. Hence the output must be $\tilde{\mathcal{N}}$. □

Recall that for arbitrary (monomial or polynomial) ideals a basis $\mathcal{F}$ is called *minimal*, if it is not possible to remove an element of $\mathcal{F}$ without losing the property that we have a basis. A similar notion can naturally be introduced for involutive bases.

Definition 6.4.3. *Let* $\mathcal{I} \subseteq \mathbb{N}_0^n$ *be a monoid ideal and* L *an involutive division. An involutive basis* $\mathcal{N}$ *of* $\mathcal{I}$ *with respect to* L *is called* minimal*, if any other involutive basis* $\mathcal{N}'$ *of* $\mathcal{I}$ *with respect to* L *satisfies* $\mathcal{N}' \subseteq \mathcal{N}$*.*

Any monomial ideal has a *unique* minimal monomial basis: take an arbitrary monomial basis and eliminate all terms having a divisor in the basis. Obviously, these eliminations do not change the span and the result is a minimal basis. Corollary 6.4.2 implies similarly that if a monoid ideal $\mathcal{I} \subseteq \mathbb{N}_0^n$ has a finite involutive basis for a given division L, then a unique minimal involutive basis exists. It can easily be constructed by taking the unique minimal basis of $\mathcal{I}$ in the usual sense and applying Algorithm 6.2 to it.

A trivial method to compute an involutive basis for an ideal $\mathcal{I}$ in a polynomial algebra $(\mathcal{P}, \star, \prec)$ of solvable type follows immediately from Remark 2.3.13: we determine first a Gröbner basis $\mathcal{G}$ of $\mathcal{I}$ and then with Algorithm 6.2 the involutive completion of the monomial set $\mathrm{lt}_\prec \mathcal{G}$. In fact, a similar approach is proposed by Sturmfels and White [173] for the construction of Stanley decompositions. However, we prefer to extend the ideas behind Algorithm 6.2 to a direct completion algorithm for polynomial ideals, as we believe that this is more efficient. As a first step, we need subalgorithms for two important tasks: *involutive normal forms* and *involutive head autoreductions*.

The design of an algorithm $\texttt{NormalForm}_{L,\prec}(g, \mathcal{H})$ determining an involutive normal form of the polynomial g with respect to the finite set $\mathcal{H} \subset \mathcal{P}$ is trivial. We may use the standard Algorithm B.1 for normal forms in the Gröbner theory (see p. 254), if we replace in Lines /1/, /5/ and /7/ of it the ordinary divisibility relation by its involutive version. Obviously, this does not affect the termination. Actually, for our purposes it is not necessary to compute a full normal form; we may stop as soon as we have obtained a polynomial that is not involutively head reducible.

A simple realisation of an algorithm $\texttt{InvHeadAutoReduce}_{L,\prec}(\mathcal{F})$ for an involutive head autoreduction is given by Algorithm 6.3. Its correctness is obvious and the termination follows again from the fact that $\prec$ is a well-order, as after each reduction step one element of $\mathcal{H}$ has a smaller leading term than before. For a full autoreduction instead of only a head autoreduction, one must consider in Line /2/ the full support of h and not only the leading term. In this case it is tempting to replace Lines /5-7/ by $\bar{h} \leftarrow \texttt{NormalForm}_{L,\prec}(h, \bar{\mathcal{H}})$. However, this is correct only for global divisions, as then the multiplicative variables are determined with respect to the set $\bar{\mathcal{H}}$ and not $\mathcal{H}$ (cf. Remark 2.3.3).

Note that in the determination of the coefficient $c \in \Bbbk$ in Line /6/ we must divide by the leading coefficient of the product $x^\mu \star f$. In general, this differs from the leading coefficient of f, as one can see from (2.5). However, for the two cases of greatest interest for us, ordinary polynomials and linear differential operators, we may take $\mathrm{lc}_\prec f$ instead.

Algorithm 6.3 Involutive Head Autoreduction in $(\mathcal{P}, \star, \prec)$

Input: a finite set $\mathcal{F} \subset \mathcal{P}$, an involutive division L
Output: an involutively head autoreduced set $\mathcal{H}$ with $\langle\mathcal{H}\rangle = \langle\mathcal{F}\rangle$
/1/ $\mathcal{H} \leftarrow \mathcal{F}$
/2/ **while** $\exists h \in \mathcal{H}, f \in \mathcal{H} \setminus \{h\} : \mathrm{le}_\prec f \mid_{L,\mathrm{le}_\prec \mathcal{H}} \mathrm{le}_\prec h$ **do**
/3/   choose such a pair (h, f)
/4/ $\bar{\mathcal{H}} \leftarrow \mathcal{H} \setminus \{h\}$
/5/ $\mu \leftarrow \mathrm{le}_\prec h - \mathrm{le}_\prec f$
/6/ $c \leftarrow \mathrm{lc}_\prec h / \mathrm{lc}_\prec (x^\mu \star f)$
/7/ $\bar{h} \leftarrow h - cx^\mu \star f$
/8/   **if** $\bar{h} = 0$ **then**
/9/ $\mathcal{H} \leftarrow \bar{\mathcal{H}}$
/10/   **else**
/11/ $\mathcal{H} \leftarrow \bar{\mathcal{H}} \cup \{\bar{h}\}$
/12/   **end_if**
/13/ **end_while**
/14/ **return** $\mathcal{H}$

Based on these two subalgorithms, we propose now Algorithm 6.4 for the computation of involutive bases in $\mathcal{P}$. It follows the same strategy as the monomial algorithm. We multiply each element of the set by its non-multiplicative variables. Then we look whether the result is already contained in the involutive span of the set; if not, it is added to the set. The check whether an element lies in the involutive span is performed via an involutive normal form computation; for each element of the span the involutive normal form is zero. In order to obtain reasonable multiplicative variables, we must take care that our set is always involutively head autoreduced.

The way in which we select in Line /5/ the next polynomial $\bar{g}$ to be treated is more or less identical with the so-called normal selection strategy in the theory of Gröbner bases (see Appendix B.2). There this strategy is known to work very well for degree compatible term orders but not so well for other orders like the purely lexicographic one. Here we will see below that the normal selection strategy is important for proving the termination of the algorithm. It is currently unclear to what extent other strategies may be used for the selection of $\bar{g}$.

In order to prove the correctness and the termination of Algorithm 6.4 we need a generalisation of the notion of local involution and of Proposition 6.3.4 to polynomial sets.

Definition 6.4.4. *The finite set* $\mathcal{F} \subset \mathcal{P}$ *is* locally involutive *for the division* L, *if* $x^j \star f \in \langle\mathcal{F}\rangle_{L,\prec}$ *for any non-multiplicative variable* $x^j \in \bar{X}_{L,\mathcal{F},\prec}(f)$ *of any polynomial* $f \in \mathcal{F}$.

Algorithm 6.4 Completion in $(\mathcal{P}, \star, \prec)$

Input: a finite set $\mathcal{F} \subset \mathcal{P}$, an involutive division L
Output: an involutive basis $\mathcal{H}$ of $\mathcal{I} = \langle\mathcal{F}\rangle$ with respect to L and $\prec$

$\mathcal{H} \leftarrow \mathtt{InvHeadAutoReduce}_{L,\prec}(\mathcal{F})$
loop
  $\mathcal{S} \leftarrow \{x^j \star h \mid h \in \mathcal{H},\ x^j \in \bar{X}_{L,\mathcal{H},\prec}(h),\ x^j \star h \notin \langle\mathcal{H}\rangle_{L,\prec}\}$
  repeat
    $\bar{g} \leftarrow \min_\prec \mathcal{S}$
    $\mathcal{S} \leftarrow \mathcal{S} \setminus \{\bar{g}\}$
    $g \leftarrow \mathtt{NormalForm}_{L,\prec}(\bar{g}, \mathcal{H})$
  until $(g \neq 0)$ or $(\mathcal{S} = \emptyset)$
  if $g \neq 0$ **then**
    $\mathcal{H} \leftarrow \mathtt{InvHeadAutoReduce}_{L,\prec}(\mathcal{H} \cup \{g\})$
  else
    return $\mathcal{H}$
  end_if
end_loop

Proposition 6.4.5. *Given a continuous division* L*, any involutively head autoreduced set* $\mathcal{F} \subset \mathcal{P}$ *that is locally involutive is also involutive.*

Proof. Local involution may be expressed as follows. Choose for an arbitrary polynomial $g \in \mathcal{F}$ a non-multiplicative variable $x^j \in \bar{X}_{L,\mathcal{F}}(g)$. Then for each $f \in \mathcal{F}$ a polynomial $P_f \in \mathbb{k}[X_{L,\mathcal{F},\prec}(f)]$ exists such that

$$x^j \star g = \sum_{f \in \mathcal{F}} P_f \star f \,. \tag{6.12}$$

As by assumption, $\mathcal{F}$ is involutively head autoreduced, (6.12) implies that the set $\mathrm{le}_\prec \mathcal{F} \subset \mathbb{N}_0^n$ is locally involutive and thus by the continuity of L involutive (Proposition 6.3.4). By Theorem 2.3.11, $\mathcal{F}$ is thus an involutive basis of $\mathcal{I} = \langle\mathcal{F}\rangle$, if we can show that $\langle \mathrm{le}_\prec \mathcal{F}\rangle = \mathrm{le}_\prec \mathcal{I}$.

We claim that for any term x^μ and for any polynomial $f_1 \in \mathcal{F}$ a polynomial $h \in \mathcal{F}$ exists such that $\mathrm{le}_\prec(x^\mu \star f_1) \in \mathcal{C}_{L,\mathrm{le}_\prec \mathcal{F}}(h)$. If this is true, we may take arbitrary linear combinations (with coefficients in $\mathbb{k}$) of such products and the leading terms cannot cancel, as $\mathrm{le}_\prec \mathcal{F}$ is a strong involutive basis of its span. But this implies immediately that the leading exponent of any polynomial in $\mathcal{I}$ lies in $\langle \mathrm{le}_\prec \mathcal{F}\rangle$ and the proposition holds.

If the term x^μ contains only variables that are multiplicative for $\mathrm{le}_\prec f_1$, nothing is to be shown. Otherwise we choose an index $j_1 \in \bar{N}_{L,\mathrm{le}_\prec \mathcal{F}}(\mathrm{le}_\prec f_1)$ such that $\mu_{j_1} > 0$. By (6.12), for each $f \in \mathcal{F}$ a polynomial $P_f^{(1)} \in \mathbb{k}[X_{L,\mathcal{F},\prec}(f)]$ exists such that $x^{j_1} \star f_1 = \sum_{f \in \mathcal{F}} P_f^{(1)} \star f$. Let the leading term on the right hand side be $\mathrm{lt}_\prec(P_{f_2}^{(1)} \star f_2)$. If the term $x^{\mu - 1_{j_1}}$ contains only variables that are multiplicative for $\mathrm{le}_\prec f_2$, we are done.

Otherwise we choose a non-multiplicative index $j_2 \in \bar{N}_{L,\mathrm{le}_\prec \mathcal{F}}(\mathrm{le}_\prec f_2)$ such that $(\mu - 1_{j_1})_{j_2} > 0$. Applying again (6.12), for each $f \in \mathcal{F}$ a polynomial $P_f^{(2)} \in \Bbbk[X_{L,\mathcal{F},\prec}(f)]$ exists such that $x^{j_2} \star f_2 = \sum_{f\in\mathcal{F}} P_f^{(2)} \star f$. Let the leading term on the right hand side be $\mathrm{lt}_\prec(P_{f_3}^{(2)} \star f_3)$ and introduce the multi index $\nu = \mathrm{le}_\prec(x^{j_1} \star f_1) - \mathrm{le}_\prec f_2$. Then $\mathrm{le}_\prec(x^\mu \star f_1) = \mathrm{le}_\prec(x^{\mu+\nu-1_{j_1}-1_{j_2}} \star f_3)$, as the multiplication $\star$ in a polynomial algebra of solvable type respects the term order $\prec$. If all variables on the right hand side are multiplicative, we are done.

Otherwise we iterate: we choose an index j_3 and decompose $x^{j_3} \star f_3$ with (6.12) into multiplicative products obtaining a new polynomial f_4 and so on. This process yields a sequence $\bigl(\nu^{(1)} = \mathrm{le}_\prec f_1, \nu^{(2)} = \mathrm{le}_\prec f_2, \dots\bigr)$ where all multi indices $\nu^{(k)}$ are elements of the finite set $\mathrm{le}_\prec\mathcal{F}$ and where to each $\nu^{(k)}$ a non-multiplicative index $j_k \in \bar{N}_{L,\mathrm{le}_\prec\mathcal{F}}(\nu^{(k)})$ exists such that $\nu^{(k+1)}|_{L,\mathrm{le}_\prec\mathcal{F}}\nu^{(k)} + 1_{j_k}$. By the definition of a continuous division, this sequence cannot become infinite and the above described process must terminate. But this implies, as claimed, the existence of a polynomial $h \in \mathcal{F}$ such that $\mathrm{le}_\prec(x^\mu \star f_1) \in \mathcal{C}_{L,\mathrm{le}_\prec\mathcal{F}}(h)$. □

As in the proof of Proposition 6.4.1, local involution of $\mathcal{H}$ is obviously equivalent to the termination condition $\mathcal{S} = \emptyset$ of the `repeat` loop in Algorithm 6.4. Thus we are now in the position to prove the following result.

Theorem 6.4.6. *Let* L *be a continuous and constructive involutive division and* $(\mathcal{P}, \star, \prec)$ *a polynomial algebra of solvable type. Assume that the (left) ideal* $\mathcal{I} \subseteq \mathcal{P}$ *is such that* $\mathrm{le}_\prec\mathcal{I}$ *possesses a finite involutive completion. Then Algorithm 6.4 terminates for any finite generating set* $\mathcal{F}$ *of* $\mathcal{I}$ *with an involutive basis.*

Proof. We begin by proving the *correctness* of the algorithm under the assumption that it terminates. The relation $\mathcal{I} = \langle\mathcal{H}\rangle$ remains valid throughout, although $\mathcal{H}$ changes. But the only changes are the addition of further elements of $\mathcal{I}$ and involutive head autoreductions; both operations do not affect the ideal generated by $\mathcal{H}$. When the algorithms terminates, we have $\mathcal{S} = \emptyset$ and thus the output $\mathcal{H}$ is locally involutive and by Proposition 6.4.5 involutive.

There remains the problem of *termination*. Algorithm 6.4 produces a sequence $(\mathcal{H}_1, \mathcal{H}_2, \dots)$ with $\langle\mathcal{H}_i\rangle = \mathcal{I}$. The set $\mathcal{H}_{i+1}$ is determined from $\mathcal{H}_i$ in Line /10/. We distinguish two cases, namely whether or not during the computation of the involutive normal form in Line /7/ the leading exponent changes. If $\mathrm{le}_\prec\bar{g} = \mathrm{le}_\prec g$, then $\langle\mathrm{le}_\prec\mathcal{H}_i\rangle = \langle\mathrm{le}_\prec\mathcal{H}_{i+1}\rangle$, as $\mathrm{le}_\prec g = \mathrm{le}_\prec h + 1_j$ for some $h \in \mathcal{H}_i$. Otherwise we claim that $\langle\mathrm{le}_\prec\mathcal{H}_i\rangle \subsetneq \langle\mathrm{le}_\prec\mathcal{H}_{i+1}\rangle$.

By construction, the polynomial g is in involutive normal form with respect to the set $\mathcal{H}_i$ implying that $\mathrm{le}_\prec g \in \langle\mathrm{le}_\prec\mathcal{H}_i\rangle \setminus \langle\mathrm{le}_\prec\mathcal{H}_i\rangle_L$. If we had $\langle\mathrm{le}_\prec\mathcal{H}_i\rangle = \langle\mathrm{le}_\prec\mathcal{H}_{i+1}\rangle$, a polynomial $h \in \mathcal{H}_i$ would exist such that

$\text{le}_\prec g = \text{le}_\prec h + \mu$ where the multi index μ has a non-vanishing entry μ_j for at least one non-multiplicative index $j \in \bar{N}_{L,\text{le}_\prec \mathcal{H}_i}(h)$. This implies that $\text{le}_\prec h + 1_j \preceq \text{le}_\prec g \prec \text{le}_\prec \bar{g}$. But we choose the polynomial $\bar{g}$ in Line /5/ such that its leading exponent is minimal among all non-multiplicative products $x^k \star h$ with $h \in \mathcal{H}_i$; hence $\text{le}_\prec \bar{g} \preceq \text{le}_\prec h + 1_j$. As this is a contradiction, we must have $\langle \text{le}_\prec \mathcal{H}_i \rangle \subsetneq \langle \text{le}_\prec \mathcal{H}_{i+1} \rangle$.

So the outer infinite loop generates an ascending chain of monoid ideals $\langle \text{le}_\prec \mathcal{H}_1 \rangle \subseteq \langle \text{le}_\prec \mathcal{H}_2 \rangle \subseteq \cdots \subseteq \text{le}_\prec \mathcal{I}$. As $\mathbb{N}_0^n$ is Noetherian, the chain must become stationary at some index N. It follows from the considerations above that in all iterations of the outer loop after the Nth one $\text{le}_\prec \bar{g} = \text{le}_\prec g$ in Line /7/. At this stage Algorithm 6.4 reduces to an involutive completion of the set $\text{le}_\prec \mathcal{H}_N$ using Algorithm 6.2. Indeed, in Line /5/ we choose the polynomial $\bar{g}$ such that $\text{le}_\prec \bar{g}$ is the same multi index as Algorithm 6.2 adds in Line /4/. By Proposition 6.4.1, the latter algorithm terminates under the made assumptions and hence Algorithm 6.4 terminates, too. □

Corollary 6.4.7. *Let L be a Noetherian division. Then every ideal $\mathcal{I} \subseteq \mathcal{P}$ possesses a finite involutive basis with respect to the division L. Furthermore, in suitably chosen variables x every ideal has a finite Pommaret basis, if the term order $\prec$ is class respecting.*

Proof. The first part is trivial, as for a Noetherian division by definition any monoid ideal possesses a finite involutive basis. Thus the conditions of Theorem 6.4.6 are always satisfied. For the second part we invoke the results of Section 2.4 on δ-regular coordinate systems. As a first step we show that every ideal has a *Pommaret* head autoreduced *Janet* basis. Indeed, let us apply Algorithm 6.4 for the Janet division with a slight modification: in the Lines /2/ and /10/ we perform the involutive head autoreductions with respect to the Pommaret division. It is obvious that if the algorithm terminates, the result is a basis with the wanted properties.

The Janet division is Noetherian by Lemma 2.1.14. Thus without our modification the termination is obvious. With respect to the Janet division every set of multi indices is involutively autoreduced. Hence a Janet head autoreduction only takes care that no two elements of a set have the same leading exponents (recall Example 2.3.5). But in Line /10/ we add a polynomial that is in involutive normal form so that no involutive head reductions are possible. As the Pommaret head autoreduction may only lead to a larger monoid ideal $\text{le}_\prec \mathcal{H}_i$, the Noetherian argument in the proof of Theorem 6.4.6 remains valid after our modification.

Once the ascending ideal chain $\langle \text{le}_\prec \mathcal{H}_1 \rangle \subseteq \langle \text{le}_\prec \mathcal{H}_2 \rangle \subseteq \cdots$ has become stationary, Algorithm 6.4 reduces to the "monomial" Algorithm 6.2. According to Corollary 2.4.8, the Pommaret head autoreductions may only increase the

Janet spans $\langle \mathrm{le}_\prec \mathcal{H}_i \rangle_J$. Thus the termination of Algorithm 6.2 is not affected by our modification and the algorithm terminates for arbitrary input.

By Proposition 2.4.6, the coordinate systems that are δ-singular for a given finite set $\mathcal{F} \subset \mathcal{P}$ form a Zariski closed set. As our modified algorithm terminates, it treats only a finite number of sets $\mathcal{H}_i$ and the coordinate systems that are δ-singular for at least one of them still form a Zariski closed set. Thus generic coordinates are δ-regular for all sets $\mathcal{H}_i$ and by Corollary 2.4.11 their Janet and their Pommaret spans coincide for a class respecting term order. But this implies that the result of the modified algorithm is not only a Janet but also a Pommaret basis. □

Theorem 2.4.10 provides us with an effective mean to detect that the completion to a Pommaret basis does *not* terminate. We follow Algorithm 6.4 with the Pommaret division (and a class respecting term order) but check after each iteration whether $|\mathcal{H}_i|_{J,\prec} = |\mathcal{H}_i|_{P,\prec}$. If no finite Pommaret basis exists in the given coordinates, sooner or later the Janet size is greater than the Pommaret size. Indeed, by the considerations above the completion with respect to the Janet division terminates. Thus either the result is simultaneously a Pommaret basis or our criterion must detect a variable which is multiplicative only for the Janet but not for the Pommaret division.

Example 6.4.8. Let us apply Algorithm 6.4 to the Pommaret completion of the set $\mathcal{F} = \{\underline{z^2} + yz - x^2,\ \underline{xz} + xy - z,\ \underline{yz} + y^2 + x^2 - x\}$ (with respect to the degree reverse lexicographic order). We have seen in Example 2.4.12 that the coordinates are not δ-regular, although the Janet and the Pommaret span of $\mathcal{F}$ coincide. According to our algorithm we must first analyse the polynomial $y(xz + xy - z)$. Its involutive normal form with respect to $\mathcal{F}$ is $-\underline{x^3}+y^2+2x^2-x$. If we determine the multiplicative variables for the enlarged set, they do not change for the old elements. For the new polynomial the Janet division yields $\{x, y\}$. But y is obviously not multiplicative for the Pommaret division. Thus our criterion tells us that the Pommaret completion will not terminate. Indeed it is easy to see that no matter how often we multiply the new polynomial by y, it will never become involutively head reducible and no finite Pommaret basis exists.

The Janet completion (with Pommaret autoreductions) yields for this example the following involutive basis

$$\begin{aligned} \mathcal{F}_J = \{&\underline{z^2} - y^2 - 2x^2 + x,\ \underline{xz} + xy - z,\ \underline{yz} + y^2 + x^2 - x, \\ &\underline{y^2} - x,\ \underline{x^3} - 2x^2 - x + y,\ \underline{x^2y} - xy + z\}\ . \end{aligned} \tag{6.13}$$

Without the Pommaret autoreductions the basis is even larger. In Example 2.4.12 we showed that the transformation $\tilde{x} = z$, $\tilde{y} = y + z$ and $\tilde{z} = x$ yields after an autoreduction the set $\tilde{\mathcal{F}}^{\triangle} = \{\underline{\tilde{z}^2} - \tilde{x}\tilde{y},\ \underline{\tilde{y}\tilde{z}} - \tilde{x},\ \underline{\tilde{y}^2} - \tilde{z}\}$. One easily checks that it is a Pommaret and thus also a Janet basis. We see again that other divisions also "feel" δ-singularity in the sense that in such coordinates they typically lead to larger bases of higher degree. For efficiency reasons one should therefore always try to work in δ-regular coordinates. $\lhd$

Our proof of Theorem 6.4.6 has an interesting consequence. Assume that the term order $\prec$ is of type ω, i. e. for any two multi indices μ, ν with $\mu \prec \nu$ only finitely many multi indices $\rho^{(i)}$ exist with $\mu \prec \rho^{(1)} \prec \rho^{(2)} \prec \cdots \prec \nu$. Then even if our algorithm does *not* terminate, it determines in a finite number of steps a Gröbner basis of the ideal $\mathcal{I}$.

Proposition 6.4.9. *Let the term order $\prec$ be of type ω. Then Algorithm 6.4 determines for any input $\mathcal{F}$ in a finite number of steps a Gröbner basis of the ideal $\mathcal{I} = \langle \mathcal{F} \rangle$.*

Proof. In the proof of Theorem 6.4.6 we introduced the set $\mathcal{H}_N$ such that $\langle \mathcal{H}_{N+\ell} \rangle = \langle \mathcal{H}_N \rangle$ for all $\ell > 0$. We claim that $\mathcal{H}_N$ is a Gröbner basis of $\mathcal{I}$.

Let $f \in \mathcal{I}$ be an arbitrary element of the ideal. As $\mathcal{H}_N$ is a basis of $\mathcal{I}$, we find for each $h \in \mathcal{H}_N$ a polynomial $g_h \in \mathcal{P}$ such that

$$f = \sum_{h \in \mathcal{H}_N} g_h \star h \,. \tag{6.14}$$

By Proposition B.2.7, $\mathcal{H}_N$ is a Gröbner basis, if and only if we can choose the coefficients g_h such that $\mathrm{lt}_\prec(g_h \star h) \preceq \mathrm{lt}_\prec f$. Assume that for f no such standard representation exists and let $\mu = \max_{h \in \mathcal{H}_N} \{\mathrm{le}_\prec g_h + \mathrm{le}_\prec h\}$. If we denote by $\bar{\mathcal{H}}_N$ the set of all polynomials $\bar{h} \in \mathcal{H}_N$ for which $\mathrm{le}_\prec g_{\bar{h}} + \mathrm{le}_\prec \bar{h} = \mu$, then we must have a non-trivial syzygy $\sum_{\bar{h} \in \bar{\mathcal{H}}_N} \big(\mathrm{le}_\prec g_{\bar{h}} + \mathrm{le}_\prec \bar{h}\big) = 0$. By Lemma 2.6.1, at least one $\bar{h} \in \bar{\mathcal{H}}_N$ exists such that some non-multiplicative variable $x^j \in \bar{X}_{L,\mathcal{H}_N}(\bar{h})$ divides $\mathrm{lt}_\prec g_{\bar{h}}$.

As $\prec$ is of type ω, after a finite number of steps the non-multiplicative product $x^j \star \bar{h}$ is analysed in Algorithm 6.4. Thus for some $n_1 \geq 0$ the set $\mathcal{H}_{N+n_1}$ contains an element $\bar{h}'$ with $\mathrm{le}_\prec \bar{h}' = \mathrm{le}_\prec (x^j \star \bar{h})$. Let $\mu = \mathrm{le}_\prec g_{\bar{h}}$, $x^{\mu - 1_j} \star x^j = c x^\mu + r_1$ and $\bar{h}' = d x^j \star \bar{h} + r_2$. Then we may rewrite

$$g_{\bar{h}} \star \bar{h} = \frac{\mathrm{lc}_\prec g_{\bar{h}}}{cd} \big[x^{\mu - 1_j} \star (\bar{h}' - r_2) - d r_1 \star \bar{h}\big] + \big(g_{\bar{h}} - \mathrm{lm}_\prec g_{\bar{h}}\big) \star \bar{h} \,. \tag{6.15}$$

As $\bar{h}'$ was determined via an involutive normal form computation applied to $x^j \star \bar{h}$ and as we know that at this stage of the algorithm the leading term does not change during the computation, the leading term on the right hand side of (6.15) is $\mathrm{lt}_\prec (x^{\mu - 1_j} \star \bar{h}')$. If the term $x^{\mu - 1_j}$ contains a non-multiplicative variable $x^k \in \bar{X}_{L,\mathcal{H}_{N+n_1}}(\bar{h}')$, we repeat the argument obtaining an element $\bar{h}'' \in \mathcal{H}_{N+n_1+n_2}$ such that $\mathrm{le}_\prec \bar{h}'' = \mathrm{le}_\prec (x^k \star \bar{h}')$.

Obviously, this process terminates after a finite number of steps — even if we do it for each $\bar{h} \in \bar{\mathcal{H}}_N$. Thus after $N + \ell$ iterations (for some finite number $\ell \geq 0$) we obtain a set $\mathcal{H}_{N+\ell}$ such that, after applying all the found relations (6.15), f can be expressed in the form $f = \sum_{h \in \mathcal{H}_{N+\ell}} \tilde{g}_h \star h$ where still $\mu = \max_{h \in \mathcal{H}_{N+\ell}} \{\mathrm{le}_\prec \tilde{g}_h + \mathrm{le}_\prec h\}$. Denote again by $\bar{\mathcal{H}}_{N+\ell} \subseteq \mathcal{H}_{N+\ell}$ the set of all those polynomials $\bar{h}$ for which this maximum is achieved.

By construction, no term $\mathrm{lt}_\prec \bar{g}_{\bar{h}}$ with $\bar{h} \in \bar{\mathcal{H}}_{N+\ell}$ contains a variable that is non-multiplicative for $\bar{h}$. Hence, by Lemma 2.6.1, it is not possible that $\sum_{\bar{h} \in \bar{\mathcal{H}}_{N+\ell}} (\mathrm{le}_\prec \tilde{g}_{\bar{h}} + \mathrm{le}_\prec \bar{h}) = 0$. But this contradicts $\mu \succ \mathrm{lt}_\prec f$. Thus each polynomial $f \in \mathcal{P}$ possesses a standard representation already with respect to $\mathcal{H}_N$ and this set is a Gröbner basis. ∎

Remark 6.4.10. All degree compatible term orders are of type ω. Purely lexicographic orders are not of type ω. Indeed, if $x \prec_{\mathrm{lex}} y$, then the infinitely many terms x^k are all less than y with respect to $\prec_{\mathrm{lex}}$. ◁

The notion of a minimal involutive basis may be extended to ideals in $(\mathcal{P}, \star, \prec)$ in close analogy to Definition B.2.10 of a minimal Gröbner basis.

Definition 6.4.11. *Let $\mathcal{I} \subseteq \mathcal{P}$ be an ideal and L an involutive division. An involutive basis $\mathcal{H}$ of $\mathcal{I}$ with respect to L is* minimal*, if $\mathrm{lt}_\prec \mathcal{H}$ is a minimal involutive basis of $\mathrm{lt}_\prec \mathcal{I}$ for L.*

Uniqueness requires now two additional assumptions. First of all, our definition of an involutive basis requires only that it is involutively head autoreduced; for uniqueness we obviously need a full involutive autoreduction. Secondly, we must normalise the leading coefficients to one, i. e. we must take a *monic* basis.

Proposition 6.4.12. *Let $\mathcal{I} \subseteq \mathcal{P}$ be an ideal and L an involutive division. Then $\mathcal{I}$ has at most one monic, involutively autoreduced, minimal involutive basis with respect to L.*

Proof. Assume that $\mathcal{H}_1$ and $\mathcal{H}_2$ are two different monic, involutively autoreduced, minimal involutive bases of $\mathcal{I}$ with respect to L and $\prec$. By definition of a minimal involutive bases, this implies that $\mathrm{lt}_\prec \mathcal{H}_1 = \mathrm{lt}_\prec \mathcal{H}_2$. As $\mathcal{H}_1$ and $\mathcal{H}_2$ are not identical, we must have two polynomials $h_1 \in \mathcal{H}_1$ and $h_2 \in \mathcal{H}_2$ such that $\mathrm{lt}_\prec h_1 = \mathrm{lt}_\prec h_2$ but $h_1 \neq h_2$. Now consider the polynomial $h = h_1 - h_2 \in \mathcal{I}$. Its leading term must lie in the involutive span of $\mathrm{lt}_\prec \mathcal{H}_1 = \mathrm{lt}_\prec \mathcal{H}_2$. On the other hand, $\mathrm{lt}_\prec h \in \operatorname{supp} h_1 \cup \operatorname{supp} h_2$. But this implies that either $\mathcal{H}_1$ or $\mathcal{H}_2$ is not involutively autoreduced. ∎

In contrast to the monomial case, one does not automatically obtain a minimal involutive basis by making some minor modifications of Algorithm 6.4. In particular, it does not suffice to apply it to a minimal basis in the ordinary sense. Gerdt and Blinkov [57] presented an algorithm that always returns a minimal involutive basis provided a finite involutive basis exists. While it still follows the same basic strategy of study all products with non-multiplicative variables, it requires a more complicated organisation of the algorithm. We omit here the details.

6.5 Linear Differential Equations

We stressed in Section 6.2 that the Cartan–Kuranishi completion is more a "method" than an algorithm, as it is not clearly specified how the required operations should be performed. We now develop for linear differential equations an effective algebraic realisation of it based on Algorithm 6.4 for involutive bases. A naive realisation of the Cartan–Kuranishi completion is immediate: one explicitly determines local representations for all differential equations $\mathcal{R}^{(s)}_{q+r}$ and symbols $\mathcal{M}^{(s)}_{q+r}$ appearing in the course of the algorithm. For linear or polynomial equations it is – at least in principle – straightforward to perform all required operations algorithmically using Gröbner bases.

Obviously, such a brute force approach is not very efficient; in general, many unnecessary prolongations (which are usually rather expensive to compute) are performed and one has to deal with rather large matrices (recall Example 6.2.2). Pure algebra in form of involutive bases leads to fairly fast algorithms for linear equations, but all geometric information is lost. For example, it is not possible to determine the number of performed prolongations and projections. We combine the geometric and the algebraic approach into a "hybrid" algorithm, so that we maintain the efficiency of pure algebra and still obtain the full geometric information.

In the sequel we assume that the coefficients of our system stem from a function field $\mathbb{k}$ where all required operations (arithmetics and differentiations) can be performed effectively and that contains the coordinates $\mathbf{x}$. Thus a typical instance of $\mathbb{k}$ is a finite algebraic extension of the field $\mathbb{Q}(x^1, \dots, x^n)$ of rational functions with rational coefficients.

We assume that the linear system is given in the form

$$\Phi^\tau(x,p) = A^{\tau\mu}_\alpha(x) p^\alpha_\mu = 0 , \qquad \tau = 1, \dots, t \tag{6.16}$$

where $0 \leq |\mu| \leq q$. Thus we are only considering homogeneous equations. But of course one could add right hand sides as discussed in Section 3.4. For the remainder of this section, we concentrate on the left hand side and call it a *row*. Thus a linear system is for us a set $\mathcal{F}$ of rows; one may also say that we are operating on a matrix representation of the system.

The derivatives are ordered according to the degree reverse lexicographic order $\prec_{\text{degrevlex}}$; the leading term of a row $f \in \mathcal{F}$ is called *leading derivative* and denoted by[4] $\operatorname{ld} f$. As usual, multiplicative and non-multiplicative variables are assigned to a row according to its leading derivative. Reductions correspond now to elementary row operations and an autoreduction yields a triangular form of the system. We introduce the following notation for the result of a reduction. Let $f = a(x) \operatorname{ld} f + \sum_{p^\alpha_\mu \prec \operatorname{ld} f} A^\mu_\alpha(x) p^\alpha_\mu$ and $g = b(x) \operatorname{ld} g + \sum_{p^\alpha_\mu \prec \operatorname{ld} g} B^\mu_\alpha(x) p^\alpha_\mu$ be two rows of the system $\mathcal{F}$ with $\operatorname{ld} f = \operatorname{ld} g$; then

$$\operatorname{rem}(f, g) = \sum_{p^\alpha_\mu \prec \operatorname{ld} f} \left(A^\mu_\alpha(x) - \frac{a(x)}{b(x)} B^\mu_\alpha(x) \right) p^\alpha_\mu \,. \tag{6.17}$$

Note that this is a purely algebraic operation; no differentiations are allowed!

In the naive realisation of the Cartan–Kuranishi completion it is necessary to differentiate every equation in the system with respect to every independent variables in order to obtain a representation of the prolonged equation. We use the Pommaret division to distinguish multiplicative and non-multiplicative variables. Our goal is to avoid as much as possible to perform differentiations with respect to multiplicative variables. As in Algorithm 6.4 they should appear only in involutive normal form computations. In order to achieve this goal and still be able to decide involution of the appearing symbols and differential equations, we need a careful book-keeping. The necessary tools are introduced in the next definition. It might appear somewhat strange at first sight, but it will prove its usefulness in the sequel.

Definition 6.5.1. *Let* f *denote a row from a linear system* $\mathcal{F}$*. We can make* f *into an* indexed row*, either a* single indexed row $f_{(k)}$ *or a* double indexed row *(or* phantom row*)* $f_{(k,\ell)}$ *by assigning to it one respectively two non-negative integers* k *and* ℓ *where* $0 \leq k < \ell$ *in the second case. We call* k *the* initial level *and* ℓ *the* phantom level *of the indexed row. For a* global level $\lambda \geq k$*, we define the* truncated involutive cone $\bar{\mathcal{C}}_\lambda$ *of an indexed row to be the following set of rows:*

$$\bar{\mathcal{C}}_\lambda(f_{(k)}) = \left\{ D_\mu f \mid 0 \leq |\mu| \leq (\lambda - k);\ \forall i > \operatorname{cls} f : \mu_i = 0 \right\} , \tag{6.18a}$$

$$\bar{\mathcal{C}}_\lambda(f_{(k,\ell)}) = \left\{ D_\mu f \mid (\lambda - \ell) < |\mu| \leq (\lambda - k);\ \forall i > \operatorname{cls} f : \mu_i = 0 \right\} . \tag{6.18b}$$

The truncated involutive cone of an indexed row determined by the numbers λ, k and possibly ℓ contains all multiplicative prolongations whose orders satisfy the given conditions. With the help of this notion, we introduce the basic object of our approach, the skeleton of a linear system.

---

[4] As we have now fixed one specific term order, we omit for notational simplicity the subscript $\prec$.

Definition 6.5.2. *A* skeleton $\mathcal{S}_\lambda$ *with global level* λ *is the union of a set* $\mathcal{U}_\lambda = \{f^\sigma_{(k_\sigma)}\}$ *of single indexed rows and a set* $\mathcal{W}_\lambda = \{f^\sigma_{(k_\sigma,l_\sigma)}\}$ *of double indexed rows such that for all* σ *the inequalities* $k_\sigma \le \lambda$ *and* $k_\sigma < l_\sigma \le \lambda$*, respectively, hold. It defines the linear system*

$$\bar{\mathcal{S}}_\lambda = \left\{\bar{\mathcal{C}}_\lambda(f_{(k)}) \mid f_{(k)} \in \mathcal{U}_\lambda\right\} \cup \left\{\bar{\mathcal{C}}_\lambda(f_{(k,\ell)}) \mid f_{(k,\ell)} \in \mathcal{W}_\lambda\right\}. \tag{6.19}$$

Thus in order to obtain the linear system $\bar{\mathcal{S}}_\lambda$ corresponding to a given skeleton $\mathcal{S}_\lambda$, we must simply add all permitted multiplicative prolongations of the contained rows and drop the indices.

In each iteration of the Cartan–Kuranishi completion, the system is *prolonged* one order. Since we are interested in autoreduced systems, we must afterwards perform a *triangulation*. We consider now how these two operations are carried out efficiently on the level of skeletons. We may assume that the system $\mathcal{F}$ to be completed is already given in triangular, i. e. (head) autoreduced, form. We turn $\mathcal{F}$ into a skeleton $\mathcal{S}_0$ by simply setting the initial level k_σ of every row f^σ to 0. The global level λ is always equal to the number of the current iteration step; if a new row enters the skeleton in this step, its initial level becomes λ. Double indexed rows arise, if during the triangulation process a row is eliminated which has already produced some multiplicative prolongations. In this case, the phantom level is set to the current global level. Thus the completion algorithm produces a sequence of skeletons

$$\mathcal{S}_0 = \mathcal{S}_0^\triangle \longrightarrow \mathcal{S}_1 \longrightarrow \mathcal{S}_1^\triangle \longrightarrow \mathcal{S}_2 \longrightarrow \mathcal{S}_2^\triangle \longrightarrow \mathcal{S}_3 \longrightarrow \cdots \tag{6.20}$$

where the $^\triangle$ indicates a skeleton in triangular form.

Algorithm 6.5 Prolongation of a Skeleton

Input: skeleton $\mathcal{S}_\lambda = \mathcal{U}_\lambda \cup \mathcal{W}_\lambda$, number n of independent variables
Output: prolonged skeleton $\mathcal{S}_{\lambda+1} = \mathcal{U}_{\lambda+1} \cup \mathcal{W}_{\lambda+1}$
/1/ $\mathcal{U}_{\lambda+1} \leftarrow \mathcal{U}_\lambda$
/2/ $\mathcal{W}_{\lambda+1} \leftarrow \mathcal{W}_\lambda$
/3/ **for all** $f^\sigma_{(\lambda)} \in \mathcal{U}_\lambda$ **do**
/4/   **for** $\operatorname{cls} f^\sigma_{(\lambda)} \le i \le n$ **do**
/5/ $\mathcal{U}_{\lambda+1} \leftarrow \mathcal{U}_{\lambda+1} \cup \left\{(D_i f^\sigma)_{(\lambda+1)}\right\}$
/6/   **end_for**
/7/ **end_for**
/8/ **return** $\mathcal{U}_{\lambda+1} \cup \mathcal{W}_{\lambda+1}$

Algorithm 6.5 merely computes the non-multiplicative prolongations of those single indexed rows that have been newly added in the previous iteration. Due to the increase of the global level multiplicative prolongations of *all* rows from the skeleton are automatically present in the linear system it defines. No rows are explicitly removed; the new global level causes the elimination of the least-order multiplicative prolongations of all phantom rows in the system. We will show below that these operations yield the desired skeleton for the prolonged system.

Algorithm 6.6 Triangulation of a Skeleton

Input: skeleton $\mathcal{S}_\lambda = \mathcal{U}_\lambda \cup \mathcal{W}_\lambda$

Output: equivalent skeleton $\mathcal{S}^\triangle_\lambda = \mathcal{U}^\triangle_\lambda \cup \mathcal{W}^\triangle_\lambda$ such that $\bar{\mathcal{S}}^\triangle_\lambda$ is in triangular form

/1/ $\mathcal{U}^\triangle_\lambda \leftarrow \emptyset$

/2/ $\mathcal{W}^\triangle_\lambda \leftarrow \mathcal{W}_\lambda$

/3/ **while** $\mathcal{U}_\lambda \neq \emptyset$ **do**

/4/ $\quad f^\sigma_{(k_\sigma)} \leftarrow \max_\prec \mathcal{U}_\lambda$

/5/ $\quad \bar{f} \leftarrow f^\sigma$

/6/ $\quad \mathcal{U}_\lambda \leftarrow \mathcal{U}_\lambda \setminus \{f^\sigma_{(k_\sigma)}\}$

/7/ $\quad$ **while** $\exists f^\tau \in \mathcal{S}_\lambda : \operatorname{ld} \bar{f} \in \bar{\mathcal{C}}_\lambda(\operatorname{ld} f^\tau)$ **do**

/8/ $\quad\quad p^\alpha_\mu \leftarrow \operatorname{ld} \bar{f}$

/9/ $\quad\quad p^\alpha_\nu \leftarrow \operatorname{ld} f^\tau$

/10/ $\quad\quad g \leftarrow D_{\mu-\nu} f^\tau$

/11/ $\quad\quad \bar{f} \leftarrow \operatorname{rem}(\bar{f}, g)$

/12/ $\quad$ **end_while**

/13/ $\quad$ **if** $\bar{f} = f$ **then**

/14/ $\quad\quad \mathcal{U}^\triangle_\lambda \leftarrow \mathcal{U}^\triangle_\lambda \cup \{f^\sigma_{(k_\sigma)}\}$

/15/ $\quad$ **else**

/16/ $\quad\quad$ **if** $\bar{f} \neq 0$ **then**

/17/ $\quad\quad\quad \mathcal{U}^\triangle_\lambda \leftarrow \mathcal{U}^\triangle_\lambda \cup \{\bar{f}_{(\lambda)}\}$

/18/ $\quad\quad$ **end_if**

/19/ $\quad\quad$ **if** $k_\sigma < \lambda$ **then**

/20/ $\quad\quad\quad \mathcal{W}^\triangle_\lambda \leftarrow \mathcal{W}^\triangle_\lambda \cup \{f^\sigma_{(k_\sigma,\lambda)}\}$

/21/ $\quad\quad$ **end_if**

/22/ $\quad$ **end_if**

/23/ **end_while**

/24/ **return** $\mathcal{U}^\triangle_\lambda \cup \mathcal{W}^\triangle_\lambda$

The triangulation operation described in Algorithm 6.6 is slightly more complicated. Nevertheless, it is very similar to the autoreduction mentioned in Section 6.4. It suffices to reduce only the single indexed rows, as the multiplicative prolongations of the phantom rows obviously have different leading derivatives. In the outer `while` loop, we take a row f from $\mathcal{U}_\lambda$ with maximal leading derivative and try to reduce it modulo the rows contained in the system $\bar{\mathcal{S}}_\lambda$. Hence we check in Line /7/ whether a reduction with a row lying in one of the truncated involutive cones is possible. In the inner `while` loop, the actual reduction is performed. If no changes occur, the row is written back into the skeleton in Line /14/; otherwise the reduced row is added with initial level λ in Line /17/. If already multiplicative prolongations of the eliminated row have been computed, its removal would cause them to vanish as well. Thus we must keep track of their existence: in Line /20/ we turn the row into a phantom row, assuring that in future iterations its multiplicative prolongations are still taken into account.

The correctness of these algorithms is equivalent to showing that the system $\bar{\mathcal{S}}_\lambda^\triangle$ defined by the skeleton $\mathcal{S}_\lambda^\triangle$ yields a local representation for the differential equation $\mathcal{R}_{q+\lambda}$, if the initial differential equation was $\mathcal{R}_q$ (represented by the skeleton $\mathcal{S}_0$). As the main step towards this result we consider the relation between the prolonged rows of $\bar{\mathcal{S}}_\lambda^\triangle$ and rows in $\mathcal{S}_{\lambda+1}$.

Proposition 6.5.3. *Consider the sequence* (6.20) *derived from a skeleton* $\mathcal{S}_0$ *by repeatedly applying Algorithms 6.5 and 6.6. For* $\lambda \geq 0$ *let* g *be a linear combination of rows of the system* $\bar{\mathcal{S}}_\lambda^\triangle$*. Then each prolongation of* g *by one order can be written as a linear combination of rows of the system* $\bar{\mathcal{S}}_{\lambda+1}$*.*

Proof. We first slightly modify Algorithm 6.6 by replacing Line /20/ with $\mathcal{U}_\lambda^\triangle \leftarrow \mathcal{U}_\lambda^\triangle \cup \{f^\sigma_{(k_\sigma)}\}$. This implies that no phantom rows occur. So we ignore, for the moment, all references to the set $\mathcal{W}_\lambda$. Any row of which multiplicative prolongations have been computed is contained in all subsequent skeletons. Of course, one thereby loses the triangular form of the systems $\bar{\mathcal{S}}_\lambda^\triangle$.

Let the skeleton $\mathcal{S}_\lambda^\triangle$ consist of the single indexed rows $\{f^\sigma_{(k_\sigma)}\}$ and let

$$g = \sum_{l=1}^{s} a_l(x) D_{\mu_l} f^{\sigma_l} \tag{6.21}$$

be a finite linear combination of differential consequences of rows in the system $\bar{\mathcal{S}}_\lambda^\triangle$. The rows in the system $\bar{\mathcal{S}}_\lambda$ are multiplicative prolongations of rows in the corresponding skeleton $\mathcal{S}_\lambda$. We formally differentiate (6.21) with respect to an arbitrary independent variable x^j obtaining

$$D_j g = \sum_{l=1}^{s} \left[a_l(x) D_{\mu_l+1_j} f^{\sigma_l} + \frac{\partial a_l(x)}{\partial x^j} D_{\mu_l} f^{\sigma_l} \right] . \tag{6.22}$$

We must show that all the rows of $\bar{\mathcal{S}}_\lambda^\triangle$ *and* their prolongations by one order are elements of $\bar{\mathcal{S}}_{\lambda+1}$. By the remark above, this is obvious for the unprolonged rows. For the other rows, we proceed by an induction on the global level λ. For $\lambda = 0$, Algorithm 6.5 computes all non-multiplicative prolongations (since $k_\sigma = \lambda = 0$ for each row $f^\sigma_{(k_\sigma)}$). The multiplicative prolongations are taken care of by the incrementation of the global level λ. Now assume $\lambda > 0$. Let $h = D_{\mu+1_j} f$ be a prolongation with $D_\mu f \in \bar{\mathcal{S}}_\lambda^\triangle$ and

$f_{(k)} \in \mathcal{S}_\lambda^\triangle$. We prove that $h \in \bar{\mathcal{S}}_{\lambda+1}$ by a Noetherian argument. The terminal cases are:

- If x^j is a *multiplicative* variable for $f_{(k)}$, we are done: all differentiations in $D_{\mu+1_j}$ are multiplicative for f and, as the global level is raised by the prolongation, $h \in \bar{\mathcal{S}}_{\lambda+1}$.
- If $|\mu| = 0$ and x^j is a *non-multiplicative* variable for $f_{(k)}$, then the prolongation is explicitly computed and therefore contained in the next system.

In all other cases, we set $c = \operatorname{cls}\mu + 1_j$ and rewrite $h = D_c(D_{\mu+1_j-1_c}f)$. Because of $|\mu| \leq (\lambda - k)$, the total derivative in the parentheses is an element of $\bar{\mathcal{S}}_{\lambda'}$ for some $\lambda' \leq \lambda$. Thus it is also contained in $\bar{\mathcal{S}}_\lambda$ and we may apply the induction hypothesis in order to write h in the form

$$h = D_c\Bigl[\sum_{l=1}^{s'} b_l(x) D_{\nu_l} f^{\sigma_l}\Bigr] . \tag{6.23}$$

Observe that for the summand h' possessing the highest leading derivative (its multi index is $\mu+1_j-1_c$) the variable x^c is by construction multiplicative. Hence $D_c h' \in \bar{\mathcal{S}}_{\lambda+1}$ as desired. With each of the remaining summands, we repeat the above manipulations. As their leading exponents are all less than $\mu + 1_j - 1_c$, this process must terminate after a finite number of steps. This proves the statement above under the made assumption.

Finally, we include the phantom rows, i. e. we change back Line /20/ of Algorithm 6.6. It follows from (6.22) that if a row is dependent of the other rows in $\bar{\mathcal{S}}_\lambda$ and therefore eliminated, all its multiplicative prolongations by one order reduce to zero modulo $\bar{\mathcal{S}}_{\lambda+1}$. Therefore it does not matter for the system (however, it does matter for its triangular form) whether we carry out these reductions or not. Non-multiplicative prolongations of phantom rows never need to be computed, as for $f_{(k,\ell)}$ always $k < \ell \leq \lambda$ holds. □

Once this somewhat messy proof is finished, we obtain as corollary the for us crucial theorem on the relation between the sequence (6.20) obtained by alternately prolonging and triangulising the initial skeleton $\mathcal{S}_0$ and the prolongations of the linear differential equation $\mathcal{R}_q$ corresponding to $\mathcal{S}_0$.

Theorem 6.5.4. *Let the system $\mathcal{F}$ be a local representation for the linear differential equation $\mathcal{R}_q$. If we turn $\mathcal{F}$ into a skeleton $\mathcal{S}_0$ as described above and form the sequence* (6.20)*, then for all $\lambda \geq 0$ the system $\bar{\mathcal{S}}_\lambda^\triangle$ is triangulised and represents the equation $\mathcal{R}_{q+\lambda}$.*

Proof. We denote by $\mathcal{F}_0 = \mathcal{F}, \mathcal{F}_1, \mathcal{F}_2, \dots$ the prolongations of the system $\mathcal{F}$. Each $\mathcal{F}_i$ is a local representation for the prolonged differential equation $\mathcal{R}_{q+1}$. If we set $\mathcal{F}_0 = \bar{\mathcal{S}}_0^\triangle$, we have also equality of the linear hulls: $\langle \mathcal{F}_0 \rangle = \langle \bar{\mathcal{S}}_0^\triangle \rangle$.

By repeatedly applying Proposition 6.5.3, we get for $\lambda \geq 0$: $\langle \mathcal{F}_\lambda \rangle = \langle \bar{\mathcal{S}}_\lambda \rangle$. As we perform only elementary row operations during Algorithm 6.6, it follows that $\bar{\mathcal{S}}^{\triangle}_\lambda$ is a representation of $\mathcal{R}_{q+\lambda}$.

It remains to show is that the systems $\bar{\mathcal{S}}^{\triangle}_\lambda$ are really in triangular form. First of all, the rows in $\mathcal{U}^{\triangle}_\lambda$ obviously possess different leading derivatives. Now suppose that there are two rows $f^1, f^2 \in \bar{\mathcal{S}}^{\triangle}_\lambda$ with $\operatorname{ld} D_{\mu_1} f^1 = \operatorname{ld} D_{\mu_2} f^2$. Let $f^1_{(k_1,\ell_1)}$ and $f^2_{(k_2,\ell_2)}$ be the corresponding rows in the skeleton $\mathcal{S}^{\triangle}_\lambda$ from which they have been derived by multiplicative prolongations. If either f^1 or f^2 is a single indexed row, we simply ignore all references made to the phantom level.

By definition of an involutive division, if two involutive cones intersect, one is entirely contained in the other one. Thus without loss of generality, we assume $\operatorname{ld} f^2 = \operatorname{ld} D_\nu f^1$ for some multiplicative prolongation D_ν. This implies that in step $k_1 + |\nu|$ the row f^2 was reduced and became a phantom row. In the next $|\mu_2|$ steps, all multiplicative prolongations of f^2 up to this order are reduced. From $|\nu| + |\mu_2| = |\mu_1| \leq (\lambda - k_1)$ it follows that $D_{\mu_2} f^2 \notin \bar{\mathcal{S}}^{\triangle}_\lambda$, and we have arrived at a contradiction. □

By working with skeletons instead of full systems, we need to compute the multiplicative prolongations only when they are required for a reduction. In each step of the algorithm, we prolong the skeleton by exactly one order and compute a triangular form. However, as we want to realise the Cartan–Kuranishi completion, we have to relate these ideas to the geometric theory.

We have already seen that the skeleton $\mathcal{S}^{\triangle}_\lambda$ yields a representation for the differential equation $\mathcal{R}_{q+\lambda}$. In fact, it contains even more information. Projecting into jet bundles of lower order, one can extract from $\mathcal{S}^{\triangle}_\lambda$ representations for all differential equations $\mathcal{R}^{(s)}_{q+r}$ with $\lambda = r + s$:

$$\mathcal{S}^{\triangle}_\lambda : \begin{cases} \mathcal{R}_{q+\lambda} \xrightarrow{\pi^{q+\lambda}_{q+\lambda-1}} \mathcal{R}^{(1)}_{q+\lambda-1} \xrightarrow{\pi^{q+\lambda-1}_{q+\lambda-2}} \mathcal{R}^{(2)}_{q+\lambda-2} \xrightarrow{\pi^{q+\lambda-2}_{q+\lambda-3}} \cdots \end{cases} . \tag{6.24}$$

Because of the triangular form of $\bar{\mathcal{S}}^{\triangle}_\lambda$, this amounts to dropping the rows of appropriate orders, although we will never do this explicitly. Instead, we identify the differential equation $\mathcal{R}^{(s)}_{q+r}$ currently considered by the values of the parameters r and s.

If we juxtapose all the sequences of the form (6.24), we arrive at the grid pictured in Figure 6.3. Starting with $\mathcal{R}_q$, the Cartan–Kuranishi algorithm moves along the horizontal lines searching for an equation with involutive symbol. If one is found, it is checked for integrability conditions by comparing it with the equation one row below and one column to the right. If the two equations differ, the process is continued in the row below, i. e. after adding the integrability conditions. Recall that in general $\big(\mathcal{R}^{(s)}_{q+r}\big)_{+1} \neq \mathcal{R}^{(s)}_{q+r+1}$;

$$\begin{tikzcd}
\mathcal{S}_0^{\triangle} \arrow[r, Rightarrow] \arrow[d, dash] & \mathcal{S}_1^{\triangle} \arrow[r, Rightarrow] \arrow[d, dash] & \mathcal{S}_2^{\triangle} \arrow[r, Rightarrow] \arrow[d, dash] & \cdots \\
\mathcal{R}_q \arrow[r] \arrow[d, dash] \arrow[dr, equal, "?"] & \mathcal{R}_{q+1} \arrow[r] \arrow[d, dash] \arrow[dr, equal, "?"] & \mathcal{R}_{q+2} \arrow[r] \arrow[d, dash] & \cdots \\
\mathcal{R}^{(1)}_{q-1} \arrow[r, dashed] \arrow[d, dash] \arrow[dr, equal, "?"] & \mathcal{R}^{(1)}_{q} \arrow[r, dashed] \arrow[d, dash] \arrow[dr, equal, "?"] & \mathcal{R}^{(1)}_{q+1} \arrow[r, dashed] \arrow[d, dash] & \cdots \\
\mathcal{R}^{(2)}_{q-2} \arrow[r, dashed] \arrow[d, dash] & \mathcal{R}^{(2)}_{q-1} \arrow[r, dashed] \arrow[d, dash] & \mathcal{R}^{(2)}_{q} \arrow[r, dashed] \arrow[d, dash] & \cdots \\
\vdots & \vdots & \vdots &
\end{tikzcd}$$

Fig. 6.3. Skeletons and the corresponding equations

to ensure equality we need an involutive symbol (cf. Part (ii) of Proposition 3.1.10). So two tasks remain to be considered:

1. provide a method for deciding which of the differential equations represented by a skeleton have involutive symbols;
2. determine whether $\mathcal{R}^{(s)}_{q+r}$ and $\mathcal{R}^{(s+1)}_{q+r}$ are equal.

The first one is solved by the following result.

Proposition 6.5.5. *Let $\mathcal{S}^{\triangle}_{\lambda}$ be a skeleton for the differential equation $\mathcal{R}_{q+\lambda}$ in δ-regular coordinates and let t denote the maximal order of a single indexed row in $\mathcal{S}^{\triangle}_{\lambda+1}$. The symbol $\mathcal{M}^{(s)}_{q+r}$ is involutive for all values r, s such that $r+s=\lambda$ and $q+r\geq t$.*

Proof. We choose r,s with $r+s=\lambda$. A representation for $\mathcal{R}^{(s)}_{q+r}$ can be obtained from $\bar{\mathcal{S}}^{\triangle}_{\lambda}$ by dropping all rows of order greater than $q+r$. We prolong this system by one order and compute a triangular form. If no non-multiplicative prolongations of order $q+r+1$ remain, Proposition 3.1.4 can be invoked to prove that the symbol $\mathcal{M}^{(s)}_{q+r}$ is involutive. By assumption, even after the prolongation and subsequent triangulation of the full skeleton no single indexed rows of this or a greater order exist in $\mathcal{S}^{\triangle}_{\lambda+1}$. As an independent non-multiplicative prolongation would result in such a row, the involution of the examined symbol can be read off from this skeleton. □

A useful side effect of this result is that it allows us to carry out additional projections while working inside a given skeleton: we can trace a vertical line in Figure 6.3 belonging to some skeleton $\mathcal{S}_\lambda^\triangle$ downwards as long as the symbols of the respective differential equations remain involutive and continue with the last such equation. This is shown in Figure 6.4 in comparison to the strategy followed in the generic version of the Cartan–Kuranishi completion (boxed equations possess involutive symbols).

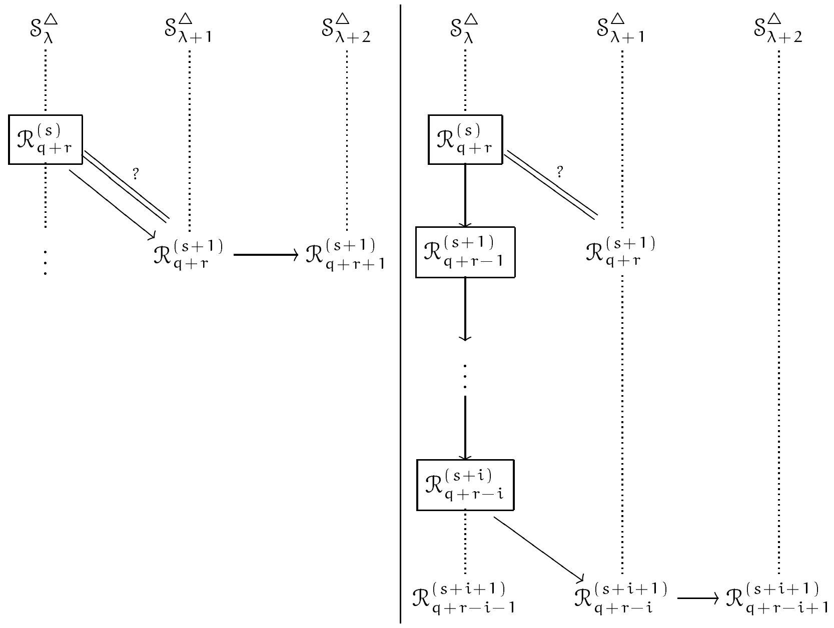

Fig. 6.4. Analysis of symbols and systems (left: classical Cartan–Kuranishi algorithm; right: new algorithm).

For the second task mentioned above, we note that it suffices to consider the dimensions of the involved manifolds, as one is always a submanifold of the other. As the triangulation guarantees algebraic independence of the rows in a skeleton, the determination of dimensions reduces to a simple counting of equations.

Definition 6.5.6. *Let* $\mathcal{S}_\lambda^\triangle$ *be a triangulised skeleton. We denote by* $\#\mathcal{S}_{\lambda,t}^\triangle$ *the number of rows of order* t *in the system* $\bar{\mathcal{S}}_\lambda^\triangle$. *The* rank vector *of the skeleton is the list*

$$\mathbf{r}_{\mathcal{S}_\lambda^\triangle} = [\#\mathcal{S}_{\lambda,0}^\triangle, \dots, \#\mathcal{S}_{\lambda,q+\lambda}^\triangle] \tag{6.25}$$

where q *is the order of the original skeleton* $\mathcal{S}_0$.

The numbers $\#\mathcal{S}^{\triangle}_{\lambda,t}$ yield straightforwardly the dimensions of all differential equations and symbols represented by $\mathcal{S}^{\triangle}_{\lambda}$. For $\lambda = r + s$

$$\dim \mathcal{R}^{(s)}_{q+r} = \dim J_{q+r}\mathcal{E} - \sum_{i=0}^{q+r} \#\mathcal{S}^{\triangle}_{\lambda,i} \ , \tag{6.26a}$$

$$\dim \mathcal{M}^{(s)}_{q+r} = \dim S_q(T^*\mathcal{X}) \underset{J_{q-1}\mathcal{E}}{\otimes} V\mathcal{E} - \#\mathcal{S}^{\triangle}_{\lambda,q+r} \ . \tag{6.26b}$$

The two differential equations $\mathcal{R}^{(s)}_{q+r}$ and $\mathcal{R}^{(s+1)}_{q+r}$ are now identical, i. e. no integrability conditions appear during the next prolongation, if the rank vectors $\mathbf{r}_{\mathcal{S}^{\triangle}_{\lambda}}$ and $\mathbf{r}_{\mathcal{S}^{\triangle}_{\lambda+1}}$ coincide in their first $q + r$ entries. The higher entries play no role.

It is straightforward to modify the Algorithms 6.5 and 6.6 so that they update the rank vector. Computing it for the initial skeleton is trivial. During the prolongation from $\mathcal{S}_\lambda$ to $\mathcal{S}_{\lambda+1}$ the following corrections are necessary for each row $f \in \mathcal{S}_\lambda$:

- If $f_{(k)}$ is a single indexed row of order t with level $k = \lambda$, prolongations with respect to all independent variables are computed. Hence the entry for order $t + 1$ is increased by n, the number of independent variables.
- If the initial level of $f_{(k)}$ with order t and class c is less than λ, $\binom{\lambda-k+c}{c-1}$ new rows of order $t + \lambda - k + 1$ enter the system.
- If $f_{(k,\ell)}$ is a double indexed row of order t and class c, there are $\binom{\lambda-k+c}{c-1}$ new rows of order $t + \lambda - k + 1$, and $\binom{\lambda-\ell+c}{c-1}$ rows of order $t + (\lambda - \ell) + 1$ are removed.

For a triangulation, the corrections are even simpler: if a row is reduced to a new row of lower order, the entry in the rank vector for the old order has to be decreased and the one for the new order has to be increased by one.

Combining all these ideas and results we finally arrive at the promised hybrid Algorithm 6.7. In contrast to the Cartan–Kuranishi Algorithm 6.1, it consists of only one loop in which the skeleton is prolonged and triangulised (Lines /8/ and /9/) using the above described algorithms. The analysis of the symbol based on Proposition 6.5.5 takes place in Line /10/. A simple comparison of rank vectors in Line /13/ signals whether integrability conditions have appeared. Note that we compare the entries of the vectors only up to the currently relevant order. The `while` loop in Lines /18–21/ realises the idea sketched in Figure 6.4 and projects as far as possible.

Strictly speaking, this is *not* a realisation of the Cartan–Kuranishi completion because of the inner `while` loop performing additional projections. A strict realisation would allow only one iteration through this loop. In practice, these additional projections are, however, a benefit. Of course, one is

Algorithm 6.7 A Hybrid Completion Algorithm for Linear Systems

Input: skeleton $\mathcal{S}_0^{\triangle}$ of the differential equation $\mathcal{R}_q$
Output: skeleton $\mathcal{S}_\lambda^{\triangle}$ and integers $r, s \in \mathbb{N}$ such that $\mathcal{R}_{q+r}^{(s)}$ is involutive

$r \leftarrow 0$
$s \leftarrow 0$
$\lambda \leftarrow 0$
systemInvolutive $\leftarrow$ *FALSE*
repeat
  $\bar{\mathbf{r}} \leftarrow \mathbf{r}_{\mathcal{S}_\lambda^{\triangle}}$
  $\lambda \leftarrow \lambda + 1$
  $\mathcal{S}_\lambda \leftarrow$ `ProlongSkeleton`$(\mathcal{S}_{\lambda-1}^{\triangle})$
  $\mathcal{S}_\lambda^{\triangle} \leftarrow$ `TrianguliseSkeleton`$(\mathcal{S}_\lambda)$
  if $\exists f_{(k)} \in \mathcal{U}_\lambda^{\triangle} : \operatorname{ord} f_{(k)} = q + r + 1$ **then** { *symbol not involutive* }
    $r \leftarrow r + 1$
  else { $\mathcal{M}_{q+r}^{(s)}$ *involutive; check formal integrability of* $\mathcal{R}_{q+r}^{(s)}$ }
    if $\forall 0 \leq i \leq q + r : \mathbf{r}_{\mathcal{S}_\lambda^{\triangle}}[i] = \bar{\mathbf{r}}[i]$ **then**
      systemInvolutive $\leftarrow$ *TRUE*
    else
      $r \leftarrow r + 1$
    end_if
    while $\neg\big(\exists f_{(k)} \in \mathcal{U}_\lambda^{\triangle} : \operatorname{ord} f_{(k)} = q + r\big)$ **do** { *project as far as possible* }
      $r \leftarrow r - 1$
      $s \leftarrow s + 1$
    end_while
  end_if
until *systemInvolutive*
return $(r, s, \mathcal{S}_\lambda^{\triangle})$

interested in those values r, s for which $\mathcal{R}_{q+r}^{(s)}$ is involutive *and* the sum $r+s$ is minimal. Our new algorithm delivers these values.

Theorem 6.5.7. *If the coordinates x are δ-regular for all appearing symbols, Algorithm 6.7 terminates with an involutive system.*

Proof. This follows immediately from the Cartan–Kuranishi Theorem 6.2.1. The only problem for the termination is that at same stage we might fail to recognise that the current symbol is involutive. In this case, Algorithm 6.7 would prolong infinitely often. But if the used coordinate system is δ-regular for each appearing symbol, involution of the symbols is easily checked by Proposition 6.5.5. □

Theorem 2.4.10 allows us to verify the condition of Theorem 6.5.7. If the used coordinates are not δ-regular for one of the symbols, then at some stage of the completion a larger Janet span appears. Thus with the help of this simple criterion, we can effectively avoid an infinite computation.

Given the triangulised skeleton $\mathcal{S}_\lambda^\triangle$ of a differential equation $\mathcal{R}_q$, we refine the above considerations about the dimensions in order to determine the indices $\beta_q^{(i)}$ of its symbol $\mathcal{M}_q$. A row $f_{(k,\ell)} \in \mathcal{S}_\lambda^\triangle$ of order t and class c obviously contributes to them, if and only if (i) $\lambda - k + t \geq q$ and (ii) $\ell = 0$ or $\lambda - \ell + t < q$. Its contribution $\beta_q^{(i)}(f_{(k,\ell)})$ is then given by

$$\beta_q^{(i)}(f_{(k,\ell)}) = \begin{cases} B(c-i+1, q-t, 1) & \text{for } 1 \leq i \leq c\,, \\ 0 & \text{for } c < i \leq n \end{cases} \tag{6.27}$$

where $B(n,q,c) = \binom{n-c+q-1}{q-1}$ denotes the number of multi indices of length n, order q and class c. The Cartan characters $\alpha_q^{(i)}$ of the equation are then computed from (4.6).

Example 6.5.8. We demonstrate the working of Algorithm 6.7 for the Janet Example 6.2.2. Our algorithm starts with the skeleton

$$\mathcal{S}_0 = \mathcal{S}_0^\triangle : \begin{cases} (u_{zz} + yu_{xx})_{(0)}\,, \\ (u_{yy})_{(0)}\,. \end{cases} \tag{6.28}$$

The initial rank vector is $\mathbf{r}_{\mathcal{S}_0^\triangle} = [0,0,2]$ and the algorithm proceeds as follows.

1. *Iteration*: There is only one non-multiplicative prolongation: u_{yyz}. It cannot be reduced and is added to the skeleton with initial level 1. Since $\mathcal{S}_1^\triangle$ contains a third-order row, the symbol $\mathcal{M}_3$ is not involutive.
2. *Iteration*: u_{yyzz} is the only new non-multiplicative prolongation and it reduces to u_{xxy}. As no row of order 4 remains in the skeleton, $\mathcal{M}_4$ is involutive. We must now compare the rank vectors up to order 3: $\bar{\mathcal{S}}_1^\triangle$ contains 6 equations of order 3, whereas $\bar{\mathcal{S}}_2^\triangle$ contains 7. Thus one integrability condition has appeared (which is of course u_{xxy}) and the algorithm continues with the equation $\mathcal{R}_3^{(1)}$.
3. *Iteration*: The computed non-multiplicative prolongations are u_{xxyy} and u_{xxyz}. Only the first row can be reduced to zero, while the second one obstructs the involution of $\mathcal{M}_3^{(1)}$.
4. *Iteration*: Again there are two non-multiplicative prolongations: u_{xxyyz} can be reduced to zero; u_{xxyyz} is replaced by u_{xxxx}. Due to the lack of an equation of order 5 in $\mathcal{S}_4^\triangle$, the symbol $\mathcal{M}_4^{(1)}$ is involutive. The rank vectors that must be compared are: $\mathbf{r}_{\mathcal{S}_3^\triangle} = [0,0,2,7,13,17]$ and $\mathbf{r}_{\mathcal{S}_4^\triangle} = [0,0,2,7,14,19,24]$. Only the first five components are taken into account, but the difference between the entries 13 and 14 signals the appearance of another integrability condition (u_{xxxx}). The next equation to be examined is $\mathcal{R}_4^{(2)}$.

5. *Iteration*: Of the two non-multiplicative prolongations, u_{xxxxy} is reduced to zero, whereas u_{xxxxz} represents an obstruction to involution for the symbol $\mathcal{M}_4^{(2)}$. The current skeleton is now

$$\mathcal{S}_5^{\triangle} : \begin{cases} (u_{xxxxz})_{(5)}\,, & (u_{yyz})_{(1)}\,, \\ (u_{xxxx})_{(4)}\,, & (u_{zz} + yu_{xx})_{(0)}\,, \\ (u_{xxyz})_{(3)}\,, & (u_{yy})_{(0)}\,, \\ (u_{xxy})_{(2)} \end{cases} \tag{6.29}$$

with rank vector $\mathbf{r}_{\mathcal{S}_5^{\triangle}} = [0,0,2,7,14,21,26,32]$.

6. *Iteration*: As both non-multiplicative prolongations u_{xxxxyz} and u_{xxxxzz} can be reduced to zero, the symbol $\mathcal{M}_5^{(2)}$ is involutive. The comparison of the rank vectors shows that no new integrability conditions have appeared. Thus the algorithm stops and outputs the skeleton $\mathcal{S}_6^{\triangle}$, which is identical with $\mathcal{S}_5^{\triangle}$ given above, for the differential equation $\mathcal{R}_5^{(2)}$. A local representation is obtained by prolonging all equations in the skeleton multiplicatively up to fifth order.

Thus the path of the Cartan–Kuranishi completion can be summarised in the following sequence of differential equations:

$$\mathcal{R}_2 \to \mathcal{R}_3 \to \mathcal{R}_3^{(1)} \to \mathcal{R}_4^{(1)} \to \mathcal{R}_4^{(2)} \to \mathcal{R}_5^{(2)}\,. \tag{6.30}$$

In order to demonstrate the superior efficiency compared to a direct implementation of the Cartan–Kuranishi completion we do some statistics. The new algorithm computes 10 non-multiplicative and 18 multiplicative prolongations (for the reductions); 5 of the latter ones are used several times. In contrast, a full determination of all systems requires the prolongation of 132 equations — almost five times more. In addition, the naive approach requires to determine row echelon forms for a number of matrices (Jacobians and symbols), the largest being an 86×84 matrix. $\triangleleft$

One may wonder why we need rank vectors, as all integrability conditions in the Janet example stem from non-multiplicative prolongations that do not reduce to zero. This is, however, only one type of integrability condition; there exists a second type for systems comprising equations of different orders. As a trivial example we consider the equation $\mathcal{R}_2$ locally represented by $u_{yy} = u_{xy} = u_x = 0$. Obviously, the symbol $\mathcal{M}_2$ is involutive. But $\mathcal{R}_2$ is not formally integrable: prolongation of the lower order equation $u_x = 0$ with respect to the *multiplicative* variable x yields the additional equation $u_{xx} = 0$ required to represent $\mathcal{R}_2^{(1)}$.

In our algorithm the skeletons $\mathcal{S}_0^{\triangle}$ and $\mathcal{S}_1^{\triangle}$ contain exactly the same rows. But their rank vectors differ: $\mathbf{r}_{\mathcal{S}_0^{\triangle}} = [0,1,2]$ and $\mathbf{r}_{\mathcal{S}_1^{\triangle}} = [0,1,3,3]$. The increase in the third entry is due to the integrability condition $u_{xx} = 0$. Our algorithm never explicitly determines this type of integrability conditions; they are solely visible through changes in the rank vector.

In the Janet example, phantom rows do not occur and the projections are always only by one order. In the next example, stemming from Lie symmetry theory [20, 129], both phenomena are present.

Example 6.5.9. We consider the determining system for the generators of Lie point symmetries of the heat equation. It consists of nine equations in three dependent variables η, ξ, τ and three independent variables t, x, u.

$$\mathcal{R}_2 : \begin{cases} \tau_u = 0 , & \tau_{ux} + \xi_u = 0 , & \eta_{uu} - \xi_{ux} = 0 , \\ \tau_x = 0 , & \tau_{xx} + 2\xi_x - \tau_t = 0 , & \eta_{xx} - \eta_t = 0 , \\ \tau_{uu} = 0 , & \eta_{ux} - \frac{1}{2}\xi_{xx} + \frac{1}{2}\xi_t = 0 , & \xi_{uu} = 0 . \end{cases} \tag{6.31}$$

As above, we obtain $\mathcal{S}_0^{\triangle}$ by dropping the right hand side of each equation and assigning an initial level of 0. During the triangulation of $\mathcal{S}_1$, the row $\tau_{xx} + 2\xi_x - \tau_t$ can be reduced by the (multiplicative) prolongation with respect to x of the row τ_x. This yields the row $\xi_x - \frac{1}{2}\tau_t$, which is added to $\mathcal{S}_1^{\triangle}$. However, the system it defines also contains multiplicative prolongations of $\tau_{xx} + 2\xi_x - \tau_t$. Thus dropping the row completely would lose information and lead to wrong values for the dimensions. Instead, its phantom level is set to 1. In the λth iteration, all rows obtained by prolonging λ times the row $\tau_{xx} + 2\xi_x - \tau_t$ are still present in $\bar{\mathcal{S}}_\lambda$ and available for reducing other rows.

The algorithm continues to produce the following sequence of equations (note the double projection in the third iteration step):

$$\mathcal{R}_2 \to \mathcal{R}_3 \to \mathcal{R}_3^{(1)} \to \mathcal{R}_2^{(3)} \to \mathcal{R}_2^{(4)} \to \mathcal{R}_3^{(4)} . \tag{6.32}$$

The final skeleton is

$$\mathcal{S}_5^{\triangle} : \begin{cases} \tau_{ttt} = 0 , & \eta_{xx} - \eta_t = 0 , & \xi_u = 0 , \\ \eta_{uu} - \xi_{ux} = 0 , & \xi_{tt} = 0 , & \xi_x - \frac{1}{2}\tau_t = 0 , \\ \eta_{ut} + \frac{1}{2}\xi_{xt} = 0 , & \tau_u = 0 , & \tau_x = 0 , \\ \eta_{ux} - \frac{1}{2}\xi_{xx} + \frac{1}{2}\xi_t = 0 . \end{cases} \tag{6.33}$$

Since only head reductions have been performed during the algorithm, there exist further simplifications. $\lhd$

6.6 Applications I: Differentiation and Perturbation Indices

In the numerical theory of differential algebraic equations (see e. g. [25, 68]) the concept of an *index*[5] plays an important role. It serves as an indicator

[5] Not to be confused with the indices $\beta_q^{(k)}$ of the symbol!

for the difficulties one has to expect, if one tries to numerically solve the given differential equation. The higher the index of an equation, the more problems arise in its integration. Typical problems are that classical methods exhibit suddenly a lower order when applied to non-normal systems or that constraints are violated, the so-called drift off the constraint manifold.

The literature abounds with definitions of indices; a partial survey may be found in [30]. A recent trend is the extension of results on differential algebraic equations to non-normal systems of partial differential equations. One speaks then often of *partial differential algebraic equations*, but this terminology is somewhat misleading. In many important cases like the Maxwell or Navier–Stokes equations, such systems do not contain any algebraic equations; the non-normality is due to the presence of equations of lower class not of lower order. We will therefore avoid this term.

In this section we show that many of the proposed index concepts can be well understood within the framework of the completion to involution. In particular, we provide intrinsic formulations of them applicable to partial differential equations, too. In contrast to most proposals in the literature, our definitions are directly based on the partial differential equations and do not require the use of integral transformations, semi-discretisations or similar means of reduction to differential algebraic equations.

We distinguish two main classes of indices. *Differentiation indices* count the number of prolongations needed until the equation satisfies some condition. In the case of "the" differentiation index, the condition is that the equation is not underdetermined. *Perturbation indices* are based on estimates on the difference between solutions of the original equation and of a perturbation of it. Essentially, the question is up to which order derivatives of the perturbation enter the estimate.

The most widely used index for differential algebraic equations is "the" differentiation index usually defined via the so-called derivative arrays. The rth array consists of a local representation of the prolonged equation $\mathcal{R}_{q+r}$. The classical approach to "the" differentiation index of a first-order ordinary differential equation $\mathcal{R}_1$ defines it as the smallest number ν such that all derivatives $\dot{\mathbf{u}}$ can be determined as functions of the independent variable t and the dependent variables $\mathbf{u}$. In the language of Section 6.1 we may say that then the differential part of the system yields an underlying equation. In an intrinsic manner we define it as follows.

Definition 6.6.1. *The* determinacy index *of a differential equation* $\mathcal{R}_q \subseteq J_q\mathcal{E}$ *in* n *independent and* m *dependent variables is the smallest integer* $\nu_d = r + s$ *such that* $\beta^{(n)}_{q+r} = m$ *for the differential equation* $\mathcal{R}^{(s)}_{q+r}$.

In order to avoid writing "the" differentiation index all the time, we have coined the new term determinacy index, as the defining property of this index is that $\mathcal{R}^{(s)}_{q+r}$ is not an underdetermined equation. ν_d has no finite value for an underdetermined equation, as then by Proposition 3.3.7 $\beta^{(n)}_{q+r} < m$ for any value of r and s. Note that, as promised, it makes no difference whether we analyse an ordinary or a partial differential equation.

Remark 6.6.2. The straightforward translation of the classical definition of the determinacy index into our geometric language would be to require that the symbol $\mathcal{M}^{(s)}_{q+r}$ vanishes, i. e. has dimension zero. For ordinary differential equations this is equivalent to Definition 6.6.1, as here $\dim \mathcal{M}^{(s)}_{q+r} = m - \beta^{(1)}_{q+r}$. However, for partial differential equations this approach yields a finite value only for equations of finite type, as only for these all derivatives of highest order can be expressed as functions of the remaining ones. Obviously, such a definition is not very useful in practice. ◁

The determinacy index cannot be used as a basis of an existence theory for solutions. It is automatically determined during the completion of the original equation $\mathcal{R}_q$, but does not require the full completion. Thus it is well possible that the equation is in fact inconsistent. We will see below in our discussion of the perturbation index, why the determinacy index is nevertheless a useful concept.

From the point of view of the completion theory, Definition 6.6.1 appears to be a bit strange: why stop in the middle of the completion? Thus a more natural index concept is simply to count the number of prolongations needed until we reach an involutive equation. In the case of differential algebraic equations this approach is in fact equivalent to the *strangeness index* introduced by Kunkel and Mehrmann [100, 101] and to the geometric index definitions in [144, 146, 175].

Definition 6.6.3. *The* involution index *of the differential equation* $\mathcal{R}_q$ *is the smallest integer* $\nu_i = r + s$ *such that* $\mathcal{R}^{(s)}_{q+r}$ *is involutive.*

Of course, one could similarly define an *integrability index*, as the smallest value $r+s$ such that $\mathcal{R}^{(s)}_{q+r}$ is formally integrable. But we will see later that involution is more important in this context. Obviously, we have $\nu_d \leq \nu_i$

The perturbation index was first introduced by Hairer et al. [67] for differential algebraic equations and later extended to partial differential equations by Campbell and Marszalek [31]. It is an analytic concept and thus we formulate it only for systems. We assume that $\mathcal{X} \subset \mathbb{R}^n$ is a compact domain and $\mathcal{E} = \mathbb{R}^m \times \mathbb{R}$. We choose a vector norm $|\cdot|$ on $\mathbb{R}^m$ and a norm $\|\cdot\|$ on some function space $\mathcal{F}$ which is usually either the maximum norm $\|\cdot\|_{L^\infty}$ or the uniform norm $\|\cdot\|_{L^1}$. In addition, we define on $\mathcal{F}$ for each integer $k > 0$ a kind of Sobolev norm

$$\|f\|_k = \sum_{0\leq|\mu|\leq k} \left\| \frac{\partial^{|\mu|} f}{\partial x^\mu} \right\| , \tag{6.34}$$

i. e. we sum the norms of f and all its partial derivatives up to order k. Of course, the function space $\mathcal{F}$ must be chosen such that these norms make sense on it.

Partial differential equations are usually accompanied by initial or boundary conditions. In order to accommodate for this, we take the following simple approach. Let $\mathcal{X}' \subseteq \partial\mathcal{X}$ be a subdomain of the boundary of $\mathcal{X}$ and introduce on the restriction $\mathcal{F}'$ of $\mathcal{F}$ to $\mathcal{X}'$ similar norms denoted by $\|\cdot\|'_k$. The conditions are then written in the form $\mathbf{\Psi}(x, \mathbf{p})\big|_{\mathcal{X}'} = 0$. This comprises most kinds of initial or boundary conditions found in applications. The highest derivative in $\mathbf{\Psi}$ determines the order ℓ of the conditions.

We consider a general system of partial differential equations

$$\mathbf{\Phi}(x, \mathbf{p}) = 0 \tag{6.35}$$

where $\mathbf{\Phi}$ takes its values in $\mathbb{R}^t$. As usually, we do not require that $t = m$ or that the Jacobian $\partial\mathbf{\Phi}/\partial\mathbf{p}$ has any special properties. We assume that we are given a smooth solution $\mathbf{u}(x)$ of it defined on the whole domain $\mathcal{X}$ and satisfying our initial or boundary conditions on $\mathcal{X}'$. We want to compare it with solutions of the perturbed equation

$$\mathbf{\Phi}(x, \mathbf{p}) = \boldsymbol{\delta}(x) \tag{6.36}$$

with a smooth right hand side $\boldsymbol{\delta}$.

Definition 6.6.4. *Let $\mathbf{u}(x)$ be a smooth solution of* (6.35) *satisfying some initial or boundary conditions of order ℓ on $\mathcal{X}'$. The system* (6.35) *has* perturbation index ν_p *along this solution, if ν_p is the smallest integer such that for any solution $\tilde{\mathbf{u}}(x)$ of the perturbed equation* (6.36) *at every point $x \in \mathcal{X}$ the estimate*

$$|\mathbf{u}(x) - \tilde{\mathbf{u}}(x)| \leq C\big(\|f - \tilde{f}\|'_\ell + \|\boldsymbol{\delta}\|_{\nu_p - 1}\big) \tag{6.37}$$

holds, whenever the right hand side is sufficiently small. Here f and $\tilde{f}$ represent the restrictions of the solutions $\mathbf{u}$ and $\tilde{\mathbf{u}}$, respectively, to $\mathcal{X}'$. The constant C may depend only on the domains $\mathcal{X}$, $\mathcal{X}'$ and on the function $\mathbf{\Phi}$.

In the case of initial value problems for differential algebraic equations this definition coincides with the one given by Hairer et al. [67]. The term

$\|f - \tilde{f}\|'_k$ takes the simple form $|\mathbf{u}(x_0) - \tilde{\mathbf{u}}(x_0)|$ where $\mathcal{X}' = \{x_0\}$. For partial differential equations our definition is almost identical with the one given by Campbell and Marszalek [31]; the only difference is that we do not include the order ℓ of the initial or boundary conditions in the index.

Whereas it is not so clear why differentiation indices should indicate the difficulty of the numerical integration of (6.35), this is rather obvious for the perturbation index. If we take for $\tilde{\mathbf{u}}(x)$ an approximate solution of (6.35), we may interpret $\boldsymbol{\delta}(x)$ as the residual obtained by entering it into the equation. (6.37) tells us that for an equation with $\nu_p > 1$ it does not suffice to keep this residual as small as possible, since also some of its derivatives enter the estimate. Strictly speaking, this implies that the considered initial or boundary value problem is ill-posed in the sense of Hadamard!

Unfortunately, it is rather difficult to derive estimates of the form (6.37), especially for partial differential equations. On the other hand, it is comparatively straightforward to compute the determinacy or the involution index. Thus it is a natural thought to bound the perturbation index by one of these. However, Campbell and Gear [30] showed that, in general, no relation between ν_d, ν_i and ν_p exists. This is not a surprise, as ν_p is based on the perturbed equation (6.36) and the perturbation may affect the completion for non-linear equations.

Example 6.6.5. Consider the following simple quasi-linear system of ordinary differential equations for two unknowns u and v:

$$vv' + u = 0\,, \qquad v = 0\,. \tag{6.38}$$

Obviously it is equivalent to $u = v = 0$. Thus its determinacy and involution indices are $\nu_d = \nu_i = 1$, respectively, as after one step we obtain an underlying equation $u' = v' = 0$.

For the perturbed system

$$vv' + u = \delta\,, \qquad v = \epsilon\,, \tag{6.39}$$

the situation is different, as we cannot eliminate the derivative v' by algebraic manipulations. The first completion step yields $v' = \epsilon'$ and only now the first equation of (6.39) becomes equivalent to $u = \delta - \epsilon\epsilon'$. The second completion step leads to $u' = \delta' - (\epsilon\epsilon')'$ and thus an underlying equation. The indices of the perturbed system are $\tilde{\nu}_d = \tilde{\nu}_i = 2$. From the solution of the system we easily read off that also $\nu_p = 2$, as u depends on ϵ'.

As shown in [30], this example can be extended. Using an m-dimensional vector of unknowns $\mathbf{u}$, one constructs systems such that for the unperturbed system still $\nu_d = \nu_i = 1$ but for the perturbed one $\tilde{\nu}_d = \tilde{\nu}_i = m$ and also $\nu_p = m$. Thus the difference between the indices of the unperturbed and the perturbed system may become arbitrarily large. ◁

Definition 6.6.6. *Let $\mathbf{\Phi}(\mathbf{x}, \mathbf{p}) = 0$ be a system of differential equations. Its* perturbed determinacy index $\tilde{\nu}_d$ *and* perturbed involution index $\tilde{\nu}_i$*, respectively, are defined as the indices of the perturbed system $\mathbf{\Phi}(\mathbf{x}, \mathbf{p}) = \boldsymbol{\delta}(\mathbf{x})$.*

Note that these perturbed indices are not intrinsically defined for differential equations $\mathcal{R}_q$ but only for systems. The reason lies simply in the fact that our approach to perturbing the equation is based on a fixed local representation. This approach also reflects the situation appearing in concrete numerical computations.

Remark 6.6.7. Definition 6.6.6 implicitly assumes that the completion process of the perturbed equation is independent of the precise form of $\boldsymbol{\delta}$. In most applications this is probably the case. Otherwise the perturbed indices are computed with the generic branch of the completion, i. e. under the assumption that the perturbations do *not* satisfy any special conditions. $\lhd$

We show now that the perturbation index may be estimated by the perturbed determinacy index. We do this first for the case of ordinary differential equations, where we can obtain very general results. For partial differential equations we need to make a number of assumptions in order to prove a rigorous theorem.

Before we get to our main theorem, we need a proposition which is a simple consequence of the *Gronwall Lemma.* Recall that this lemma states that if a continuous function $\psi : [a, b] \to \mathbb{R}$ satisfies $\psi(x) \leq A + B\int_a^x \psi(\xi)d\xi$ for some constants $A, B \geq 0$, then the estimate $\psi(x) \leq Ae^{B(x-a)}$ holds for all $x \in [a, b]$. A proof of it is contained in almost any textbook on ordinary differential equations (see e. g. [188]). Based on it we can compute the perturbation index ν_p for normal systems of ordinary differential equations.

Proposition 6.6.8. *Let $\mathbf{\Phi}(x, \mathbf{u}, \mathbf{u}') = 0$ be an implicit ordinary differential equation such that the Jacobian $\partial\mathbf{\Phi}/\partial\mathbf{u}'$ has a bounded inverse on the interval $\mathcal{X} = [0, X]$. Furthermore, let the function $\mathbf{\Phi}$ be Lipschitz continuous in $\mathbf{u}$ with constant L_0 and let $\tilde{\mathbf{u}}(x)$ be a solution of the perturbed equation $\mathbf{\Phi}(x, \tilde{\mathbf{u}}, \tilde{\mathbf{u}}') = \boldsymbol{\delta}(x)$. Then for all $x \in [0, X]$*

$$|\mathbf{u}(x) - \tilde{\mathbf{u}}(x)| \leq C\Big(|\mathbf{u}(0) - \tilde{\mathbf{u}}(0)| + \int_0^x |\boldsymbol{\delta}(\xi)|d\xi\Big) \tag{6.40}$$

where the constant C depends only on $\mathbf{\Phi}$ and the length X of the interval.

Proof. We solve the original and the perturbed equation for the derivatives $\mathbf{u}'$ and $\tilde{\mathbf{u}}'$, respectively, which is always possible under the made assumptions. This yields the equations $\tilde{\mathbf{u}}' = \boldsymbol{\phi}(x, \tilde{\mathbf{u}}, \boldsymbol{\delta})$ and $\mathbf{u}' = \boldsymbol{\phi}(x, \mathbf{u}, 0)$. The function $\boldsymbol{\phi}$ satisfies a Lipschitz condition in $\tilde{\mathbf{u}}$ and $\boldsymbol{\delta}$ with a constant L_1 depending on

L_0 and the bound on $(\partial\mathbf{\Phi}/\partial\mathbf{u}')^{-1}$. Now we integrate both equations from 0 to x, subtract them and take the norm:

$$\begin{aligned}|\mathbf{u}(x)-\tilde{\mathbf{u}}(x)| \le |\mathbf{u}(0)-\tilde{\mathbf{u}}(0)| + \\ \Big|\int_0^x \big[\boldsymbol{\phi}\big(\xi,\mathbf{u}(\xi),0\big)-\boldsymbol{\phi}\big(\xi,\tilde{\mathbf{u}}(\xi),\boldsymbol{\delta}(\xi)\big)\big]\,d\xi\Big|\,.\end{aligned} \tag{6.41}$$

Interchanging the integration and the norm, we exploit the Lipschitz continuity of $\boldsymbol{\phi}$ to obtain

$$|\mathbf{u}(x)-\tilde{\mathbf{u}}(x)| \le |\mathbf{u}(0)-\tilde{\mathbf{u}}(0)| + L_1\int_0^x \big(|\mathbf{u}(\xi)-\tilde{\mathbf{u}}(\xi)|+|\boldsymbol{\delta}(\xi)|\big)\,d\xi\,. \tag{6.42}$$

Obviously, $\int_0^x |\boldsymbol{\delta}(\xi)|d\xi$ is a monotonically increasing function of x on the interval $[0,X]$ taking its maximal value at $x=X$. Applying Gronwall's Lemma with $\psi(x)=|\mathbf{u}(x)-\tilde{\mathbf{u}}(x)|$ and $A=|\mathbf{u}(0)-\tilde{\mathbf{u}}(0)|+L_1\int_0^X|\boldsymbol{\delta}(\xi)|d\xi$, $B=L_1$ yields immediately the claimed estimate with $C=\max(1,L_1)\exp(L_1X)$. □

Thus a fully implicit system of ordinary differential equations has generally the perturbation index $\nu_p=1$, as the right hand side of (6.40) is the L^1 norm of $\boldsymbol{\delta}$. But note that in order to derive (6.42) we had to interchange in (6.41) the integration and the norm. If we start with an *explicit* system, so that the perturbed form is $\tilde{\mathbf{u}}'=\mathbf{f}(x,\tilde{\mathbf{u}})+\boldsymbol{\delta}$, such an interchange is not necessary and we obtain on the right hand side of (6.40) a term of the form $|\int_0^x \boldsymbol{\delta}(\xi)d\xi|$. Interpreting $\|\boldsymbol{\delta}\|_{-1}$ as the (maximum) norm of $\int_0^x\boldsymbol{\delta}(\xi)d\xi$, we see that such a system has in fact the perturbation index $\nu_p=0$.

Theorem 6.6.9. *Let* $\mathbf{\Phi}(x,\mathbf{u},\mathbf{u}')=0$ *be a differential algebraic equation. Then*

$$\tilde{\nu}_d \le \nu_p \le \tilde{\nu}_d+1\,. \tag{6.43}$$

Proof. The essential step of the proof is to show that the completion of the perturbed form of the given differential algebraic equation leads to an underlying equation of the form

$$\mathbf{\Phi}^c(x,\tilde{\mathbf{u}},\tilde{\mathbf{u}}',\boldsymbol{\delta},\boldsymbol{\delta}',\dots,\boldsymbol{\delta}^{(\tilde{\nu}_d)})=0 \tag{6.44}$$

with $\partial\mathbf{\Phi}^c/\partial\boldsymbol{\delta}^{(\tilde{\nu}_d)}\neq 0$. Then we apply Proposition 6.6.8 and get an estimate of the form (6.37) with $\nu_p-1=\tilde{\nu}_d$. The lower bound is obtained by taking into account the remark above: if the integration can be left within the norm, it kills one derivative. ν_p cannot be less than $\tilde{\nu}_d$, as the solution of (6.44) depends at least on $\boldsymbol{\delta}^{(\tilde{\nu}_d-1)}$, and by definition $\tilde{\nu}_d$ is the smallest number of differentiations leading to an underlying equation.

Before the completion we transform the perturbed differential algebraic equation into semi-explicit form; this is a purely algebraic process which does not introduce any derivatives. As outlined in Section 6.1, the completion process takes then the following simple form: the algebraic equations are differentiated and it is checked whether this yields any new equations; the completion stops as soon as no new *algebraic* equations arise.

In the first step the algebraic part of the differential algebraic equation is differentiated. If this leads to any new equations, they contain a first-order derivative of a perturbation. If some of the new equations are algebraic, they are differentiated again. Thus any new equations arising out of them contains a second-order derivative of a perturbation and so on. The perturbed determinacy index $\tilde{\nu}_d$ counts the last completion step where new *differential* equations appear, i. e. equations that are part of the underlying equation $\mathbf{\Phi}^c = 0$. Thus these equations depend effectively on derivatives of order $\tilde{\nu}_d$ of the perturbations and $\mathbf{\Phi}^c$ is of the form claimed above. □

Example 6.6.10. We consider again the system of Example 6.6.5. As already mentioned, here $\nu_p = \tilde{\nu}_d$ in agreement with our theorem. We are at the lower bound, as the underlying equation is of the *explicit* form $u' = \delta' - (\epsilon\epsilon')'$, $v' = \epsilon'$ and we can thus perform the integration within the norm. ◁

Now we want to extend these results to partial differential equations. We cannot expect to obtain a theory with the same level of generality, as estimates on solutions of partial differential equations are much more difficult to find than for ordinary differential equations. So we will make a rigorous statement only for a rather narrow class of equations. Looking at the proof of Theorem 6.6.9, we see that it depends essentially on the concept of an underlying equation and that this is the reason why the determinacy and not the involution index is appearing.

So far we have a notion of an underlying equation, or more precisely of an underlying vector field, only for ordinary differential equations. Thus we need first a more general definition. The basic idea is that an underlying equation is always a *normal* equation certain solutions of which coincide with solutions of our given differential equation.

Definition 6.6.11. *Let $\mathcal{R}_q \subseteq J_q\mathcal{E}$ be an arbitrary differential equation. A normal differential equation $\bar{\mathcal{R}}_{q+r} \subset J_{q+r}\mathcal{E}$ such that any (local) solution $\sigma : \mathcal{X} \to \mathcal{E}$ of $\mathcal{R}_q$ also solves $\bar{\mathcal{R}}_{q+r}$ is an* underlying equation *for $\mathcal{R}_q$.*

Let $\mathcal{R}^{(s)}_{q+r}$ be the involutive completion of $\mathcal{R}_q$. We may take a local representation of it in Cartan normal form. Assuming that $\mathcal{R}^{(s)}_{q+r}$ is not underdetermined, the subsystem of class n defines a normal equation satisfying the conditions of Definition 6.6.11. Thus, as for ordinary differential equations, the completion provides us with a simple method for the construction of a underlying equation. Again, we do not need a full completion; the determinacy index ν_d equals the number of prolongations necessary to find the underlying equation.

Theorem 6.6.12. *Let $\mathcal{R}_q$ be a linear equation and assume that the right hand side of its underlying equation $\bar{\mathcal{R}}_{q+r}$ obtained via a completion can be written in the form $A\mathbf{u}$ where A is the generator of a strongly continuous semigroup acting on an appropriately chosen Banach space. Then $\tilde{\nu}_d \leq \nu_p \leq \tilde{\nu}_d + 1$.*

Proof. As the basic idea behind semigroup theory [134, 149] is to treat normal partial differential equations as ordinary differential equations on some infinite-dimensional Banach space, we try to follow the same strategy as in the proof of Theorem 6.6.9.

Let us assume for simplicity that $q + r = 1$. We slightly change our notation and write t instead of x^n. Derivatives with respect to x^n are denoted by a dot. Adding a perturbation $\boldsymbol{\delta}$, the underlying equation has then the form $\dot{\mathbf{u}} = A\mathbf{u} + \boldsymbol{\delta}$ where $\mathbf{u}$ and $\boldsymbol{\delta}$ are considered as functions of t taking values in the chosen Banach space. It is well-known that the solution $\tilde{\mathbf{u}}$ of this equation is given by

$$\tilde{\mathbf{u}}(t) = S(t)\tilde{\mathbf{u}}(0) + \int_0^t S(t-\tau)\boldsymbol{\delta}(\tau)\,d\tau\,. \tag{6.45}$$

Here $S(t)$ is the semigroup generated by A and $\tilde{\mathbf{u}}(0)$ an arbitrary element of the domain of A.

We are interested in an estimate of the difference between $\tilde{\mathbf{u}}(t)$ and the solution $\mathbf{u}(t)$ obtained for $\boldsymbol{\delta} \equiv 0$. If $\|\cdot\|$ denotes the norm of the used Banach space, it follows immediately from (6.45) that

$$\|\tilde{\mathbf{u}}(t) - \mathbf{u}(t)\| \leq C\Big(\|\tilde{\mathbf{u}}(0) - \mathbf{u}(0)\| + \int_0^t \|\boldsymbol{\delta}(\tau)\|\,d\tau\Big) \tag{6.46}$$

for some positive constant C, as the norms of the operators $S(t)$ of a strongly continuous semigroup are bounded on a finite interval.

If the perturbed determinacy index of $\mathcal{R}_q$ is $\tilde{\nu}_d$, the perturbation $\boldsymbol{\delta}$ of the underlying equation depends on derivatives up to order $\tilde{\nu}_d$ of the perturbation of the original system. Thus we get $\nu_p \leq \tilde{\nu}_d + 1$. By exactly the same reasoning as in the case of ordinary differential equations, it may happen that the semigroup $S(t)$ has a particularly simple structure such that we obtain (6.46) with the integration within the norm. In this case $\nu_p = \tilde{\nu}_d$; obviously no lower value is possible. $\square$

6.7 Applications II: Semi-Discretisations

In this section we study *semi-discretisations* of partial differential equations. For simplicity, we restrict to first-order linear homogeneous systems with constant coefficients. We assume that we are given a partial differential equation in $n+1$ independent variables. Then we discretise by some finite difference method the spatial derivatives with respect to n of these variables. This yields a differential algebraic equation in the remaining independent variable. We are interested in the relation between the involution indices of the original partial differential equation and the obtained differential algebraic equation.

Thus consider the first-order linear homogeneous system

$$\sum_{i=1}^{n+1} M_i \mathbf{u}_{x^i} + N\mathbf{u} = 0 \; . \tag{6.47}$$

The matrices M_i and N are here completely arbitrary (we are *not* assuming that we are dealing with a weakly overdetermined system). Instead of (6.47) we complete to involution the *perturbed* system

$$\sum_{i=1}^{n+1} M_i \mathbf{u}_{x^i} + N\mathbf{u} = \boldsymbol{\gamma}(\mathbf{x}) \; . \tag{6.48}$$

As we are dealing with a linear constant coefficients system, the perturbation does not affect the completion. However, it serves as a convenient mean of "book-keeping" which prolongations are needed during the completion.

Definition 6.7.1. *The* involution index ν_ℓ in direction x_ℓ *of the system* (6.47) *is the maximal number of differentiations with respect to* x^ℓ *appearing in a* $\boldsymbol{\gamma}$*-derivative contained in the involutive completion of the perturbed system* (6.48)*.*

For the semi-discretisation we proceed as follows. Assume that x^ℓ is the independent variable "surviving" the discretisation; in other words, afterwards we are dealing with an ordinary differential equation containing only x^ℓ-derivatives. In order to simplify the notation, we rewrite (6.47). We denote x^ℓ by t and renumber the remaining independent variables as x^i with $1 \leq i \leq n$. Then we solve as many equations as possible for a t-derivative, as we consider these as the derivatives of highest class. This yields a system of the form

$$E\mathbf{u}_t = \sum_{i=1}^{n} A_i \mathbf{u}_{x_i} + B\mathbf{u} + \boldsymbol{\delta} \; , \tag{6.49a}$$

$$0 = \sum_{i=1}^{n} C_i \mathbf{u}_{x_i} + D\mathbf{u} + \boldsymbol{\epsilon} \, . \tag{6.49b}$$

Here we assume, as for a weakly overdetermined system, that the matrix E is of maximal rank, i. e. every equation in the subsystem (6.49a) really depends on a t-derivative. And we assume that the coordinates (x, t) are δ-regular. However, we do not pose any restrictions on the ranks of the matrices C_i.

In (6.49) we have introduced perturbations $\boldsymbol{\delta}$ and $\boldsymbol{\epsilon}$ which are related to the perturbations $\boldsymbol{\gamma}$ in (6.48) by linear transformations with constant coefficients. As we are only interested in the number of differentiations applied to them during the completion, such a transformation has no effect.

We discretise the "spatial" derivatives, i. e. the derivatives with respect to the x^i, on a grid where the points $x_{\mathbf{k}}$ are labelled by integer vectors $\mathbf{k} = [k_1, \dots, k_n]$. $\mathbf{u}_{\mathbf{k}}(t)$ denotes the value of the function $\mathbf{u}$ at the point $x_{\mathbf{k}}$ at time t. We approximate in (6.49) the spatial derivative $\mathbf{u}_{x_i}(x_{\mathbf{k}}, t)$ by the finite difference

$$\delta_i \mathbf{u}_{\mathbf{k}}(t) = \sum_{\ell_i = -a_i}^{b_i} \alpha_{\ell_i}^{(i)} \mathbf{u}_{[k_1, \dots, k_i + \ell_i, \dots, k_n]}(t) \tag{6.50}$$

with some real coefficients $\alpha_{\ell_i}^{(i)}$. Thus different discretisations are allowed for different values of i, but $\mathbf{u}_{x_i}$ is everywhere discretised in the same way. Entering the approximations (6.50) of the spatial derivatives into (6.49) yields the differential algebraic equation

$$E\dot{\mathbf{u}}_{\mathbf{k}} = \sum_{i=1}^{n} A_i \delta_i \mathbf{u}_{\mathbf{k}} + B\mathbf{u}_{\mathbf{k}} \,, \tag{6.51a}$$

$$0 = \sum_{i=1}^{n} C_i \delta_i \mathbf{u}_{\mathbf{k}} + D\mathbf{u}_{\mathbf{k}} \,. \tag{6.51b}$$

Theorem 6.7.2. *Let the coordinates* (x, t) *be* δ*-regular for* (6.47). *The involution index of the differential algebraic equation* (6.51) *obtained in the above described semi-discretisation of* (6.47) *with respect to* x^ℓ *is* ν_ℓ.

Proof. We must compare what happens during the completion of (6.49) and (6.51), respectively. Determining the index ν_ℓ requires keeping track of when some the perturbations are differentiated with respect to t. The index of the differential algebraic equation tells us how many prolongations of the algebraic equations in (6.51b) lead to integrability conditions. Thus we must show that whenever a perturbation in (6.49) is differentiated with respect to t an integrability condition appears in (6.51) and vice versa.

For the partial differential system (6.49) the differentiations with respect to t lead to the second-order equations

$$E\mathbf{u}_{tt} = \sum_{i=1}^{n} A_i \mathbf{u}_{x_i t} + B\mathbf{u}_t + \boldsymbol{\delta}_t \,, \tag{6.52a}$$

$$0 = \sum_{i=1}^{n} C_i \mathbf{u}_{x_i t} + D\mathbf{u}_t + \boldsymbol{\epsilon}_t \,. \tag{6.52b}$$

As t is non-multiplicative for all equations in (6.49b), the equations (6.52b) are decisive for our analysis. The spatial differentiations of the partial differential system (6.49) yield for $1 \leq j \leq n$

$$E\mathbf{u}_{x_j t} = \sum_{i=1}^{n} A_i \mathbf{u}_{x_i x_j} + B\mathbf{u}_{x_j} + \boldsymbol{\delta}_{x_j} \,, \tag{6.53a}$$

$$0 = \sum_{i=1}^{n} C_i \mathbf{u}_{x_i x_j} + D\mathbf{u}_{x_j} + \boldsymbol{\epsilon}_{x_j} \,. \tag{6.53b}$$

All equations in (6.53a) are the result of differentiations with respect to multiplicative variables. In (6.53b) we cannot make a similar statement, as we have not transformed our original system into a full triangular form and thus do not know which variables are multiplicative for which equation. However, this information is not needed later.

Similarly, we differentiate the differential algebraic equation (6.51)

$$E\ddot{\mathbf{u}}_k = \sum_{i=1}^{n} A_i \delta_i \dot{\mathbf{u}}_k + B\dot{\mathbf{u}}_k \,, \tag{6.54a}$$

$$0 = \sum_{i=1}^{n} C_i \delta_i \dot{\mathbf{u}}_k + D\dot{\mathbf{u}}_k \,. \tag{6.54b}$$

In addition, we write down – in formal analogy to the equations (6.53) obtained by spatial differentiations – the following equations obtained by applying the difference operators δ_j for $1 \leq j \leq n$ to (6.51)

$$E\delta_j \dot{\mathbf{u}}_k = \sum_{i=1}^{n} A_i \delta_i \delta_j \mathbf{u}_k + B\delta_j \mathbf{u}_k \,, \tag{6.55a}$$

$$0 = \sum_{i=1}^{n} C_i \delta_i \delta_j \mathbf{u}_k + D\delta_j \mathbf{u}_k \,. \tag{6.55b}$$

They are obviously just linear combinations of equations in (6.54); however, their addition simplifies the subsequent analysis.

Given all these equations, we analyse where either integrability conditions or obstructions to involution might occur. We distinguish two cases. Case I analyses (6.52b) and (6.54b), respectively. New equations arise whenever it is not possible to express all equations in (6.52b) (in (6.54b), respectively) as linear combinations of the equations in (6.53a) (in (6.55a), respectively). Obviously, the number of new equations is the same in both systems in view of their identical structure and those for the differential algebraic equation are semi-discretisations of those for the original partial differential system. For the partial differential system each of them contains a term ϵ_t. In the case of the differential algebraic system all new equations are integrability conditions; for the partial differential system some of them might represent obstructions to involution, namely those of second order.

Case II occurs only for the partial differential system: integrability conditions or obstructions to involution could arise *within* the subsystem (6.53b). No simple criterion exists for this, as we have not transformed to a full triangular form. But this is not important for us, as all these new equations contain only spatial derivatives of the perturbations and thus do not affect our theorem. There are no corresponding equations for the differential algebraic equation, as (6.55) is algebraically dependent of the original system.

If Case I does not lead to any new equations, no hidden constraints exist in the differential algebraic equation. Thus it is involutive and of index 0. The partial differential system is not necessarily involutive, as Case II might still lead to new equations. Our theorem would be wrong, if differentiating these with respect to the non-multiplicative variable t yielded again new equations. However, this cannot happen: by interchanging the order of differentiations one sees that any equation arising this way may also be generated by combining spatial derivatives of equations stemming from Case I.[6] This implies that no additional differentiations with respect to t of the perturbations are performed – no matter how many further iterations within Case II are necessary. Hence the index ν_ℓ vanishes in accordance with our claim.

If Case I yields new equations, both the involution index of the differential algebraic equation and the index ν_ℓ are at least 1. We may then continue in precisely the same manner. For the partial differential system we take all equations in (6.52) and (6.53) and first differentiate them with respect to t. Then we differentiate them with respect to the spatial variables x^k. Similarly, we take the equations in (6.54) and (6.55) and differentiate them with respect to t. Finally, we consider what we obtain by applying the difference operators δ_k to them.

The analysis of these third-order systems proceeds as above. We distinguish the same two cases: Case I involves time derivatives and thus adds *one* differentiation with respect to t to the perturbations; Case II concerns only the partial differential system and does not lead to any t-differentiations.

[6] This is exactly the same argument as in the proof of Proposition 3.1.10 where all calculations involved are performed in full detail.

New equations arising in Case II do not lead at later stages of the completion to any additional t-differentiations of the perturbations, as we may always interchange the order of differentiations and thus all the differentiations with respect to t have already been studied at an earlier stage in Case I.

In this manner we proceed to higher-order systems until no new equations appear in Case I. Because of our assumption that the coordinates are δ-regular, this happens after a finite number of steps. It is a clear that at stage k new equations arise only out of the new equations of stage $k-1$, so that those of Case I in stage k contain derivatives of the perturbations with precisely k differentiations with respect to t. Thus the involution index of the differential algebraic equation (6.51) is indeed equal to the maximal number of differentiations with respect to t applied to the perturbations ϵ. But this is just the definition of the involution index ν_ℓ in direction x^ℓ. □

This result is somewhat surprising. While one surely expects that integrability conditions of the original partial differential equation induce integrability conditions in the differential algebraic equation obtained by semi-discretisation, Theorem 6.7.2 says that obstructions to involution also turn into integrability conditions upon semi-discretisation. Thus even if the original partial differential equation is formally integrable, the differential algebraic equation might contain integrability conditions.

Example 6.7.3. Consider the following linear system for two unknown functions $v(x,t)$, $w(x,t)$:

$$\begin{pmatrix} v \\ w \end{pmatrix}_t = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} v \\ w \end{pmatrix}_x , \qquad 0 = \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} v \\ w \end{pmatrix}_x \tag{6.56}$$

This is the same system as in Example 5.4.3 written in matrix notation. We know that it is formally integrable but not involutive, as differentiating the constraint equation with respect to the non-multiplicative variable t yields (after a simplification) a new second-order equation as obstruction to involution: $w_{xx} = 0$.

A semi-discretisation with backward differences for the spatial derivatives leads to the differential algebraic equation

$$\Delta x \begin{pmatrix} \dot{v}_n \\ \dot{w}_n \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} v_n - v_{n-1} \\ w_n - w_{n-1} \end{pmatrix} , \quad 0 = \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} v_n - v_{n-1} \\ w_n - w_{n-1} \end{pmatrix} . \tag{6.57}$$

As predicted by Theorem 6.7.2, it is not involutive, as it has the integrability condition $w_n - 2w_{n-1} + w_{n-2} = 0$ obviously representing an approximation of the obstruction to involution $w_{xx} = 0$ by centred differences. ◁

Corollary 6.7.4. *Let ν_i be the involution index of* (6.47) *and assume that all one-forms* $dx^\ell = 0$ *are non-characteristic. Then any differential algebraic equation arising by a semi-discretisation of* (6.47) *has an involution index less than or equal to* ν_i.

Proof. We trevally have $\nu_\ell \leq \nu_i$ for all ℓ. □

For the weakly overdetermined systems introduced in Section 5.6, Theorem 6.7.2 can be strengthened to the statement that the differential algebraic equation obtained by a semi-discretisation with respect to t is formally integrable, *if and only if* the original system of partial differential equations is involutive [165]. And this result does not only hold for semi-discretisations by finite difference methods but also for a spectral approach. Using periodic boundary condition we may expand the solution into a spatial Fourier series with time-dependent coefficients. These coefficients satisfy then a differential algebraic equation which is formally integrable, *if and only if* the original system was involutive.

6.8 Notes

We described the completion process for ordinary differential equations first in terms of the tangent bundle, as in this form it has been rediscovered many times; see e. g. [121, 126, 144, 146, 175]. In some of these works the authors also treat non-autonomous systems within the tangent bundle. This requires some slightly awkward constructions. Usually, the time variable is included in the configuration space, so that one deals with the manifold $\mathcal{U} \times \mathbb{R}$. Then one defines the "restricted" tangent bundle as the set of all those vectors in $T(\mathcal{U} \times \mathbb{R})$ which have a 1 in the last component. The jet bundle offers a much more natural framework for such systems and the restricted tangent bundle is easily recovered with the contact map. An intrinsic proof of the equivalence of the tangent and the jet bundle approach to completion was given in [50].

The notions of continuous and constructive divisions were introduced by Gerdt and Blinkov [56]. Obviously, these are rather technical concepts and especially constructivity is often difficult to prove. They are designed exactly in such a way that our proof of Proposition 6.4.1 works. Combined they ensure that no infinite chains can appear in Algorithm 6.2. Despite its complicated definition, it is obvious that constructivity is a necessary property for the correctness of our algorithms. Otherwise we would be forced to consider also multiplicative indices. For continuity the situation is less clear. While it is sufficient for Proposition 6.3.4 to hold, it is not known whether it is also necessary.

The basic ideas of Algorithm 6.4 are already contained in the Janet–Riquier Theory of differential equations (see the Notes of Chapter 2). Our formulation is essentially due to Blinkov, Gerdt and Zharkov [56, 57, 195] (compare also the notes at the end of Chapter 2). In recent years, they have spent, together with further collaborators, much effort on making the algorithm more efficient and on providing fast implementations. [57] contains a more sophisticated version of Algorithm 6.4 which has in particular for Janet bases some advantages.

As already mentioned in the Notes of Chapter 2, Apel [8] introduced his own concept of an involutive division. His divisions are defined with respect to only one subset $\mathcal{N} \subseteq \mathbb{N}_0^n$. As during the completion the set $\mathcal{H}$ is changed in every step, we always need a new division for the new set $\mathrm{le}_{\prec}\mathcal{H}$ and Apel discusses some heuristic strategies for choosing the divisions.[7] Whether this additional freedom can be exploited to achieve a higher efficiency is an open question. A definite advantage of this approach is that it is almost trivial to prove the existence of an involutive basis for arbitrary ideals.

However, one should note the following difference. In Apel's approach it is not even possible to formulate the task to determine an involutive basis for a prescribed involutive division, as we do not know the relevant set $\mathrm{le}_{\prec}\mathcal{H}$, before the completion is over. Thus this approach is only useful, if the goal is to somehow get some Stanley decomposition (or if the involutive theory is only used as an alternative for Buchberger's algorithm to compute Gröbner bases); the role of the involutive division is mainly that of a technical trick to improve efficiency. Up to now, this is not really a disadvantage, as not much is known about special properties of Stanley decompositions for different divisions (with the exception of Pommaret bases!). But one might expect that in the future the involutive division will gain a similar importance as the term order for a Gröbner basis and that for different tasks one is interested in different involutive divisions.

Our algorithm can in principle be applied to nonlinear equations, too, provided the nonlinearities are only polynomial. However, the reduction process is then more subtle; one must resort to *pseudo-reductions* (see e. g. [123]). This makes it much harder to interpret the outcome. In particular, it is possible that one leaves the differential ideal spanned by the original system, as equations must be multiplied with differential polynomials. In the language of differential algebra one obtains a *coherent autoreduced set* and our algorithm may be considered as an alternative to the classical *Kolchin–Ritt algorithm* [93, 151].

A special situation arises for quasi-linear equations. Since they are linear in the derivatives of highest order, all reductions can be performed effectively as linear operations. However, we cannot expect the equation to remain quasi-linear during the completion, as the integrability conditions may be fully non-linear. Provided all appearing equations are quasi-linear, Algorithm 6.4 can be used without problems and produces the correct result. In this case, also no problems with the regularity of the equation appear. A simple example are the Navier–Stokes equations.

The hybrid algorithm of Section 6.5 combining geometric and algebraic ideas was developed in the master's thesis of Hausdorf [70] (see also [71]). Within this thesis, the algorithm was also implemented in the computer algebra system *MuPAD*.

[7] One should note that Apel admits the case $\mathcal{N} = \mathbb{N}_0^n$, i. e. divisions defined on the set of all multi indices. For such divisions, there is no need to change the divisions within a computation and in fact one can formulate the classical involutive divisions like the Pommaret and the Janet division in his framework. In this case his theory largely coincides with ours.

Compared with earlier works [157, 161] using a naive realisation of the Cartan–Kuranishi completion it is much more efficient. The results of Section 2.4 on detecting δ-singular coordinate systems can also be incorporated into the algorithm; for details we refer to [71, Section 5]. For the application to non-linear equations, the same comments as above for the purely algebraic case apply.

The brute force realisation of the Cartan–Kuranishi completion can, in principle, handle equations with polynomial non-linearities fully algorithmically, as all necessary operations can be performed effectively using (algebraic) Gröbner bases. However, this approach requires the determination of a Gröbner basis for the local representation of each appearing differential equation. As the number of variables and polynomials in these bases are rapidly growing with each prolongation, this seems to be feasible only for fairly small examples becoming involutive very soon.

The completion approaches discussed in this chapter are not the only ones; a number of alternatives exists. Gröbner bases are by now a standard tool in commutative algebra and algebraic geometry. Several algorithms for their construction has been designed and their improvement is a very active field of research. More on these bases can be found in Appendix B.2.

As any involutive basis is also a Gröbner basis, it appears natural to compare Algorithm 6.4 with Buchberger's Algorithm B.2. A direct implementation of the latter one is very inefficient; only the use of optimisations like the two Buchberger criteria (Propositions B.2.15 and B.2.16) for detecting unnecessary reductions makes it usable. In contrast, a direct implementation of Algorithm 6.4 yields already reasonable results, although improvements are possible. Some are discussed in [58, 59] and references therein. The second article also contains tables with the results of extensive benchmarks showing that the involutive approach is highly competitive as long as the involutive basis is not considerably larger than the Gröbner basis.

Some simple heuristic explanations of these observations are possible. Algorithm 6.4 implicitly also computes S-polynomials but less than a naive implementation of the Buchberger algorithm. Both Buchberger criteria are not completely but to a large extent automatically contained in the strategy of multiplying only by non-multiplicative variables. For the first criterion this is obvious; for the second one it follows from Proposition 2.6.2. Some further arguments are given in [8].

This strategy also has certain similarities with the so-called *Hilbert driven Buchberger algorithm* introduced by Traverso [179]. In this variant of the Buchberger algorithm one assumes that one knows already the Hilbert function of the ideal and this knowledge is used to avoid further unnecessary reductions. In concrete computations, this idea turns out to be quite effective. Unfortunately, the determination of the Hilbert function requires usually a Gröbner basis. Thus the main application of this technique consists of transforming a Gröbner basis with respect to one term order into one with respect to another term order.

In this context, our algorithm can be understood as follows. Given a basis of the ideal, we make a conjecture about its Hilbert function based on the assignment of multiplicative variables. The multiplication of the elements by their non-multiplicative variables checks whether the conjectured function is already the correct Hilbert function. The algorithm stops, as soon as this is the case. Note that in contrast to the Hilbert driven Buchberger algorithm no a priori knowledge of the Hilbert function is required.

Proposition 6.4.9 says that even if Algorithm 6.4 does not terminate, it yields in a finite number of steps a Gröbner basis for term orders of type ω. In this form, this result is only of theoretical interest, as it is unclear how one should detect the Gröbner basis. Using standard criteria would destroy all potential advantages of our algorithm. For the special case of Pommaret bases, Apel [7] found a simple criterion, so that a variant of Algorithm 6.4 can be used for the construction of Gröbner bases independent of the existence of a finite involutive basis.

In the case of differential equations one would like for many applications to have a simple algorithm to determine all integrability conditions. We have seen in Example 6.2.2 that they significantly simplify the explicit solution of the differential equation. Unfortunately, so far nobody has been able to find a finite criterion for formal integrability. Thus all approaches construct more than just the integrability conditions and they differ in what this "more" is. Most of them have been studied in the context of Lie symmetry theory [20, 129] for the treatment of the determining systems, which are typically large overdetermined linear systems of partial differential equations. Hence a number of references to theoretical works and implementations can be found in the review articles of Hereman [74, 75].

We mention here only three approaches popular in computer algebra. The probably first implementation of the *Janet–Riquier Theory* in a computer algebra system is due to Topunov [178]. Reid [147] developed a *standard form*, also inspired by the Janet–Riquier Theory, and implemented it in MAPLE. The approach is fully algorithmic only for linear equations, but this suffices for applications in symmetry theory. In the more recent work [148] he speaks of a "reduced involutive form" which is not identical with our notion of involution. Mansfield [112] introduced *differential Gröbner bases* by combining ideas from differential algebra and classical Gröbner basis theory; she also provides a MAPLE implementation. One should, however, be careful with the name. There does not yet exist a generally accepted definition of how Gröbner bases should be generalised to differential equations. Mansfield's bases cannot solve the differential ideal membership problem and thus one may argue whether they should be called Gröbner bases.

This last approach is rooted in *differential algebra*, a branch of algebra studying rings and fields with an additional structure, namely derivations, that models in an algebraic manner the basic properties of differentiation. Some classical references are [89, 93, 151]. As an example how this theory may be used in the context of completion, see also [23].

A high worst case complexity is a common problem for all these approaches. As already mentioned, this is not very surprising, as most of them comprise Gröbner basis computations as a special case. Thus the complexity results obtained for Gröbner bases (see Appendix B.2) represent a lower bound for the completion in the differential case. Besides this fairly simple observation not much is known about the complexity of all the mentioned completion algorithms.

The relation between the formal theory and the index concepts in the numerical theory of differential algebraic equations was first noted by Le Vey [104, 105] and Piirilä and Tuomela [136]. Tuomela and Arponen [183, 184] discuss some concrete numerical methods based on these ideas. The indices were generalised to partial differential equations in [163]. These works concentrated on differentiation indices; the estimation of perturbation indices by perturbed differentiation indices was first done in [72] based on earlier works by Campbell and Gear [30].

The definition of indices for partial differential equations is still an active field of research. Most definitions in the literature (see e. g. [31, 108, 109]) are based on first reducing to only one independent variable by semi-discretisations, integral transforms or some other methods. Then the index of the arising differential algebraic equation is used. Obviously, the result depends in general on the used reduction technique. In contrast, our index definitions can be directly applied to the given partial differential equation.

It is of great interest to relate these two approaches, as such reduction methods often form the basis of concrete numerical methods. If the index computed directly from the partial differential equation differs from the index of the differential algebraic equation obtained by the chosen numerical method, this could indicate that the method destroys some structural properties of the original differential equation. But of course this requires that the "right" indices are used in both cases. Currently not much is known about such comparisons. The results we presented in Section 6.7 on the semi-discretisation of linear systems with constant coefficients are the first ones in this direction and stem from [165].

An obvious question is whether similar results hold for linear systems with *variable* coefficients. This is not the case, as differential and difference operators obey different Leibniz rules. The proof of Theorem 6.7.2 depends decisively on the possibility to mimic the operations on the partial differential equation on the differential algebraic equation. This is not possible for systems with variable coefficients, as the evaluation points of the coefficients have to be taken into account.

Hence for arbitrary discretisations the indices of the partial differential equation and of the differential algebraic equation might differ considerably. However, it is conceivable that for *special* discretisations something similar to Theorem 6.7.2 holds. In fact, this should be considered as a very good and desirable property. In recent times, some researchers have studied so-called *mimetic* discretisations (see for example [82]). While their motivation is not an involution analysis, their approach should lead to similar results.