Differential Elimination-Completion Algorithms for DAE and PDAE

Gregory J. Reid[1]

Department of Mathematics and Statistics
Simon Fraser University
Burnaby, B.C. Canada. V5A 1S6
E-mail: reid@cecm.sfu.ca

Ping Lin[2]

Department of Computer Science
Stanford University, Gates 2B, MC 9025
Stanford, CA 94305-9025
E-mail: plin@sccm.stanford.edu

and

Allan D. Wittkopf[3]

Department of Mathematics and Statistics
Simon Fraser University
Burnaby, B.C. Canada. V5A 1S6
E-mail: wittkopf@cecm.sfu.ca

[1]GR gratefully acknowledges support from the Natural Sciences and Engineering Research Council of Canada (NSERC). Especially acknowledged are U. Ascher's suggestions to explore linearized perturbation analysis of DAE. Also acknowledged are helpful discussions with E. Mansfield and E. Hubert about singular solutions.

[2]PL acknowledges support from U. Ascher's NSERC grant.

[3]ADW acknowledges support from G. Reid's NSERC grant.

Abstract

DAE and PDAE are systems of ordinary and partial differential-algebraic equations with constraints. They occur frequently in applications such as constrained multibody mechanics, space-craft control and incompressible fluid dynamics.

A DAE has differential index r if a minimum of $r+1$ differentiations of it are required before no new constraints are obtained. While DAE of low differential index (0 or 1) are generally easier to solve numerically, higher index DAE present severe difficulties.

Reich, Rabier and Rheinboldt have presented a geometric theory and an algorithm for reducing DAE of high differential index to DAE of low differential index. Rabier and Rheinboldt also provided an existence and uniqueness theorem for DAE of low differential index. We show that for analytic autonomous first order DAE this algorithm is equivalent to the Cartan-Kuranishi algorithm for completing a system of differential equations to involutive form. The Cartan-Kuranishi algorithm has the advantage that it also applies to PDAE and delivers an existence and uniqueness theorem for systems in involutive form. We present an effective algorithm for computing the differential index of polynomially nonlinear DAE. A framework for the algorithmic analysis of perturbed systems of PDAE is introduced and related to the perturbation index of DAE. Examples including singular solutions, the Pendulum and the Navier-Stokes equations are given. Discussion of computer algebra implementations is also given.

1 Introduction

1.1 Differential Elimination-Completion Algorithms

The Cartan-Kuranishi (CK) algorithm and the algorithm described geometrically by Reich, Rabier and Rheinboldt which form the focus of this article, are examples of differential elimination-completion (DEC) algorithms. Each step of such algorithms consists of either eliminating redundant equations or (partially) completing the system by appending certain derivatives of equations or integrability conditions to obtain a system with the same solutions.

Cauchy and Kovaleskaya's existence and uniqueness theorem for PDE of Cauchy Kovaleskaya type, was generalized to wider classes of PDE in works near the beginning of this century by Tresse [56] and Riquier [45], and subsequently in the geometrical works of Cartan [8] and others [17, 3]. These authors not only provided existence and uniqueness theorems, but also DEC algorithms for manipulating systems into forms to which these theorems could be applied.

Such methods involve the application of a finite number of differentiations and algebraic manipulations to the original system of PDE and produce a system with the same solutions. Since they do not require knowledge of the solutions of the system they are amenable to computer implementation. The development of flexible, efficient computer algebra systems such as Maple, Mathematica, Macsyma and Reduce has prompted vigorous activity in the development and computer implementation of DEC algorithms. The review article [20] and the introduction of [42] discuss and give numerous references to implementations of such methods. Well-developed theory and efficient computer implementations of DEC algorithms are now available for linear systems. Indeed systems containing thousands of linear PDE have now been successfully analyzed with such DEC algorithms [40, §9.1]. To date such algorithms have been mainly used in geometrical applications such as determining symmetries of differential equations.

Only recently has there been progress on DEC algorithms for nonlinear systems and in particular for systems which are polynomial functions of their dependent variables and derivatives [4, 7, 30, 42, 50]. This has built upon the breakthrough Buchberger's Gröbner basis algorithm brought for polynomial algebraic equations. The Gröbner basis algorithm has been implemented in most computer algebra systems. It transforms systems of polynomial equations into one with the same solutions, called a Gröbner basis. Existence and uniqueness theorems for solutions and many other properties of the original system can be determined from the Gröbner basis of a system. We discuss how the use of Gröbner Bases and methods introduced in [41], can be used to make the DEC algorithms discussed in this article effective for polynomially nonlinear differential equations. Thomas [54, 55] gives an algebraic treatment of Rabier and Rheinboldt's algorithm and the index of quasi-linear DAE. In §5 these methods, which are based on primary decomposition, are extended to PDAE.

1.2 Differential-Algebraic Equations

There has been much recent interest in developing numerical methods to solve Differential-Algebraic Equations (DAE). These are systems of ODE that are in implicit form $\mathbf{g}(t, \mathbf{u}, d\mathbf{u}/dt) = 0$. If the Jacobian of $\mathbf{g}$ is not of full rank with respect to $d\mathbf{u}/dt$ then it may not be possible to solve for $d\mathbf{u}/dt$ to express such a system as an explicit ODE $d\mathbf{u}/dt = \mathbf{f}(t, \mathbf{u})$. Many effective numerical solvers and existence and uniqueness results are known for explicit ODE, but the situation is not as complete for DAE despite much recent progress.

Gear [13, 14] introduced the *differential index* of a DAE and its definition was refined and made coordinate independent in [38, 37]. A DAE has differential index r if a minimum of $r+1$ differentiations of it are required before no new constraints are obtained. Numerical simulations and theoretical investigations have shown the numerical difficulty in solving such systems to be strongly dependent on this index. DAE of low differential index (0 or 1) are numerically the most desirable cases while higher order index DAE can present severe numerical difficulties.

Gear [13, 14] showed that the differential index of a DAE could be reduced by differentiation and this has led to a number DEC algorithms for reducing the differential index of DAE [13, 14, 32, 35, 37]. Many others have contributed substantially to this algorithm with Reich [38] making a very significant contribution to its geometric development. For accuracy of reference we will refer to the geometric form of this algorithm described in the paper of Rabier and Rheinboldt [37] as the Rabier-Rheinboldt algorithm. By a sequence of r differentiation and elimination steps this algorithm converts a DAE of differential index r to one of index zero with the same solutions. Rheinboldt and Rabier [37] also state and prove an existence and uniqueness theorem for the output index zero DAE.

We informally apply the Rabier-Rheinboldt algorithm to the equations for a Pendulum: the classic example of a DAE of differential index three. Geometric details for this example are given in [37].

Example 1 Pendulum *The Lagrange equations of motion for a unit length planar pendulum moving under constant gravity are*

$$d^2x/dt^2 + \lambda x = 0, \quad d^2y/dt^2 + \lambda y + g = 0, \quad x^2 + y^2 - 1 = 0. \tag{1}$$

Rewriting the above as a first order[1] system gives:

$$\begin{aligned} x_t &= u, \;\; y_t = v, \;\; u_t = -\lambda x, \;\; v_t = -\lambda y - g, \\ 0 &= x^2 + y^2 - 1. \end{aligned} \tag{2}$$

[1] By the order of an expression we mean the maximum order of any derivative appearing in the expression. For example the orders of $dx/dt + d^2y/dt^2$, $\partial^2 u(r,s)/\partial r \partial s$ and $u(r,s)$ are respectively 2, 2 and 0.

In this case we have a DAE *for* (x, y, u, v, λ). *Notice that it is not equivalent to an explicit* ODE *since the system above does not include* λ_t. *We call* $x^2 + y^2 - 1 = 0$ *a zeroth order constraint for the system.*

Differentiating the constraint equation and cancelling a factor of 2 yields

$$xx_t + yy_t = 0. \tag{3}$$

Elimination by substituting for x_t, y_t *from (2) yields the new constraint*

$$xu + yv = 0. \tag{4}$$

Differentiation of this constraint yields

$$xu_t + yv_t + x_t u + y_t v = 0. \tag{5}$$

Elimination by substituting for u_t, v_t, x_t, y_t *from (2) yields the new constraint*

$$-(x^2 + y^2)\lambda - gy + u^2 + v^2 = 0. \tag{6}$$

The simplification $x^2 + y^2 = 1$, *although helpful, is not essential to the Rabier-Rheinboldt algorithm. It has not been used to ease the comparison of this algorithm that will later be made with the CK (Cartan-Kuranishi) algorithm. Differentiating this last constraint yields*

$$-(x^2 + y^2)\lambda_t - 2(xx_t + yy_t)\lambda - gy_t + 2uu_t + 2vv_t = 0. \tag{7}$$

We proceed to the fourth elimination-differentiation step. The derivatives u_t, v_t, x_t, y_t *can be eliminated from (7) by using (2). However no previous expression is available for* λ_t. *Hence it is impossible to eliminate* λ_t *to obtain a new zeroth order constraint. Differentiation of the existing (unaltered) constraints yields nothing new. Thus the process of differentiation and elimination terminates since no new zeroth order constraints were found after 4 elimination-differentiation steps. Its output is the system of equations from the first 3 elimination-differentiation steps: (2-7). The input system (2) is said to have differential index* 3 *since a sequence of* $4 = 3 + 1$ *elimination-differentiation steps was required before no new constraints were obtained. The output system has differential index* 0.

Application of the existence and uniqueness theorem of Rabier and Rheinboldt [37] to the output DAE *shows that the initial value problem*

$$x(t^0) = x^0,\ y(t^0) = y^0,\ u(t^0) = u^0,\ v(t^0) = v^0,\ \lambda(t^0) = \lambda^0 \tag{8}$$

with $(x^0, y^0, u^0, v^0, \lambda^0)$ *satisfying the output system (2-7), has a unique local solution.*

Of course the Pendulum system is most easily numerically solved by introducing polar coordinates $x = \cos\theta$, $y = -\sin\theta$ *to eliminate the constraint and convert it to an explicit* ODE $\theta_{tt} + g\sin\theta = 0$. *The Pendulum system has been used as an illustration of behaviour in more complicated* DAE *where explicit construction of such a coordinate transformation is not always practical.*

1.3 Jet Space Methods for DAE and PDAE

Jet space methods associate a given system of ordinary differential or partial differential equations $R = 0$ with a locus of points in a Jet space. Such methods concern themselves with the geometrical study of this locus, which we call the jet variety $\mathcal{R}$, and its relationship with the solutions of the differential equation. They have a long history [8, 10] and have undergone many modern developments [27, 53, 17, 34]. We will obtain known results for DAE using Jet space methods. Such methods also provide a unifying framework in which to view both DAE and PDAE. Little is known about PDAE.

In §2 we illustrate by example how to form the jet variety $\mathcal{R}$ of a given differential equation $R = 0$. This is accomplished by replacing derivatives of solutions in the differential equation by formal variables to form a system of algebraic equations. This process is familiar to many working with DAE. The jet variety is the solution set of this algebraic system and is *not* the solution set of the differential equation. By differentiation the graphs of solutions can be embedded as curves lying in the jet variety (see Figure 1). The union of these embedded graphs forms a variety which we call the *solution variety* $\mathcal{R}_{\text{sol}}$. In general $\mathcal{R}_{\text{sol}} \subset \mathcal{R}$. When $\mathcal{R}_{\text{sol}} = \mathcal{R}$ the differential equation is said to be *locally solvable*.

Perhaps the simplest DEC algorithm of Jet space methods from a geometric viewpoint is the CK Algorithm [34, 50]. By the famous Cartan-Kuranishi Theorem [27], after application of a finite number of prolongations (differentiations) and projections (eliminations) this algorithm renders any system of PDAE or DAE locally solvable or inconsistent. In the former case, the original system of differential equations is replaced with an equivalent system with jet variety $\tilde{\mathcal{R}}$ and solution variety $\tilde{\mathcal{R}}_{\text{sol}}$ such that $\tilde{\mathcal{R}}_{\text{sol}} = \tilde{\mathcal{R}}$. The locally solvable form produced by this algorithm is called an involutive system. Involutive systems are locally solvable and contain (involve) all their integrability conditions. They also allow an existence and uniqueness theorem for local analytic solutions of the original system (the Cartan-Kähler Theorem [23]).

Geometrical methods focus on the form of the variety of $\mathcal{R}$ rather than a specific form of the equations. Algebraic manipulations are allowed on a system of equations that do not alter its variety. For instance, in Example 1, omitting equations (3) and (5) of the output system does not alter its jet variety. Also use of the equations $x^2 + y^2 = 1$ and (3) substantially simplifies some of the equations without changing the output system's jet variety.

It is a familiar process to reduce the differential index of a DAE by differentiation and elimination. This method goes back to the work of Gear [14, 13]. We will prove that the CK algorithm and Rabier-Rheinboldt algorithm [37] are equivalent for first order analytic autonomous DAE. In view of Lewy's famous example [28] of a smooth linear partial differential equation without a solution it has been difficult to drop the requirement of analyticity in the PDE case. In the DAE case Rabier and Rheinboldt have shown that this assumption can be weakened [37].

Blanket locality and analyticity assumption: *We always assume that all differential equations and mappings are local analytic functions of their arguments. For simplicity of exposition we will sometimes discuss local properties in a global manner.*

We will apply the CK algorithm to the following PDAE.

Example 2 Incompressible Navier-Stokes Equations

$$\mathbf{H} := \mathbf{u}_t + (\mathbf{u}\cdot\nabla)\mathbf{u} - \mu(\nabla\cdot\nabla)\mathbf{u} + \nabla p - \mathbf{f} = 0, \tag{9}$$

$$\Phi := \nabla\cdot\mathbf{u} = 0. \tag{10}$$

Normally these are supplemented with initial-boundary conditions

$$\mathbf{u}|_{\partial\Omega} = \mathbf{b}, \ \ \mathbf{u}|_{t=0} = \mathbf{a} \tag{11}$$

in a bounded two or three-dimensional region Ω. *Here* $\mathbf{u}(\mathbf{x}, t)$ *represents the velocity of a viscous incompressible fluid,* $p(\mathbf{x}, t)$ *the pressure,* $\mathbf{a}(\mathbf{x})$ *the prescribed initial velocity,* $\mathbf{f}(\mathbf{x}, t)$ *the prescribed external force and* $\mathbf{b}(\mathbf{x}, t)$ *the prescribed velocity boundary values.*

The system above can be viewed as three partial differential equations $\mathbf{H} = 0$ *with constraint equation* $\Phi = 0$. *This appears to be one of the first* PDAE *for which proven* DAE *methods such as stabilization and regularization methods have been successfully applied and analyzed [29].*

The work of this article is related to that in [50, 52, 51]. Seiler has been fruitfully applying the CK algorithm to unify and relate methods in field theories such as constrained mechanics. In [52] it is shown that the Dirac algorithm [9] for identifying constraints for the equations of Hamiltonian mechanics produces the same output as the CK algorithm. In [51] the CK algorithm is used to give a geometric interpretation of methods arising in the symplectic formalism which treats Lagrangians which are linear in their velocities.

2 Jet space methods for systems of differential equations

We will introduce some of the major concepts of Jet space methods informally through examples. For detailed treatments see [48, 34, 3]. For an introductory treatment which is related to the flavour of our presentation see Olver [33].

Consider an ODE with dependent variable $u(x)$ and independent variable x. Then J^r, the Jet space of order r of the ODE, is the space with coordinates (or jet variables) $x, p_0, p_1, ..., p_r$ which for simplicity we take as $\mathbb{R}^{r+2}$. The natural notation of u, u_x, u_{xx} will be used instead of p_0, p_1, p_2 etc. We emphasise that u, u_x etc are not derivatives of solutions of the ODE, but instead formal coordinates.

2.1 Prolongation of solutions of an ODE

Consider an ODE $R(x, u(x), D_x u(x)) = 0$, which has analytic general solution $u(x; c)$ for $x \in \mathbb{R}$ where c is an arbitrary constant. Then the graph of a solution corresponding to c in the x-u plane $\mathbb{R}^2 = J^0$ is

$$\Gamma_c := \{(x, u(x; c)) \in J^0 : R(x, u(x; c), D_x u(x; c)) = 0\}. \tag{12}$$

In this case $J^1 = \mathbb{R}^3 = \{(x, u, u_x) : \ x, u, u_x \in \mathbb{R}\}$. The prolongation to order 1 of Γ_c is the curve defined by

$$\mathcal{D}\Gamma_c := \{(x, u(x; c), D_x u(x; c)) \in J^1 : R(x, u(x; c), D_x u(x; c)) = 0\}. \tag{13}$$

Example 3 *The* ODE

$$R(x, u(x), D_x u(x)) = D_x u(x) - (3x^2 - 2x + 1)/\cosh(u(x)), \tag{14}$$

has general solution $u(x; c) = \sinh^{-1}(x^3 - x^2 + x + c)$ for real x. The solution with $c = 0$ lies in the x-u plane and is displayed as the bold planar curve in Figure 1. Its prolonged graph is the non-planar bold curve displayed in Figure 1. A family of graphs of solutions and their prolongations are also displayed in Figure 1.

The solution variety of the ODE in J^1 is the union of prolonged graphs of solutions $\bigcup_c \mathcal{D}\Gamma_c$. The solution variety for Example 3 is the shaded surface in Figure 1.

In a similar way we can define the prolongation to order r of a family of solutions of a differential equation as the union of the prolongations of the graphs of the individual solutions:

$$\bigcup_c \mathcal{D}^r\Gamma_c := \{(x, u(x; c), D_x u(x; c), ..., D_x^r u(x; c)) \in J^r : R(x, u(x; c), D_x u(x; c)) = 0\}. \tag{15}$$

2.2 Jet variety of an ODE

The jet variety $\mathcal{R}$ of the ODE $R = 0$ is obtained by replacing the solution $u(x)$ and its derivative $D_x u(x)$ in $R(x, u(x), D_x u(x)) = 0$ by their corresponding Jet coordinates u and u_x:

$$\mathcal{R} := \{(x, u, u_x) \in J^1 : R(x, u, u_x) = 0\}. \tag{16}$$

Thus $\mathcal{R}$ is a surface in J^1. Note that (x, u, u_x) is not a solution of the ODE, but rather a point in $\mathbb{R}^3$ satisfying the algebraic equation $R(x, u, u_x) = 0$. As a consequence of (15), $\mathcal{R}_{\mathrm{sol}} \subseteq \mathcal{R}$. An ODE is said to be *locally solvable* at a point of its jet variety if at least one prolonged graph of a smooth solution passes through that point. We see that the ODE of Example 3 is locally solvable since the union of all its solution graphs is equal to its jet variety. This is not always the case.

2.3 Jet variety of a system of PDE

The above discussion extends easily to systems of PDE. For detailed treatments see [3, 34, 48]. As in the ODE case the jet variety is obtained from the equations of a system by replacing solutions and their derivatives in the system with their corresponding jet variables.

Let $R = (R^1, R^2, ..., R^N) = 0$ be a system of PDE with independent variables $x = (x_1, x_2, ..., x_n)$, and dependent variables $u = (u^1, u^2, ..., u^m)$. Let $\underset{k}{u}$ denote the kth order jet variables obtained by replacing the kth order derivatives of $u(x)$ by formal coordinates.

Definition 4 *The jet variety of a qth order system of pde* $R(x, u, \underset{1}{u}, ..., \underset{q}{u}) = 0$ *is*

$$\mathcal{R} = \{(x, u, \underset{1}{u}, ..., \underset{q}{u}) \in J^q : R(x, u, \underset{1}{u}, ..., \underset{q}{u}) = 0\}.$$

Again the prolonged graphs of solutions will lie in the jet variety of the system; with equality occurring if the system is locally solvable. We illustrate this extension through examples.

Example 5 *Consider the system of* PDE *with dependent variable* $u(x, y)$ *and independent variables* x, y*:*

$$R^1 = D_x u(x, y) - 1 = 0, \quad R^2 = (D_y u(x, y))^2 - u(x, y) = 0. \tag{17}$$

The jet space of this example is J^1 *since the system is first order.* J^1 *has formal coordinates* x, y, u, u_x, u_y *so locally* $J^1 \approx \mathbb{R}^5$*. The jet variety of* $R = 0$ *is obtained by replacing the solution* $u(x, y)$ *and its derivatives* $D_x u(x, y)$*,* $D_y u(x, y)$ *with formal coordinates* u*,* u_x*,* u_y*:*

$$\mathcal{R} := \{(x, y, u, u_x, u_y) \in J^1 : u_x - 1 = 0, (u_y)^2 - u = 0\}. \tag{18}$$

Figure 2 gives a graphical depiction of the jet variety. The x, y *dependence has been suppressed. Note that the prolonged graphs of any solutions must lie in the intersection of the varieties corresponding to the two equations.*

2.4 Geometric Differentiation in Jet space

We briefly describe the geometric operation of differentiation (called prolongation) in the Jet space. For a general treatment see [3, 34, 48].

Definition 6 *If the equations of* $R = 0$ *satisfy certain constant rank conditions described later*

$$\mathcal{D}\mathcal{R} = \{w \in J^{q+1} : R^l = 0, D_{x_i} R^l = 0\}.$$

Thus a single prolongation of the jet variety of a qth order system is simply the system together with all the equations resulting from taking total derivatives of each equation of the system with respect to each independent variable. This is a simplified version of the more complicated, but more efficient prolongation given by Pommaret [34]. Away from points where the rank conditions fail (*singular points*), the system of differential equations corresponding to $\mathcal{DR}$ has the same local analytic solutions as $R = 0$. This definition is independent of smooth invertible changes of coordinates $(x,u) \mapsto (\tilde{x},\tilde{u}) = (g(x), h(x,u))$. Indeed $D_{x_i} R^l = 0$ is equivalent to $D_{x^i}\tilde{x}^j D_{\tilde{x}^j} R^l = 0$, which by virtue of the invertibility of the transformation is equivalent to $D_{\tilde{x}^j} R^l = 0$. The constant rank conditions are required to ensure that differentiation on the variety is the same as differentiation of the equations of the variety. For examples of failure of the two to coincide see §2.7 and [42]. For systems of order q, higher order prolongations are defined by repeated application of $\mathcal{D}$. For example, $(\mathcal{D} \circ \mathcal{D})\mathcal{R} = \mathcal{D}(\mathcal{DR})$ is a variety in J^{q+2}.

Example 7 *The operators of formal total differentiation with respect to independent variables x and y for Example 5 are*

$$D_x := \frac{\partial}{\partial x} + u_x \frac{\partial}{\partial u} + u_{xx}\frac{\partial}{\partial u_x} + u_{xy}\frac{\partial}{\partial u_y} + \cdots \tag{19}$$

$$D_y := \frac{\partial}{\partial y} + u_y \frac{\partial}{\partial u} + u_{xy}\frac{\partial}{\partial u_x} + u_{yy}\frac{\partial}{\partial u_y} + \cdots \tag{20}$$

The prolongation of the system to order two of the example system (17) is

$$\begin{aligned} \mathcal{DR} &:= \{(x,y,u,u_x,u_y,u_{xx},u_{xy},u_{yy}) \in J^2 : R^1 = 0, R^2 = 0, \\ &\qquad D_x R^1 = 0, D_y R^1 = 0, D_x R^2 = 0, D_y R^2 = 0\}, \\ &= \{(x,y,u,u_x,u_y,u_{xx},u_{xy},u_{yy}) \in J^2 : u_x - 1 = 0, (u_y)^2 - u = 0, \\ &\qquad u_{xx} = 0, u_{xy} = 0, 2u_y u_{xy} - u_x = 0, 2u_y u_{yy} - u_y = 0\}. \end{aligned} \tag{21}$$

Thus $\mathcal{DR}$ is a variety in $J^2 = \mathbb{R}^8$.

2.5 Geometric Elimination in Jet space

The geometric operation of elimination projects a point the given jet space down to one in the jet space of one order less.

Definition 8 *A single projection (elimination) of the jet variety $\mathcal{R}$ in J^{q+1} is*

$$\mathcal{ER} := \{(x, u, \underset{1}{u}, ..., \underset{q}{u}) \in J^q : R(x, u, \underset{1}{u}, ..., \underset{q}{u}, \underset{q+1}{u}) = 0\}$$

Thus $\mathcal{E}$ maps a point in J^{q+1} to one in J^q by simply removing the jet variables of order $q+1$. In practise to carry out this projection one eliminates the jet variables of order $q+1$, to obtain equations of order q or less. Away from singular points the system of differential equations corresponding to $(\mathcal{E} \circ \mathcal{D})\mathcal{R}$ has the same local analytic solutions as $R=0$.

Example 9 *In Example 7 we obtained the variety of the prolonged system $\mathcal{D}\mathcal{R}$ as given by the system (21). The projection $\mathcal{E} : J^2 \to J^1$ maps a point $(x, y, u, u_x, u_y, u_{xx}, u_{xy}, u_{yy}) \in \mathcal{R} \subseteq \mathcal{R}$ to a point (x, y, u, u_x, u_y) in J^1:*

$$\mathcal{E}(x, y, u, u_x, u_y, u_{xx}, u_{xy}, u_{yy}) = (x, y, u, u_x, u_y). \tag{22}$$

To compute this projection we eliminate $u_{xy}=0$ yielding the projected system in J^1 as

$$(\mathcal{E} \circ \mathcal{D})\mathcal{R} = \{(x, y, u, u_x, u_y) \in J^1 : u_x - 1 = 0, (u_y)^2 - u = 0, u_x = 0\}. \tag{23}$$

Thus $(\mathcal{E} \circ \mathcal{D})\mathcal{R}$ is empty since the varieties corresponding to the three equations have no common point of intersection (see Figure 3). Since the system of differential equations corresponding to $(\mathcal{E} \circ \mathcal{D})\mathcal{R}$ has the same solutions as $R=0$ it follows that the original system has no solution. This fact was not obvious from the original system, and was uncovered by the process of prolongation and projection. In particular the original system was not locally solvable since its jet variety contained points through which no solutions passed.

In general $(\mathcal{E} \circ \mathcal{D})\mathcal{R} \subseteq \mathcal{R}$ with inequality being the usual case. It is desirable to make a system locally solvable since otherwise it implies that there are missing equations (such as $u_x = 0$ in the above example) and the solution variety differs from the jet variety (i.e. $\mathcal{R}_{\mathrm{sol}} \subset \mathcal{R}$).

In an attempt to make this work accessible we have often introduced simplified notation. For example, projection from J^{p+1} to J^p is often written $\pi_p^{p+1} : J^{p+1} \to J^p$. We have tried to avoid this and other indicial notations since they can complicate the description of the algorithms given in the paper. Thus the definition of the projection is bound to the space on which it acts.

The above examples suggest the following simple algorithm to attempt to make a system locally solvable.

Algorithm 1

> **Input** $\mathcal{R}$
> **while** $\mathcal{R} \neq (\mathcal{E} \circ \mathcal{D})\mathcal{R}$ **repeat** $\mathcal{R} := (\mathcal{E} \circ \mathcal{D})\mathcal{R}$
> **Output** $\mathcal{R}$

2.6 Definition of Symbol

Although Algorithm 1 succeeds in making many ODE and some PDE locally solvable it suffers from some defects which can be remedied by the introduction of the *symbol*. We will informally describe the symbol and use examples to indicate how it can be used to detect singular solutions and integrability conditions.

Definition 10 *Denote the qth order jet variables* $\underset{q}{u}$ *of a qth order system of* PDE *by* $w^1, w^2, \ldots$. *The symbol of* $R = 0$ *is the jacobian matrix with respect to the system's highest derivatives:*

$$\textit{Symbol } R := \left[\frac{\partial R^j}{\partial w^i}\right] = \frac{\partial R}{\partial \underset{q}{u}}. \tag{24}$$

Thus the symbol is the linearized highest order part of the equation.

2.7 Symbol and singular solutions

The following example has been provided by E. Mansfield. Consider the first order ODE

$$R(x, u, u_x) := x u_x^2 - 3 u u_x + 9x^2 = 0 \tag{25}$$

which has $w^1 = u_x$. Then from (24)

$$\text{Symbol } R := \left[\frac{\partial R}{\partial u_x}\right] = [2xu_x - 3u]\,. \tag{26}$$

If we compute

$$\begin{aligned} D_x R &= (2xu_x - 3u)u_{xx} - 2u_x^2 + 18x = 0 && (27) \\ D_x^2 R &= (2xu_x - 3u)u_{xxx} + \text{lower order terms} = 0 && (28) \end{aligned}$$

we see that the Symbol is the coefficient of the highest derivative in the equations defining the prolongations $\mathcal{DR}$,$\mathcal{D}^2\mathcal{R}$, ... of $\mathcal{R}$. To compute a formal power series solution of the ODE:

$$u(x) = u^0 + u_x^0(x - x^0) + u_{xx}^0(x - x^0)^2/2! + \ldots \tag{29}$$

requires expressions for the derivatives u_x, u_{xx},... at x^0 (i.e. we need $u_x^0 = u_x(x^0)$, $u_{xx}^0 = u_{xx}(x^0)$ etc). These can be computed recursively from lower order derivatives at points where the $2xu_x - 3u \neq 0$, that is at points where the symbol has constant nonzero rank.

Two cases arise:

Case 1: $\frac{\partial R}{\partial u_x} = 2xu_x - 3u \neq 0$, $xu_x^2 - 3uu_x + 9x^2 = 0$,

Case 2: $\frac{\partial R}{\partial u_x} = 2xu_x - 3u = 0$, $xu_x^2 - 3uu_x + 9x^2 = 0$.

Case 1 is equivalent to an explicit ODE, i.e. to a DAE of index 0 with general solution $u(x; c) = c + x^3/c$ (see the thin curves in Figure 3). In Case 2, following the Rabier-Rheinbold algorithm, we eliminate the derivative u_x to find that

$$u^2 - 4x^3 = 0 \tag{30}$$

which is the bold curve in Figure 3. Next we differentiate this equation to obtain $2uu_x - 12x^2 = 0$. Elimination amongst the three first order equations yields no new zeroth order constraints. So the Rabier-Rheinboldt algorithm terminates after one step. This example illustrates how cases can arise when the symbol fails to have constant rank.

Every point of the singular solution $u^2 - 4x^3 = 0$ is touched tangentially by a solution from the family constituting the general solution. Two solutions of the ODE pass through every point (x^0, u_x^0) to the left of the singular solution $u^2 - 4x^3 = 0$ in Figure 3. Thus the initial condition $u(x^0) = u^0$ is not enough to specify a unique solution of the ODE. This non-uniqueness is removed by specifying u_x^0 and is illustrated by the jet variety of the ODE having two sheets (corresponding to the positive and negative roots which arise when solving for u_x from (25)). Graphs of solutions lying in the jet variety are shown in Figure 4. An algorithm and a general theory for singular solutions of a general first order ODE has been given by E. Hubert [21].

Finally we mention an alternative which instead of introducing cases, embeds the jet variety $\mathcal{R} \subseteq J^q$ into a larger space K^q which we call the case covering space. Thus $\mathcal{R}$ is embedded in $\mathcal{S} \subseteq K^q$. For example, consider the ODE $(u-1)(u_x - u) = 0$, which has pivot $(u-1) = p$. Introduce K^1 with jet coordinates (x, u, u_x, p, p_x). The initial form of the case covering system $S = 0$ is given by $p(u_x - u) = 0, (u-1) = p$. A single prolongation of $S = 0$ by the CK algorithm yields $S = 0, p(u_{xx} - u_x) + p_x(u_x - u) = 0, u_x = p_x$, and then projection yields the equation $u_x = p_x$. The CK algorithm then terminates with the output system $p(u_x - u) = 0, (u-1) = p, u_x = p_x$. In particular this system covers both cases $p \neq 0$ and $p = 0, p_x = 0$. This process is closer to the geometrical CK algorithm, where all cases are prolonged at the same time.

2.8 Integrability conditions and formal integrability

An integrability condition of a system of PDE from an elementary point of view results from the equality of mixed partial derivatives of the system.

A geometric method for determining when to test for integrability conditions is based on properties of the system's highest order part (i.e. its Symbol). This is undoubtedly the most complicated part of the geometric jet space methods for PDE.

In this section we define integrability conditions from the point of view of jet space methods and what it means for a system to have its integrability conditions satisfied (i.e. to be *formally integrable*).

Consider $R = 0$ given by

$$u_{xx} - yu = 0, \quad u_{yy} = 0 \tag{31}$$

where x, y are the independent variables. Carrying out Algorithm 1

$$\mathcal{DR} := \quad \{(x, y, u, u_x, u_y, ..., u_{xyy}, u_{yyy}) \in J^3 : u_{xx} - yu = 0, u_{yy} = 0,$$

$$u_{xxx} - yu_x = 0, u_{xyy} = 0, u_{xxy} - yu_y - u = 0, u_{yyy} = 0\} \quad (32)$$

defines a six dimensional submanifold of $J^3 = \mathbb{R}^{12}$. No new equations of order 2 or less can be obtained by eliminating jet variables of order 3 from (32). Thus $(\mathcal{E} \circ \mathcal{D})\mathcal{R} = \mathcal{R}$ and the algorithm terminates immediately with its output in the form (31). But amongst the equations of $\mathcal{D}^2\mathcal{R}$ are

$$u_{xxyy} - yu_{yy} - 2u_y = 0, \quad u_{xxyy} = 0. \quad (33)$$

So the equation set describing $(\mathcal{E} \circ \mathcal{D})\mathcal{D}\mathcal{R}$ contains

$$-yu_{yy} - 2u_y = 0 \quad (34)$$

which is genuinely new since it lowers the dimension of the submanifold defined by $\mathcal{D}\mathcal{R}$. Thus $(\mathcal{E} \circ \mathcal{D})\mathcal{D}\mathcal{R} \subset \mathcal{D}\mathcal{R}$ and $\mathcal{D}\mathcal{R}$ and $\mathcal{R}$ are not locally solvable since there are points in $\mathcal{R}$ and $\mathcal{D}\mathcal{R}$ through which no graphs of solutions pass. Thus Algorithm 1 does not always detect all contraints for PDE. The problem here is that $(\mathcal{E} \circ \mathcal{D})\mathcal{R} = \mathcal{R}$ was not enough to ensure that $(\mathcal{E} \circ \mathcal{D})\mathcal{D}\mathcal{R} = \mathcal{D}\mathcal{R}$. The source of the problem was the non-vanishing integability condition resulting from equating the cross partials u_{xxyy}, u_{yyxx}. A system which contains all its integrability conditions is said to be formally integrable.

Definition 11 *A system of differential equations $R = 0$ is said to be formally integrable if $(\mathcal{E} \circ \mathcal{D})\mathcal{D}^k\mathcal{R} = \mathcal{D}^k\mathcal{R}$ for $k = 0, 1, 2, ...$*

To obtain local solvability the system must be prolonged far enough (i.e. k in the definition above must be big enough), to ensure that such integrability conditions are all captured.

3 Cartan-Kuranishi Algorithm

An algorithm that delivers an existence and uniqueness theorem for systems of PDE should make such systems formally integrable. If they are not formally integrable then there are undetermined restrictions on the initial values (or more generally the values of derivatives at a point x^0) necessary for existence of solutions.

The corner-stone of the CK algorithm which addresses these difficulties is the following theorem [34]:

Theorem 12 *If a system $R = 0$ has involutive symbol and $(\mathcal{E} \circ \mathcal{D})\mathcal{R} = \mathcal{R}$ then it is formally integrable.*

A formally integrable system which has involutive symbol is called an *involutive system.* Involutive systems are locally solvable and allow an existence and uniqueness theorem (the Cartan-Kähler theorem [23]).

In the abstract theory, a symbol is involutive if certain homology groups vanish [17, 53]. This is undoubtedly the most complicated aspect of the CK algorithm below. Pommaret [34] has given a test in local coordinates, that involves simple rank computations on the symbol. This test, although it has been implemented in the symbolic language Axiom [50], is still complicated by the existence of certain singular coordinate systems, called delta irregular systems. We have postponed the description of this test to the appendix.

The following algorithm addresses the deficiencies of Algorithm 1 and outputs an involutive system. It is based on the works of Cartan [8], Ehresmann [10], Goldschmidt [17] and Pommaret [34].

Algorithm 2: Cartan-Kuranishi Algorithm

> **Input** $\mathcal{R}$
> **repeat**
>   **while** $\mathcal{R} \neq (\mathcal{E} \circ \mathcal{D})\mathcal{R}$ **repeat** $\mathcal{R} := (\mathcal{E} \circ \mathcal{D})\mathcal{R}$
>   **while** Symbol $\mathcal{R}$ is not involutive **repeat** $\mathcal{R} := \mathcal{D}\mathcal{R}$
> **until** Symbol $\mathcal{R}$ is involutive and $\mathcal{R} = (\mathcal{E} \circ \mathcal{D})\mathcal{R}$
> **Output** $\mathcal{R}$

The sequence of dimensions of $(\mathcal{E} \circ \mathcal{D})^j\mathcal{R}$ is strictly decreasing and bounded above by the finite number of jet variables in a system $\mathcal{R}$ which enters the first loop. Therefore the first loop terminates. A homological proof of termination of the second loop with a system $\mathcal{D}^k\mathcal{R}$ which has involutive symbol is given in [34], together with a (very large) bound on k as a function of the order of $\mathcal{R}$ and the number of its independent and dependent variables. The termination of the entire algorithm can be given a sketchy justification using the ideas of Tresse [56]. Cartan, one of the great geometers of this century, tried but failed to rigorously prove the termination of the above algorithm. Finally in a very deep work Kuranishi [27], proved the termination of the above algorithm. If the system yields a functional relation amongst its independent variables it has no solutions. By convention we define this to be the null involutive system.

Example 13 *We apply the CK algorithm to the system $\mathcal{R}$ given by (31) in the previous section. In the appendix we show that Symbol $\mathcal{R}$ is not involutive, but that Symbol $\mathcal{D}\mathcal{R}$ is involutive where $\mathcal{D}\mathcal{R}$ is given by (32). Computation of $(\mathcal{E}\circ\mathcal{D})\mathcal{D}\mathcal{R}$ yields the new equation $-u_{yy}-2u_y = 0$ and with a little simplification the equations can be written*

$$u_y = 0, u_{xx} - yu = 0, u_{yy} = 0, u_{xxx} - yu_x = 0, u_{xxy} - u = 0, u_{xyy} = 0, u_{yyy} = 0.$$

Subsequent applications of $(\mathcal{E} \circ \mathcal{D})$ yield the following results: $(\mathcal{E} \circ \mathcal{D})^2\mathcal{D}\mathcal{R}$ yields the new equation $u_{xy} = 0$, $(\mathcal{E} \circ \mathcal{D})^3\mathcal{D}\mathcal{R}$ yields the new equation $u_{xxy} = 0$ and then $u = 0$. The CK algorithm then terminates with $(\mathcal{E} \circ \mathcal{D})^4\mathcal{D}\mathcal{R}$ and output system:

$$u = u_x = u_y = u_{xx} = u_{xy} = u_{yy} = u_{xxx} = u_{xxy} = u_{xyy} = u_{yyy} = 0.$$

It can also be checked that this has involutive symbol using the techniques described in the Appendix.

The reader should be able to much more quickly arrive at the same result, namely that the only solution is $u = 0$, by a calculation based on the more usual method of equating cross partials (see the Appendix). One of the sources of the laborious nature of the above calculation is the insistence that it be broken down into coordinate-independent operations. See Seiler [50], for work on improving the efficiency of the Cartan-Kuranishi process and Mansfield [30, 31] for a deep result linking the two approaches.

3.1 Constant rank condition

In order to apply this geometric algorithm as an algorithm acting upon a system of equations, the system must be adjusted after each projection, if possible to an equivalent one satisfying the constant rank conditions discussed in this section.

Definition 14 *Let $R(x, z) = 0$ be a qth order system with independent variables x and jet variables $z = (u, \underset{1}{u}, ..., \underset{q}{u})$ corresponding to dependent variables and their derivatives. Then this system satisfies the constant rank conditions at $(x^0, z^0) = (x^0, u^0, \underset{1}{u}^0, ..., \underset{q}{u}^0) \in \mathcal{R} \subseteq J^q$ if there exist nonzero constants α, β such that*

$$rank\frac{\partial R(x,z)}{\partial(x,z)} = \alpha = rank\frac{\partial R}{\partial z},\ \ rank\frac{\partial R}{\partial \underset{q}{u}} = \beta \tag{35}$$

in a neighbourhood of (x^0, z^0) in the usual Euclidean norm.

Examples of the constant rank condition and its implications for algorithms are given in [41, 42]. The points which fail the constant rank condition have to be treated in separate runs of the algorithm (for example see Case 2 in §2.7). A tree of cases based on inequations or pivots results and must be treated in other runs. For polynomial systems this tree is finite.

If $R = (u_x + u)(u_x - u) = 0$ then the first equality in (35) is $\mathrm{rank}\frac{\partial R(x,u,u_x)}{\partial(x,u,u_x)} = \mathrm{rank}\,[0 - 2u 2u_x] = 0$ if $u_x = u = 0$ and is 1 otherwise. The constant rank condition does not hold along the intersection of $u_x - u = 0 = u_x + u$ since $R = 0$ is not a manifold there (i.e. there is no smooth change of coordinates that converts the locus of $R = 0$ near this intersection into a hyperplane). When the first equality in (35) holds it implies that $\mathcal{R}$ is a submanifold of J^q.

Consider $R = \{u_x - u^2 - 1 = 0, u_x - u^2 - x = 0\}$, then $\mathrm{rank}\frac{\partial R(x,u,u_x)}{\partial(x,u,u_x)} = 2 = \alpha$ so $\mathcal{R}$ is a submanifold of J^1. However $\mathrm{rank}\frac{\partial R(x,u,u_x)}{\partial(u,u_x)} = 1 \neq 2$ thus the first condition fails. The reason for the failure is the existence of a functional relation $x - 1 = 0$ on the independent variable x. The equation $R = u - u_x^2$ satifies all the conditions, except the last which fails in the case $u_x = u = 0$.

Finally consider $R = (u_x - u)^2(u_x + u)^3 = 0$. Then rank$\frac{\partial R(x,u,u_x)}{\partial(x,u,u_x)} = 0$ at $(x^0, u^0, u_x^0) \in \mathcal{R}$ and is 1 elsewhere. So the rank is not constant in any neighbourhood of (x^0, u^0, u_x^0). This problem can be fixed by removing the multiplicities to obtain a system which satisfies the constant rank conditions, except at the intersection (see the example above which has the same locus).

If these constant rank conditions hold then the following three geometric conditions hold. The first equality implies that the tangent space exists at the point (i.e. the variety is a submanifold of J^q in a neighbourhood of the point). The second equality implies that there is a neighbourhood for which this submanifold projects diffeomorphically onto the space of independent variables (i.e. the tangent space is transverse to the fibres over the space of independent variables). This is called the *independence condition* since it implies that there are no functional relations amongst the independent variables. The last condition implies that the dimension of one projection of the tangent space is constant in a neighbourhood of $(x^0, u^0, \underset{1}{u}^0, ..., \underset{q-1}{u}^0)$.

A point (x^0, z^0) on $\mathcal{R}$ is non-singular if it satisfies these geometrical conditions. In the neighbourhood of a non-singular point it is always possible to find an equivalent representation of the equations so that the constant rank conditions hold. The third constant rank condition has the interpretation that β of the qth order jet variables can be expressed in terms of the other jet variables.

3.2 Effective implementation for linear DAE

The reader unfamiliar with Gröbner bases may find it helpful to read this section before the one that follows. Gauss reduction leads to easy implementation of the CK algorithm for linear systems of ODE [50]. There are several implementations of the Rabier-Rheinboldt algorithm for linear DAE [35, 32].

The test given in the Appendix easily shows that systems of ODE always have involutive symbol. Thus for ODE the CK algorithm simply iterates the step $\mathcal{R} := (\mathcal{E} \circ \mathcal{D})\mathcal{R}$ until termination.

Consider linear systems having coefficients which are analytic functions of the independent variable x. Order the jet variables $\underset{q}{u}, ..., \underset{1}{u}$, and the corresponding columns of the system's coefficient matrix from left to right in the same way. Gauss reduction of the system, eliminating higher order jet variables in favour of lower ones, yields a system which is equivalent apart from the non-vanishing of certain pivots that occurred in the process. For example $xu_x + u = 0, u_x + xu = 0$ has gauss reduced form $u_x = 0, u = 0$ apart from the pivot condition $x^2 - 1 \neq 0$. The resulting system is inconsistent if it contains a nontrivial equation of the form $f(x) = 0$. If it is consistent then it satisfies all the constant rank conditions away from the relatively small set of values of x for which the pivots vanish.

Thus formal differentiation on the equations is the same as geometric differentiation away from points where the pivots vanish.

After $\mathcal{DR}$ is calculated, then Gauss elimination of the jet variables determines whether there are any new relations of order q or less (i.e. implements the test $(\mathcal{E} \circ \mathcal{D})\mathcal{R} = \mathcal{R}$).

Notice for this process to be guaranteed to be effective on a computer, we need to be able to recognize whether an expression appearing in a pivot is zero or nonzero. This is certainly guaranteed if the coefficients are rational functions of x with rational number coefficients. Only one case is produced by such an algorithm.

3.3 Effective implementation for polynomially nonlinear DAE

Effective implementation of the CK algorithm for polynomially nonlinear DAE is achieved by use of a variation of Buchberger's Gröbner basis algorithm. This algorithm can be regarded as a generalization of the Gauss algorithm to polynomial equations. Each step of the Gröbner basis algorithm consists of elementary algebraic operations. There has been an explosion in applications of these techniques. See [2] for a detailed treatment. Most importantly Gröbner bases enable us to decide whether a system has solutions (i.e. is consistent) and provides a reduction procedure for testing whether one polynomial system is a geometrical consequence of another. The version of the Gröbner basis algorithm we use, RadGBasis, is one which finds a Gröbner basis for the radical of the polynomial ideal generated by a system of polynomials (see [16] for an algorithm over the complex field and [1, 12, 46] for an algorithm over the reals). Unlike the usual Gröbner basis algorithm, it enables the first constant rank condition in (35) to be satisfied, and enables the test for whether a system is a consequence of another to be geometrical.

As shown in the Appendix, systems of DAE always have involutive symbol so the CK algorithm simply repeats $\mathcal{R} := (\mathcal{E} \circ \mathcal{D})\mathcal{R}$ until equality.

We apply the CK algorithm to systems $R = 0$ which are polynomially nonlinear in a single independent variable x, and their jet variables $u, \underset{1}{u}, ..., \underset{q}{u}, ...$, and which also have rational number coefficients. The Gröbner basis algorithm we use, RadGBasis, takes as input such systems, together with an ordering of the variables, which ranks these variables in so-called increasing lexicographic order $x \prec u \prec \underset{1}{u} \prec ... \prec \underset{q}{u} \prec ...$. This order, in a analogous fashion to the Gauss algorithm, is used by RadGBasis to eliminate higher order jet variables in terms of lower ones.

We will first briefly describe the main aspects for effective implementation.

We first apply the RadGBasis algorithm to the input system $R = 0$, to obtain the system in the form $\mathrm{RadGBasis}(R) = 0$. In similar fashion to the Gauss algorithm, the use of the lexicographic order ensures that the system is inconsistent if and only if $\mathrm{RadGBasis}(R) = 0$ contains an equation of the form $f(x) = 0$ or $1 = 0$ [2]. If such a relation is detected then the algorithm terminates with the system being inconsistent.

If the system is consistent we calculate $\mathcal{D}\mathcal{R}$ by symbolically differentiating the equations of $\text{RadGBasis}(R) = 0$. The resulting system is linear in its $(q+1)$th order jet variables, with coefficients which in general are nonlinear functions of lower order jet variables.

The Gauss algorithm is applied to the $(q+1)$th order equations to obtain the qth order system $(\mathcal{E} \circ \mathcal{D})\mathcal{R}$. During this elimination divisions by coefficients of the $(q+1)$th order jet variables may have occurred. The resulting pivots, are stored in sequence, in a pivot list $\Lambda \neq 0$. Also, the elimination can be carried out using fraction-free elimination so that $(\mathcal{E} \circ \mathcal{D})\mathcal{R}$ is polynomially nonlinear.

We now have to decide whether $\mathcal{R} = (\mathcal{E} \circ \mathcal{D})\mathcal{R}$, that is whether $(\mathcal{E} \circ \mathcal{D})\mathcal{R}$ is a consequence of $\mathcal{R}$. When the systems are linear this question can be answered by reducing $(\mathcal{E} \circ \mathcal{D})\mathcal{R}$ modulo $\mathcal{R}$ (or equivalently simplification by substitution from $\mathcal{R}$). A well-known generalization [2] which we call Gröbner reduction, exists for polynomial systems. If when Gröbner reduction is applied to reduce $(\mathcal{E} \circ \mathcal{D})\mathcal{R}$, it reduces to 0 modulo $\mathcal{R}$ then $(\mathcal{E} \circ \mathcal{D})\mathcal{R}$ is a consequence of $\mathcal{R}$. In that case termination occurs with output $\mathcal{R}$ and pivots $\Lambda \neq 0$. If not, then $\mathcal{R}$ is replaced with $(\mathcal{E} \circ \mathcal{D})\mathcal{R}$ and the process is repeated.

This process, is continued, and terminates with either a system which is inconsistent, or a system which is involutive and a list of inequations $\Lambda \neq 0$.

Next we adjoin one of the pivots, now set to zero, as an equation to the original system. The algorithm then returns a new system and a new output list of pivots. A binary tree of cases results, based on decisions as to whether pivots vanish or not. Each path in the tree corresponds to an ascending chain of polynomial ideals in a polynomial ring and hence is of finite length. Termination follows from Konig's Lemma [2] (see [47] for the partial differential case).

To make sure that a formal (symbolic) derivative of the equations performed by a computer corresponds to a geometric differentiation we need the first and second constant rank conditions in (35) to be satisfied.

The example $R = (u_x - u)^2(u_x + u)^3 = 0$ of the previous section indicates that spurious multiplicities lead to the failure of constant rank. However the Gröbner basis of a system does not remove these multiplicities. For example the Gröbner basis of the single equation above leaves it unaltered. Unfortunately an approach based on explicit factorization is not effective in general in view of Galois' famous result that it is not always possible to explicitly solve equations of order greater than four. There are however effective methods which produce the multiplicity-free (square-free) form of a polynomial without the knowledge of its factors [16]. The generalization of this idea to systems of polynomial equations yields a notion of a multiplicity-free system (in algebraic terminology, such a system is said to be a generating set for the radical of the ideal generated by the system).

In [42] we show that at non-singular points of the variety, the constant rank condition which is the first equality in (35) will be satisfied by $\text{RadGBasis}(R) = 0$. The version of the Gröbner basis algorithm we use, RadGBasis, is one which finds a Gröbner basis for the radical of the polynomial ideal generated by a system of polynomials (see [16, §9] for an algorithm over the complex field and [1, 12, 46] for an algorithm over the reals).

If no relation of form $f(x) = 0$ is found, then the constant rank condition given by the second equality in (35) is satisfied at non-singular points of the variety by $\text{RadGBasis}(R) = 0$.

These conditions are enough to ensure that the geometric prolongation $\mathcal{DR}$ is equivalent to the one obtained by taking symbolic derivatives. The nonvanishing of the pivots, also ensures that the last constant rank condition in (35) is satisfied. The requirement that $R = 0$ is a Gröbner basis for the radical at each iteration can be weakened to requiring that $R = 0$ is a generating set for the radical, but not to radical ideal membership testing [47]. The calculations in this paper generally use generating sets instead of Gröbner bases since they often have fewer equations than their associated Gröbner bases.

An alternative method for achieving constant rank is to row reduce the Jacobians, with the assumption that the pivots do not vanish on $R = 0$. The cases where one or more of the pivots vanish are then treated in separate runs. A combination of this method and the use of the dimension determination algorithm of [2] is used in Seiler's implementation of the CK algorithm [49, 50].

The above discussion also applies to PDAE. The only extra condition that has to be satisfied in that case is that the Symbol of the system is made involutive. Because of its technical nature we have left that discussion to the appendix. The version of the CK algorithm given in this paper was discussed in [41]. The reader is warned is that although these algorithms are terminate in a finite number of steps, this number can be very large. Practical implementation of such methods is a significant challenge (see Seiler [50] for progress in this direction).

4 Connection between the Cartan-Kuranishi and Rabier-Rheinboldt Algorithms

Using the operations $\mathcal{D}$ and $\mathcal{E}$ Rabier and Rheinboldt's algorithm [37] can be expressed in the following form:

Algorithm 3: Rabier-Rheinboldt Algorithm

Input $\mathcal{R}$
while $\mathcal{R} \neq \mathcal{R} \cap (\mathcal{D} \circ \mathcal{E})\mathcal{R}$ **repeat** $\mathcal{R} := \mathcal{R} \cap (\mathcal{D} \circ \mathcal{E})\mathcal{R}$
Output $\mathcal{R}$

In order to apply this geometric algorithm, after input and each projection, the equations should be adjusted so that the constant rank conditions (35) are satisfied at non-singular points of the variety.

Rabier-Rheinboldt state their method for non-autonomous first order DAE and extend it to autonomous DAE by using the familiar trick of adjoining an equation $w_t = 1$ to $R = 0$. Our representation of their algorithm enables it to be in principal applied to non-autonomous DAE of order greater than one and also to PDAE.

In what follows we consider autonomous analytic first order DAE $R(u, u_t) = 0$ for $u(t)$. The constant rank conditions (35) for this case reduce to

$$\frac{\partial R}{\partial u_t} = \alpha, \quad \frac{\partial R(u, u_t)}{\partial (u, u_t)} = \beta \tag{36}$$

where α and β are nonzero constants in a neighbourhood of $(t^0, u^0, {u_t}^0) \in \mathcal{R}$.

Our aim is to compare the application of the Cartan-Kuranishi and Rabier-Rheinboldt algorithms to an input system of autonomous DAE $R^{(0)}(u(t), u_t(t)) = 0$ with corresponding jet variety $\mathcal{R}^{(0)}$. We will demonstrate that they are equivalent. The symbol of an ODE is always involutive [34] (also see the Appendix) so the only difference between the two algorithms in this case is in the form of their iteration step. The form of the iteration step for the CK algorithm is $\mathcal{R} := (\mathcal{E} \circ \mathcal{D})\mathcal{R}$ and for the Rabier-Rheinboldt algorithm $\mathcal{R} := \mathcal{R} \cap (\mathcal{D} \circ \mathcal{E})\mathcal{R}$.

Suppose we apply the Rabier-Rheinboldt algorithm to the input DAE and that a singular DAE does not result. Then a sequence of submanifolds $J^1 \supseteq \mathcal{R}^{(0)} \supseteq \mathcal{R}^{(1)} \supseteq \ldots$ results [37], with $\bigcap_i \mathcal{R}^{(i)} \neq \emptyset$. Following Reich [38], Rabier and Rheinboldt [37] define the *differential index* of the DAE as the smallest integer ν such that $\mathcal{R}^{(\nu)} = \mathcal{R}^{(\nu+1)}$:

$$J^1 \supseteq \mathcal{R}^{(0)} \supset \mathcal{R}^{(1)} \supset \cdots \supset \mathcal{R}^{(\nu)} = \mathcal{R}^{(\nu+1)}. \tag{37}$$

They show that $\mathcal{R}^{(\nu)} = \bigcap_i \mathcal{R}^{(i)}$, and that there are local solutions $u(t)$ passing through any $(t^0, u^0, u_t^0) \in \mathcal{R}^{(\nu)}$.

Kuranishi's famous prolongation theorem [27] shows that the application of the CK algorithm to $\mathcal{R}^{(0)}$ results in the sequence of submanifolds

$$J^1 \supseteq \mathcal{R}^{(0)} \supset \mathcal{C}^{(1)} \supset \mathcal{C}^{(2)} \supset \mathcal{C}^{(3)} \supset \cdots \supset \mathcal{C}^{(\mu)} = \mathcal{R}^{(\mu+1)} \tag{38}$$

with an identical existence and uniqueness result to that of Rabier and Rheinboldt. Indeed we have:

Theorem 15 *Let $R^{(0)}(u(t), u_t(t)) = 0$ be an autonomous analytic* DAE *which has differential index ν. Then $\nu = \mu$ and $\mathcal{C}^{(k)} = \mathcal{R}^{(k)}$, $k = 1, \ldots, \nu$.*

Proof:

Suppose that the input DAE is

$$R(t, u, u_t) = 0$$

Then the last constant rank condition in (36), is that rank$\frac{\partial R}{\partial u_t} = \beta$ in a neighbourhood of $(t^0, u^0, u_t^0) \in \mathcal{R}$. Thus by the Implicit Function Theorem, there are functions f, G such that the equations may be written in the local solved-form

$$v_t - f(t, u, \tilde{v}_t) = 0, \; G(t, u) = 0. \tag{39}$$

Here the derivatives u_t, have been partitioned into two sets, $u_t = (v_t, \tilde{v}_t)$. The solved-form has β equations, and calculating ranks of the above system verifies that G can not depend on $\tilde{v}_t$.

The second constant rank condition in (36), is that rank$\frac{\partial R(t,u,u_t)}{\partial(u,u_t)} = \alpha$ in a neighbourhood of $(t^0, u^0, u_t^0) \in \mathcal{R}$. Applying this condition to (39), implies that there is a function g, and a partition of coordinates $u = (z, \tilde{z})$, such that $G(t, u) = 0$ has the local solved-form $z - g(t, \tilde{z}) = 0$. This solved-form has $\alpha - \beta$ equations. The remaining constant rank condition, rank$\frac{\partial R(t,u,u_t)}{\partial(t,u,u_t)} = \alpha$ in a neighbourhood of $(t^0, u^0, u_t^0) \in \mathcal{R}$ excludes the possibility of additional relations amongst $t, \tilde{z}$. Thus the system can be written in the semi-explicit form

$$v_t - f(t, u, \tilde{v}_t) = 0,\ z - g(t, \tilde{z}) = 0. \tag{40}$$

For autonomous DAE there is no explicit dependence on t.

Applying a projection in the first step of the Rabier-Rheinboldt algorithm yields $\mathcal{E}\mathcal{R}$ given by $z - g(t, \tilde{z}) = 0$. Then $\mathcal{R} \cap (\mathcal{D} \circ \mathcal{E})\mathcal{R}$ is defined by (40), together with $z_t - g_t - g_{\tilde{z}}\tilde{z}_t = 0$. Applying the CK algorithm, $\mathcal{D}\mathcal{R}$ is given by (40), together with

$$v_{tt} - f_t - f_u u_t - f_{\tilde{v}_t}\tilde{v}_{tt} = 0,\ z_t - g_t - g_{\tilde{z}}\tilde{z}_t = 0.$$

Projection yields the same system as that given by the Rabier-Rheinboldt iteration.

It is understood in this process that the system is adjusted at each step so that the constant rank conditions hold at as many points as possible. Writing the system in the semi-explicit form (40) at each step and reapplying the argument above completes the proof. ■

Example 16 *The equivalence derived in Theorem 15 does not hold for* PDE. *For example consider the system $u_x = yu$, $u_y = u$ on J^1 with independent variables x, y and dependent variable u. Then projection $\mathcal{E} : J^1 \to J^0$ does not produce any new equations. Hence $(\mathcal{D} \circ \mathcal{E})\mathcal{R} = \mathcal{R}$ and the Rabier-Rheinboldt algorithm terminates without any change to $\mathcal{R}$. However $\mathcal{D}\mathcal{R}$ has amongst its equations $u_{xy} = yu_y + u$ and $u_{xy} = u_x$. Thus $(\mathcal{E} \circ \mathcal{D})\mathcal{R}$ has the equation $yu_y + u - u_x = 0$ which simplifies to $u = 0$ by virtue of the original system. Hence $(\mathcal{E} \circ \mathcal{D})\mathcal{R} \neq \mathcal{R} \cap (\mathcal{D} \circ \mathcal{E})\mathcal{R}$. Thus the CK algorithm can determine restrictions for* PDAE *not detectable by the Rabier-Rheinboldt algorithm.*

5 Calculation of the Differential Index

We restrict ourselves to polynomial systems with rational coefficients. Our effective implementation of the CK algorithm yields an effective determination of the differential index. However, as will be discussed, there may be components of the DAE with lower differential index and more sophisticated methods are required [54].

With one exception, the same techniques that were used to make an effective implementation of the CK algorithm for polynomially nonlinear DAE, can be used to produce an effective implementation of the Rabier-Rheinboldt algorithm. The systems to which the projection $\mathcal{E}$ is applied are generally nonlinear in their highest derivatives. Instead of using the Gauss algorithm, RadGBasis applies the projection by using an elimination ordering which ranks all derivatives of highest order lexicographically greater than those of strictly lower order. Cases result from the pivot conditions in the matrix of the Symbol of the system. This technique can also be used as an alternative in the case where the highest derivatives occur linearly, and may result in fewer cases.

We apply the Rabier-Rheinboldt algorithm to the system

$$\begin{aligned} &u_t + 2v_t - 2u - 3v = 0, \quad 2u_t + 3v_t = 0, \\ &u^3 + 6u^2v + 12uv^2 + 8v^3 - 4u^2 - 12uv - 9v^2 = 0, \\ &u^3 + 6u^2v + 12uv^2 + 8v^3 + 4u^2 + 12uv + 9v^2 = 0. \end{aligned} \tag{41}$$

In the first step, RadGBasis reduces the constraint equations to the equivalent form $u = 0, v = 0$. Differentiating these equations, does not introduce any new equations, so the algorithm terminates and the system has differential index 0. The output system can be written as $u_t = 0, v_t = 0, u = 0, v = 0$.

Consider the case where an algebraic Gröbner basis algorithm GBasis is used instead of RadGBasis. It can be checked that 4 differentiation-elimination steps are required for termination, with a very complicated output system with an apparently different differential index of 3.

To clarify this apparently paradoxical behavior consider the class of systems:

$$u_t = v, v_t = 0, u^k = 0, v^{k-1} = 0, \tag{42}$$

where $k > 1$ is a positive integer. An ideal I is radical if $z^n \in I$ implies $z \in I$. Hence RadGBasis reduces $u^k = 0, v^{k-1} = 0$ to $u = 0, v = 0$. Consequently the class above has differential index 0.

We now apply Gröbner bases, instead of RadGBasis to (42). The constraint is already a Gröbner basis in any ordering. Differentiating the constraint yields $ku^{k-1}u_t = 0$, $(k-1)v^{k-2}v_t = 0$. Elimination using $u_t = v$, $v_t = 0$ yields the equation $u^{k-1}v = 0$ which is new in the sense that it is not in the ideal generated by $u^k = 0, v^{k-1} = 0$. Similarly the next differentiation elimination step yields the equation $u^{k-2}v = 0$, and the calculation terminates after $k + 1$ steps with an apparent differential index of k and the output system:

$$u_t = v, v_t = 0, u^k = 0, v^{k-1} = 0, u^{k-1}v = 0, u^{k-2}v = 0, ..., uv = 0, v = 0. \tag{43}$$

Thus, if Gröbner bases are used instead of RadGBasis, the number of differential-elimination steps required for termination *can be arbitrarily large.* This disadvantage of Gröbner bases should be balanced against the fact that algorithms for computing RadGBasis are considerably more complicated than those for computing Gröbner bases (and not even implemented in the real case!). Note that the system (41) is obtained by applying the coordinate transformation $(t, u, v) \to (t, au + bv, cu + dv)$ with $a = 1, b = 2, c = 2, d = 3, k = 3$ to the class (42). Thus (42) has the same differential index as (41), since the differential index is a geometrical invariant.

Consider a system of polynomially nonlinear DAE with m dependent variables and differential index ν. The dimension of the manifold of constraints is at most $m + 1$. Each differential-elimination step of the geometric algorithm lowers this dimension by at least one. Since the dimension of the constraint manifold after termination is at least one, it follows that $m + 1 - \nu \geq 1$ and consequently, that $\nu \leq m$.

Example 17 *Consider the system $u_t = 0, v_t = u(u-1), u(v-1) = 0$. After one step of the Rabier-Rheinboldt algorithm, we find that the radical is generated by the above system together with $u(u-1) = 0$. The algorithm then terminates at the next step and the system has differential index 1. Notice that the* DAE *has differential index 0 on the component $u = 0$ and index 1 on the component $u = 1, v = 1$.*

Example 18 *Application of the Rabier-Rheinboldt algorithm to $u_t = 0, v_t = v, u(u - v) = 0$ yields the new constraint $uv = 0$ and after elimination the output system is $u_t = 0, v_t = v, u = 0$. The differential index of the system is 1, but the differential index on the system's only component is 0. Thus there is not necessarily a component that has the differential index of the system.*

To compute exact values for the differential index we need to compute dimensions for the irreducible components of an algebraic variety. Applications ideally require the isolation of equations describing the irreducible components, and how these change at each iteration when new polynomials resulting from prolongation and elimination are adjoined to the system.

The simplest method is to consider cases based on case-splitting on the Jacobians for the rank conditions. This has the disadvantage of case explosion, and poor representation of the geometry of the variety. A more sophisticated method, that has often proved efficient in algebraic applications, is the method of Characteristic sets. This method gives a good approximation of the primary decomposition of the variety [43, 58, 57]. This is a straightforward method which gives cases described by equations and inequations. For each case sharp values of the dimensions can be determined.

The method of equidimensional decomposition [15, 11], does not depend on factorization, and gives a decomposition of the radical of an ideal I generated by the given system in the form $\sqrt{I} = I_1^{d_1} \cap I_2^{d_2} \cap ... \cap I_n^{d_n}$ where each $I_j^{d_j}$ is an intersection of k_j irreducible prime ideals of the same dimension d_j. The number n, the dimensions d_j and the equations for $I_j^{d_j}$ can be effectively determined without factoring. The geometry of the equidimensional components of the same dimension is identical, although the number of components k_j can currently not be effectively determined without factoring.

Another method is to compute the primary decomposition of I over the rational number field. This gives a good approximation to the primary decomposition over the complex number field. In particular the dimensions of the irreducible components can be determined.

In many applications the variables are real and an algorithm is required for creating the real radical. For example the ideal generated by $u^2 + v^2 = 0$ is radical over the complex numbers and the variety is the union of two intersecting hyperplanes which is singular along $u = v = 0$. The ideal generated by $u^2 + v^2$ is not radical in the real case. In that case the locus consists only of $u = v = 0$, and is regular at each of its points. The real radical is generated by u, v.

An algorithm for creating the real radical exists in principal [1]. However it is considerably more complicated than the radical over the complexes and still seems some time away from practical implementation.

In the real case it is possible to give a description of the irreducible components, ultimately in terms of roots of univariate polynomials. The roots can not be explicitly given, but a description terms of codings of real roots of univariate polynomials is determined. The codings are based on signs and sign changes of the univariate polynomials and their derivatives [12, 46].

We mention that after the CK algorithm has terminated on a system of PDAE, primary decomposition can be used to analyse the geometry of the components. This remark also applies to the family of *rif* algorithms studied in [47].

6 Compatibility analysis of perturbed systems

We apply perturbation techniques [25] combined with methods for reducing systems of differential equations to involutive forms to DAE and PDAE. Hairer et al. [19, 18] introduced the perturbation index of a DAE. This index is a measure of the sensitivity of solutions of a DAE to perturbations of its equations. Gear [13] showed that the perturbation index is closely related to the differential index. Perturbing a differential equation is equivalent to deforming or perturbing its jet variety. It is natural to investigate how the distance between solutions with close or identical initial data, depends on the size of the perturbation. For physically meaningful solutions it seems reasonable to require that this difference vanishes as the perturbation goes to zero. Following a suggestion by Ascher we also explore linearized perturbation techniques.

6.1 Exact perturbation analysis

We suggest a framework for investigating the compatibility of perturbed systems using involutive form algorithms.

Suppose that $R(x,u,\underset{1}{u},...,\underset{p}{u}) = 0$ is a pth order input system of PDAE with solution $u(x)$ and involutive form $CK(R(x,u,\underset{1}{u},...,\underset{p}{u})) = 0$. Adding perturbation terms P to $R = 0$, yields a perturbed system $\hat{R} = R + P = 0$ which has perturbed solution $\hat{u}(x)$. Let $\delta(x)$ be perturbation variables and $P = P(x,\hat{u},\underset{1}{\hat{u}},...,\underset{p}{\hat{u}},\delta,\underset{1}{\delta},...,\underset{p}{\delta})$ with $P = 0$, when $\delta = 0$. There are many possible forms for the perturbation term P, with perhaps the simplest being $P = \delta(x)$.

Regarding the perturbations as given functions we can reduce the perturbed system to involutive form. In so doing the perturbation functions are required to satisfy a number of compatibility equations and inequations. The CK algorithm can also be applied to the compatibility system. The resulting output cases yield existence and uniqueness theorems for local analytic solutions of the perturbed systems. These theorems and perturbed solutions can then be used to examine stability of the system.

We illustrate such exact perturbation analyses by several examples.

Example 19 *Consider the system $u_x = 0, u_y = 0$ and for a constant perturbation δ, an associated perturbed system $\hat{u}_x + \delta sin(y) = 0$, $\hat{u}_y = 0$. By virtue of its integrability condition, the associated system only has solutions if $\delta = 0$,* regardless of how tiny the perturbation variable is.
A perturbed system of form $\hat{u}_x = \delta^1(x,y)$, $\hat{u}_y = \delta^2(x,y)$ has solutions only if the perturbation variables satisfy the perturbation compatibility system $\delta^1_y - \delta^2_x = 0$.
A perturbed system of form $\hat{u}_x = \hat{u}\delta^1(x,y)$, $\hat{u}_y = \delta^2(x,y)$, yields two cases. In one case, in addition to the above system, we have $\delta^2_x = \delta^1\delta^2$ and $\delta^1_y = 0$. In the other case $\delta^1_y \neq 0$, $u = (\delta^2_x - \delta^1\delta^2)/\delta^1_y$ together with two second order equations for δ^1, δ^2.

For the following perturbation analyses, the CK or Rabier-Rheinboldt algorithm is modified by ranking all derivatives (and dependent variables) of the perturbations lower than all derivatives of the given system. The Cartan-Kähler existence uniqueness theorem can be applied to the joint output systems, by observing that it first applies to the compatibility system which only involves the perturbations. Then, the existence and uniqueness theorem applies to the remaining system if we regard (in theory) that the local analytic functional forms of the perturbations, resulting from the first application of the existence and uniqueness theorem, have been substituted into that system.

Example 20 **Navier-Stokes equations** *Adding a perturbation term $(\delta^1, \delta^2, \delta^3, \epsilon)$ to the Navier-Stokes equations yields*

$$\hat{\mathbf{u}}_t + (\hat{\mathbf{u}} \cdot \nabla)\hat{\mathbf{u}} - \mu(\nabla \cdot \nabla)\hat{\mathbf{u}} + \nabla\hat{p} - \mathbf{f} = \boldsymbol{\delta}, \nabla \cdot \hat{\mathbf{u}} = \epsilon \tag{44}$$

where the perturbation variables $\boldsymbol{\delta} = (\delta^1, \delta^2, \delta^3)$ and ϵ are functions of x, y, z, t. The involutive form $CK(\hat{R}) = 0$ of the perturbed system is the system above *augmented with*

$$D_x(\nabla\cdot\hat{\mathbf{u}}) = \epsilon_x,\ D_y(\nabla\cdot\hat{\mathbf{u}}) = \epsilon_y,\ D_z(\nabla\cdot\hat{\mathbf{u}}) = \epsilon_z,\ D_t(\nabla\cdot\hat{\mathbf{u}}) = \epsilon_t,$$

$$\nabla^2\hat{p} - \nabla\cdot\mathbf{f} + \sum_{i=1}^{3}\sum_{j=i}^{3}\hat{u}^i_{x_j}\hat{u}^j_{x_i} = -\epsilon_t - (\hat{\mathbf{u}}\cdot\nabla)\epsilon + \mu(\nabla\cdot\nabla)\epsilon + \nabla\cdot\boldsymbol{\delta} \quad (45)$$

where $(x_1, x_2, x_3) = (x, y, z)$.

Example 21 *Consider the deformed pendulum equations* $\hat{R} = R - \delta = 0$*:*

$$\begin{aligned} \hat{x}_t &= \hat{u} + \delta^1,\ \hat{y}_t = \hat{v} + \delta^2,\ \hat{u}_t = -\hat{\lambda}\hat{x} + \delta^3,\ \hat{v}_t = -\hat{\lambda}\hat{y} - g + \delta^4, \\ & \hat{x}^2 + \hat{y}^2 - 1 = \delta^5. \end{aligned} \quad (46)$$

$CK(\hat{R})$ *is given by* $\hat{R}$ *above together with*

$$\begin{aligned} \hat{x}\hat{u} &+ \hat{y}\hat{v} = \tfrac{1}{2}\delta^5_t - \hat{x}\delta^1 - \hat{y}\delta^2, \\ \hat{\lambda} &- (\hat{u}^2 + \hat{v}^2) + g\hat{y} = -\tfrac{1}{2}\delta^5_{tt} - \hat{\lambda}\delta^5 + \hat{x}(\delta^1_t + \delta^3), \\ & +\hat{y}(\delta^2_t + \delta^4) + 2\hat{u}\delta^1 + 2\hat{v}\delta^2 + Q_1 \\ \hat{\lambda}_t &+ 3g\hat{v} = -\tfrac{1}{2}\delta^5_{ttt} - \hat{\lambda}_t\delta^5 - 2\hat{\lambda}\delta^5_t \\ & -3g\delta^2 + 3\hat{u}(\delta^1_t + \delta^3) + 3\hat{v}(\delta^2_t + \delta^4) \\ & +\hat{x}(\delta^1_{tt} + \delta^3_t) + \hat{y}(\delta^2_{tt} + \delta^4_t) + Q_2. \end{aligned}$$

Here $Q_1 = (\delta^1)^2 + (\delta^2)^2$ *and* $Q_2 = 3\delta^1(\delta^1_t + \delta^3) + 3\delta^2(\delta^2_t + \delta^4)$.

Example 22 *Consider* $\hat{u}_t = \hat{w}\hat{v}_t - \hat{v}\hat{w}_t + \delta^1, \hat{v} = \delta^2, \hat{w} = \delta^3$ *which is given in [19, 18]. Application of the CK algorithm gives the output system:*

$$\hat{u}_t = \delta^3\delta^2_t - \delta^2\delta^3_t + \delta^1, \hat{v} = \delta^2, \hat{w} = \delta^3.$$

Thus the perturbed system and the unperturbed system have differential index 1.

Example 23 *Consider* $\hat{u}_t = \hat{v}^2 + \delta^1, \hat{u} = \delta^2$*. When* $\delta^1 = \delta^2 = 0$ *the system has differential index 2, and the output system is* $u_t = v_t = u = v = 0$*. Application of the CK algorithm to the perturbed system results in two cases both of differential index 2. If* $\hat{v} \neq 0$ *the output system also contains* $\hat{v}^2 = \delta^2_t - \delta^1, 2\hat{v}\hat{v}_t = \delta^2_{tt} - \delta^1_t$*. If* $\hat{v} = 0$*, and this is included in the initial system, the output contains in addition,* $\hat{v}_t = 0$ *and* $\delta^1 = \delta^2_t$*.*

Consider a system $R(t, u, u_t) = 0$ and the perturbed system $\hat{R} = R(t, \hat{u}, \hat{u}_t) - \delta = 0$. Hairer et al [19, 18] introduced the concept of a perturbation index, which is the minimum ν, such that an estimate of the following form exists:

$$\|\hat{u}(t) - u(t)\| < K\left[|\hat{u}(t^0) - u(t^0)| + \|\underset{-1}{\delta}\| + \ldots + \|\underset{\nu-2}{\delta}\| + \|\underset{\nu-1}{\delta}\|\right].$$

In the above, $| \ |$ is the Euclidean norm, $\| \ \|$ is the maximum norm on some sufficiently small time interval $[t^0, t^0 + T]$. Also $\underset{k}{\delta}$, is the kth derivative of the vector of perturbations δ, and $\underset{-1}{\delta} = \int_{t^0}^{t^0+T} \delta(\tau) d\tau$. Here $R = 0$ and $\hat{R} = 0$ should be consistently initialized with close initial values and have unique solutions. The perturbation δ has been suggested as a model of small rapid fluctuations that result from round-off error. Such oscillations have large derivative and this has been offered as an explanation of some instability observed in high index problems.

The perturbation index is not necessarily defined, even though the differential index may be defined. Gear [13] gives the example, $u^2 = 0$, which has differential index 1, but for which it is obvious that no estimate of the type above can exist. Choosing the perturbation $\delta^1 = 0, \delta^2 = \epsilon \sin(\omega t), \delta^3 = \epsilon \cos(\omega t)$ which is bounded for small ϵ, in Example 22, yields $\hat{u}_t = \epsilon^2 \omega$. Then $\hat{u} = \epsilon^2 \omega t$, $\hat{v} = \epsilon \sin(\omega t)$, $\hat{w} = \epsilon \cos(\omega t)$ with initial values at $t = 0$ given by $\hat{u}^0 = 0, \hat{v}^0 = 0, \hat{w}^0 = \epsilon$. Thus $\hat{u}$ is unbounded as $\omega \to \infty$. This stability bound is typical for index two rather than index one problems. However when a linearized perturbation analysis is applied, as described in the next section, this problem has a linearized perturbation index of 1, and the difference between linearized and perturbed linearized solutions is small. Ascher has observed that this lower linearized perturbation index appears to be in closer agreement with the results of numerical experiments.

6.2 Linearized perturbation analysis

In the previous section we saw that the path taken by the CK algorithm on perturbed systems is not generally the same as that taken on the unperturbed system.

In this section we show, following a suggestion by Ascher, however that the CK algorithm behaves in similar fashion on the perturbed linearized form of a system.

Definition 24 *Let $u(x)$ be a solution of CK$(R(x, u, \underset{1}{u}, ..., \underset{q}{u})) = 0$, and suppose that $R = 0$ satisfies the constant rank conditions. The linearization of $R = 0$ about the solution u is*

$$T_u(R) := R_u \, \bar{u} + R_{\underset{1}{u}} \, \underset{1}{\bar{u}} + ... + R_{\underset{q}{u}} \, \underset{q}{\bar{u}} = 0 \tag{47}$$

on solutions u of CK(R)=0. Here the $\underset{k}{\bar{u}}$ are the linearized variables corresponding to the jet variables $\underset{k}{u}$ and $R_{\underset{k}{u}}$ denotes the partial derivatives of R with respect to the kth order jet variables.

The constant rank conditions are needed to ensure that geometrical derivatives correspond to symbolic derivatives. As usual, to effectively realize these conditions for polynomial systems, we need generating sets for the radicals of the polynomial ideals, arising in the application of the CK algorithm.

Example 25 *We construct the linearized pendulum equations. Taking our linearized variables as $\bar{x}, \bar{y}, \bar{u}, \bar{v}, \bar{\lambda}$, we obtain the linearized pendulum equations as*

$$\begin{aligned} \bar{x}_t &= \bar{u},\ \ \bar{y}_t = \bar{v},\ \ \bar{u}_t = -\lambda\bar{x} - x\bar{\lambda},\ \ \bar{v}_t = -\lambda\bar{y} - x\bar{\lambda}, \\ 0 &= 2x\bar{x} + 2y\bar{y}. \end{aligned}$$

Denoting the constraint by $\Psi = x^2 + y^2 - 1$, the calculation for its linearization is $\Psi_x\bar{x} + \Psi_y\bar{y} = 0 = 2x\bar{x} + 2y\bar{y}$. Equivalently this result can be obtained from the coefficient of the ϵ term after making the substitutions $x \mapsto x + \epsilon\bar{x}$, $y \mapsto y + \epsilon\bar{y}$ in the original equation.

For purposes of illustration we apply one step of the Rabier-Rheinboldt algorithm or equivalently the CK algorithm. Differentiation of the constraint $\Psi = 0$ yields $2(x\bar{x}_t + y\bar{y}_t + x_t\bar{x} + y_t\bar{y}) = 0$ which when simplified subject to the linearized system and CK(R)=0 yields $2(x\bar{u} + y\bar{v} + u\bar{x} + v\bar{y}) = 0$. Notice that if we had first applied one step of the CK algorithm to the original system, we obtain $xu + yv = 0$, which when linearized yields the same result. Indeed we have:

Theorem 26 *Suppose that the first two constant rank conditions in (36) are satisfied initially and after every application of $\mathcal{D}$ and $\mathcal{E}$ during the application of the CK algorithm to $R = 0$. Then T_u commutes with $\mathcal{D}$ and $\mathcal{E}$.*

Proof: Let $\mathrm{T}_u R = \bar{R} = 0$ be the linearization of a qth order system $R = 0$ about a solution u of $CK(R) = 0$.

The constant rank assumptions imply that in local coordinates T_u and $\mathcal{D}$ can be computed using:

$$\mathrm{T}_u = \bar{u}^k_K \partial_{u^k_K},\ \ D_{x^i} = \partial_{x^i} + u^k_{L+1_i}\partial_{u^k_L} + \bar{u}^k_{L+1_i}\partial_{\bar{u}^k_L} + \ldots$$

Here K, L are multi-indices and 1_i is the multi-index, whose ith entry is 1 and whose remaining entries are zero. The preservation of the constant rank conditions implies that we can use the local formulae above to compute $\mathrm{T}_u\mathcal{D}\mathcal{R}$ and $\mathcal{D}\mathrm{T}_u\mathcal{R}$. A straightforward calculation with the above explicit forms of the operators, taking care of the allowable ranges of the indices, verifies the commutation of linearization with prolongation:

$$\mathrm{T}_u\mathcal{D}\mathcal{R} = \mathcal{D}\mathrm{T}_u\mathcal{R}.$$

The constant rank conditions imply that the Implicit Function Theorem can be invoked to write $R = 0$ as $v - F = 0, G = 0$ where v consists of some strictly qth order jet variables, and G is a function of jet variables of order $q - 1$ or less.

Thus $\mathcal{E}\mathcal{R}$ is equivalent to $G = 0$. Applying the constant rank conditions for R in the form $v - F = 0, G = 0$ gives

$$\operatorname{rank}\frac{\partial G(x, u, \underset{1}{u}, ..., \underset{q-1}{u})}{\partial(x, u, \underset{1}{u}, ..., \underset{q-1}{u})} = \operatorname{rank}\frac{\partial G(x, u, \underset{1}{u}, ..., \underset{q-1}{u})}{\partial(u, \underset{1}{u}, ..., \underset{q-1}{u})} = \text{constant},$$

in a neighbourhood of $(q-1)$th prolongation of the solution u. Thus the formula above can be applied to compute $\mathrm{T}_u G$. The preservation of the constant rank conditions implies that we can use the local formulae above to compute $\mathrm{T}_u\mathcal{E}\mathcal{R}$ and $\mathcal{E}\mathrm{T}_u\mathcal{R}$. Explicit calculation then shows:

$$\mathrm{T}_u\mathcal{E}\mathcal{R} = \mathcal{E}\mathrm{T}_u\mathcal{R}.$$

■

It is tempting to think that this result would imply that the CK algorithm follows exactly the same path on the linearized system $\mathrm{CK}(\bar{R}) = 0$ and the original system $\mathrm{CK}(R) = 0$. Applying the CK algorithm to the system $R = 0$ given by $u_x = v^2, u = 0$ of Example 23, yields $v^2 = 0$ and consequently $v = 0$ in constant rank form, after the first differentiation-elimination step. The algorithm terminates after the next step, with one additional equation $v_x = 0$. However linearizing the initial system about the solution $u = 0, v = 0, u_x = 0, v_x = 0$ yields $\bar{u}_x = 2v\bar{v}, \bar{u} = 0$ and thus $\bar{u}_x = 0, \bar{u} = 0$ on solutions of $\mathrm{CK}(R) = 0$. The CK algorithm then terminates on the linearized system with an output system $\bar{u}_x = 0, \bar{u} = 0$, which is *not* the linearization of the involutive form of $\mathrm{CK}(R) = 0$.

The problem here is that a projected system, $v^2 = 0$, arose which is not of constant rank. When the rank conditions are maintained at each step of the CK algorithm we have:

Theorem 27 *Suppose that the rank conditions (36) are satisfied without needing to change the form of equations at each step of the CK algorithm. The same sequence of prolongations that makes the unperturbed system $R = 0$ involutive also makes the linearized system $\bar{R} = 0$ involutive. The original system, the linearized system and the perturbed linearized system all have the same differential index in a neighbourhood of u. Furthermore these operations make the perturbed linearized system $\hat{\bar{R}} = \delta$ involutive provided that δ satisfies a linear homogeneous compatibility system.*

Proof: The Symbol of the linearized system $\bar{R} = 0$ and $R = 0$ are equivalent: so both become involutive at the same time and have identical delta-regularity properties (see Appendix). Application of the previous theorem then shows that the same sequence of prolongations and projections that makes $R = 0$ involutive in a neighbourhood of $u(x)$ also makes its linearization involutive.

Consider the perturbed linearized system $\hat{\bar{R}} = \delta$. Since the system is linear, the rank conditions are automatically satisfied, and symbolic differentiation is the same as geometric differentiation. Given a linear system, $Bw = \mu$, then the same operations that reduce $Bw = 0$ to Gauss reduced form, $Cw = 0$, also reduce $Bw = \mu$ to Gauss reduced form $Cw = f(\mu)$, together with a linear compatibility system $g(\mu) = 0$. Both f, g are linear and are explicitly determined by the reduction. The same reasoning shows that the operations that make the linearized system involutive, also make the perturbed linearized system involutive, provided that the perturbations satisfy a linear homogeneous system of differential equations. ∎

To apply this result algorithmically to polynomially nonlinear systems it is necessary to verify that the constant rank conditions are maintained during application of the CK algorithm to the original unperturbed system. For polynomial systems, this can be effectively implemented by checking that the ideals generated during the process are already radical, as presented. Requiring that the Symbol has constant rank, at each step, is assured by our case splitting analysis of the Symbol equations. Finally the presence of relations amongst the independent variables can be determined by Gröbner Bases using elimination orderings as we have already discussed.

Example 28 *To apply this result to the linearized pendulum equations, one first checks that the above conditions hold when the CK (or Rabier-Rheinboldt) algorithm is applied to the original system. At each step, it can be verified that the system is a generating set for a radical ideal, the independence condition is satisfied and the Symbol has constant rank. Thus the perturbed linearized system completes to involution in the same way as the original system.*

The perturbed linearized system $\hat{\hat{R}} = \delta$ *is given by*

$$\begin{aligned} \hat{\hat{x}}_t &= \hat{\hat{u}} + \delta^1, \;\; \hat{\hat{y}}_t = \hat{\hat{v}} + \delta^2, \;\; \hat{\hat{u}}_t = -\lambda\hat{\hat{x}} - x\hat{\hat{\lambda}} + \delta^3, \;\; \hat{\hat{v}}_t = -\lambda\hat{\hat{y}} - x\hat{\hat{\lambda}} + \delta^4, \\ & 2x\hat{\hat{x}} + 2y\hat{\hat{y}} = \delta^5. \end{aligned}$$

Differentiation of the constraint $\Psi = 0$ *yields* $2(x\hat{\hat{x}}_t + y\hat{\hat{y}}_t + x_t\hat{\hat{x}} + y_t\hat{\hat{y}}) = \delta^5_t$. *From* $CK(R) = 0$ *we have* $x_t = u, y_t = v$ *and from the perturbed system* $\hat{\hat{x}}_t = \hat{\hat{u}} + \delta^1, \hat{\hat{y}}_t = \hat{\hat{v}} + \delta^2$. *Thus after elimination:*

$$D_t\Psi = 2(u\hat{\hat{x}} + v\hat{\hat{y}} + x\hat{\hat{u}} + y\hat{\hat{v}}) = \delta^5_t - 2x\delta^1 - 2y\delta^2.$$

Easy calculations verify the identical completion to involutive form on the perturbed and unperturbed linearized systems.

Suppose that both the differential index ν_{diff} and the perturbation index ν_{pert} are defined for a DAE. Gear [13] shows, under certain assumptions that he does not state explicitly, that $\nu_{\text{diff}} \leq \nu_{\text{pert}} \leq \nu_{\text{diff}} + 1$. Gear assumes that the Implicit Function Theorem can be invoked at each stage of his differential elimination process. This is equivalent to our requirement of the rank conditions being maintained. He also assumes the availability of a local existence and uniqueness theorem (which can be that of Rabier and Rheinboldt [36, 37] or the Cartan-Kähler Theorem [23]). To determine the perturbation index Gear integrates the difference between the completed unperturbed system and the perturbed system, to obtain an estimate for the norm of the difference between the perturbed and unperturbed solutions. This estimate shows that $\nu_{\text{pert}} \leq \nu_{\text{diff}} + 1$. For linear systems Gear [13] refines this to $\nu_{\text{pert}} \leq \nu_{\text{diff}}$. Gear next assumes that certain perturbations, those with highest derivatives in the completed system, can be chosen freely. This enables him to choose a rapid oscillatory form for these perturbations and use the perturbed and unperturbed constraint systems to show that $\nu_{\text{diff}} \leq \nu_{\text{pert}}$. Thus $\nu_{\text{diff}} = \nu_{\text{pert}}$ in our case. Gear does not mention that in general the perturbations are constrained by a compatibility system.

We have not as yet given a full analysis of the compatibility system for the exact perturbation analysis to substantiate to our satisfaction that Gear's claim that the perturbations with highest derivative can be chosen arbitrarily. In the linear case this claim is easily verified, and as a result of specializing Gear's results [13] to the linear case:

Theorem 29 *Suppose that $R = 0$ is a first order system of analytic* DAE, *which satisfies the conditions of the previous theorem. Further suppose that the involutive form of the system shows that the system can be consistently initialized with initial values $u(t^0) = u^0, u_t(t^0) = u_t^0$ and corresponding unique solution $u(t)$, and differential index ν_{diff} for the component of the* DAE *containing $u(t)$. Then*

$$\nu_{\text{pert}} = \nu_{\text{diff}}$$

where ν_{pert} is the perturbation index of the perturbed linearized system.

7 Discussion

We have studied differential-elimination algorithms from an algebraic and a geometric perspective. In the context of jet space, derivatives in a differential system $R = 0$ are replaced by formal independent coordinates. The jet variety $\mathcal{R}$ is the solution set of these algebraic equations.

For systems of first order DAE, differentiation and elimination produces a sequence of subvarieties: $\mathcal{R} = \mathcal{R}^{(0)} \supseteq \mathcal{R}^{(1)} \supseteq \mathcal{R}^{(2)} \supseteq \ldots$. Geometrically it is the variety which is important and not its equations. However geometrical differentiation of the equations does not necessarily correspond to symbolic differentiation of the equations, the culprit being the presence of multiplicities and singularities in the equations.

For polynomially nonlinear DAE, there is an effective algebraic algorithm for removing all such multiplicities through the calculation of radicals of ideals. As a consequence we obtained an algorithmic method for which the sequence above is strictly decreasing, and becomes constant at some integer ν:

$$\mathcal{R} = \mathcal{R}^{(0)} \supset \mathcal{R}^{(1)} \supset \mathcal{R}^{(2)} \supset \ldots \supset \mathcal{R}^{(\nu)} = \mathcal{R}^{(\nu+1)}.$$

This yields a local existence theorem, for solutions passing through regular points $(x^0, u^0, u_x^0) \in \mathcal{R}^{(\nu)}$.

We showed that the CK algorithm is equivalent to the Rabier-Rheinboldt algorithm for analytic DAE. Unlike the CK algorithm, the Rabier-Rheinboldt algorithm does not deliver an existence and uniqueness theorem for PDAE due to its lack of consideration of integrability conditions.

Computation of sharp values for the differential index on components requires the dimensions of the components of the manifold of constraints, generated at each stage of the Rabier-Rheinboldt or CK algorithms. These can be obtained from the radical, or by using the method of characteristic sets. Isolating equations for the components can be effectively achieved in the complex case by calculating the primary decomposition and have been investigated by Gabriel Thomas [54]. The real case requires more difficult methods.

Given a system of DAE $R(t, u, u_t) = 0$ and a perturbed system $\hat{R} = R(t, \hat{u}, \hat{u}_t) - \delta(t) = 0$, the perturbation index introduced by [19, 18] is roughly the smallest ν such that $\|\hat{u}(t) - u(t)\|$ is bounded by the norm of the $(\nu - 1)$th derivative of the perturbation δ. We illustrated this *exact perturbation analysis* for the pendulum and Navier-Stokes equations.

Motivated by Ascher we studied a linearized perturbation index. In that case u and $\hat{u}$ in the estimate are replaced by the solutions of the linearized and perturbed linearized systems respectively. We were able to show that provided the sequence of systems produced when the CK algorithm is applied to $R = 0$ maintains certain constant rank conditions, then the differential index of the perturbed linearized system is the same as the differential index of $R = 0$. In experiments Ascher has observed that the linearized perturbation index is a more faithful measure of instability than the exact perturbation index. A possible explanation is that since numerical methods are based on local linearization, a linearized stability analysis is more relevant for studying their stability.

Another major goal of the article has been to argue for jet spaces as a natural venue for investigating PDAE. Fundamental aspects of the DAE case, for example an existence and uniqueness theorem for local analytic solutions, carry directly over to the PDAE case.

The Navier-Stokes equations illustrate that PDAE will be far more challenging than DAE. For the Navier-Stokes equations it is tempting to think that the second order spatial derivatives of the perturbations in the right hand side of (45) will dominate all others and lead to significant numerical errors. However, numerical analysts working on time-dependent PDE are more concerned with time evolution error. For example, a popular numerical method is the method of lines in which the incompressible Navier-Stokes are replaced by a system of time-dependent DAE. In that method the spatial derivative operators (divergence, gradient etc) are replaced with matrices. One more derivative of (45) is needed to obtain a time derivative of the pressure and in particular and equation for $\nabla^2 \hat{p}_t$. If we regard the ∇^2 as invertible [29] then from a DAE viewpoint the Navier-Stokes is of differential index 2 in time. This view is supported by an energy estimate analysis [29]. DAE-like stabilization and regularization techniques for DAE of differential index two were successfully applied to this PDAE in [29] to control the high index behavior.

Successful concepts of index for PDAE will be those that reflect the behaviour seen in numerical simulations. Although the CK algorithm suggests natural candidates for such an index, a PDAE could possess several such indices, depending on the initial or boundary value problem being investigated. Differential-elimination algorithms offer systematic methods for determining constraints and consistent initial conditions for DAE and PDAE. This offers the numerical analyst and modeler, flexibility in their choice of model. The constraints which are incorporated will generally depend on the problem being treated.

A Involutivity and Formal Integrability

The Symbol can be regarded as a way of testing for the presence of possible integrability conditions, in other words for whether $(\mathcal{E} \circ \mathcal{D})\mathcal{R}$ will yield any restrictions on $\mathcal{R}$. We first describe Pommaret's test in local coordinates for involutivity of the symbol.

Suppose $R = 0$ is an analytic system of order q, with independent variables $x_1, x_2, ..., x_n$ and dependent variables $u^1, u^2, ..., u^m$. Here u^i_J where $J = (j_1, j_2, ..., j_n)$ is the usual multi-index notation for jet variables.

The multi-index J is said to be of class k if the first non-vanishing entry of J is j_k. Order the columns of Symbol $\mathcal{R}$ so that columns with multi-indices of higher class appear to the left of those with multi-indices of lower class.

Row-reduce Symbol $\mathcal{R}$. Define

$$\beta^{(k)} := \#\text{ rows of class } k$$

in the row-reduced matrix Symbol $\mathcal{R}$.

For example, for $R = 0$ given by the system of Example 13, $u_{xx} - yu = 0, u_{yy} = 0$,

$$\text{Symbol } \mathcal{R} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

The first column corresponds to the single class 2 coordinate u_{yy} while the 2nd and 3rd correspond to the class 1 jet variables u_{xy} and u_{xx}. Hence $\beta^{(1)} = 1$, $\beta^{(2)} = 1$.

A delta-regular coordinate system is one in which the sums $\sum_{i=k}^{i=n} \beta^{(i)}$ take their maximum values (almost all coordinate systems are delta-regular). In general

$$\text{rank Symbol } \mathcal{D}\mathcal{R} \geq \sum_{k=1}^{n} k\beta^{(k)}. \tag{48}$$

If rank Symbol $\mathcal{DR} = \sum_{k=1}^{n} k\beta^{(k)}$ then the coordinate system is delta-regular, and Symbol $\mathcal{R}$ is involutive. For ODE or DAE, $n = 1$ and $\beta^{(1)} =$ rank Symbol $\mathcal{DR}$, and thus such systems always have involutive Symbol.

For the system of Example 13 rank Symbol $\mathcal{DR} = 4 > 3 = \sum_{k=1}^{n} k\beta^{(k)}$. However prolonging again we find that $\beta^{(1)} = 3$, $\beta^{(2)} = 1$ and rank Symbol $\mathcal{D}^2\mathcal{R} = 5 = \sum_{k=1}^{n} k\beta^{(k)}$. So the system now has involutive Symbol. Thus integrability conditions can be tested for by making a single prolongation. In particular this yields the integrability condition resulting from the equality of the second partial of $u_{xx} - yu = 0$ with respect to y, to the second partial of $u_{yy} = 0$ with respect to x.

If the test (48) is applied to the wave equation $u_{xy} = 0$ expressed in characteristic coordinates, we find that the inequality is strict, at all orders of prolongation. Thus we are not able to conclude whether the Symbol is involutive or not. We stress that involutivity is a property which is independent of the coordinate system. Changing coordinates to $r = x+y, s = x-y$, transforms the wave equation to $u_{rr} - u_{ss} = 0$. Then (48) becomes an equality and the wave equation is involutive (in this and the original coordinate system). The reason for the failure of the first application of the test is that the x, y-coordinate system is not delta-regular. Indeed in a delta-regular coordinate system, Symbol $\mathcal{R}$ is involutive if (48) is an equality. It is always possible to make a coordinate system delta-regular, by making affine changes of the form $x \mapsto Ax$, where A is a constant invertible matrix, and this coupled with (48) yields a coordinate independent test for involutivity.

In a delta-regular coordinate system the Cartan characters are calculated directly from the involutive form of a system. These encode information about the degree of generality of local analytic solutions of an involutive PDE (especially see [50]).

The algorithm below uses natural integrability conditions resulting from the equality of mixed partial derivatives. These conditions, $I = \text{IC}(\mathcal{R})$, are used as a test for how far to prolong $\mathcal{R}$ before testing again for $\mathcal{R} = (\mathcal{E} \circ \mathcal{D})\mathcal{R}$. It does not need to test for delta-regularity. However it has the disadvantage that it doubles the order of the system from $|\mathcal{R}|$ to $2|\mathcal{R}|$, when the integrability conditions do not reduce to zero modulo simplification with respect to $R = 0$ (i.e. $I \overset{\mathcal{R}}{\not\to} 0$).

Algorithm 4

Input $\mathcal{R}$ linear in its $|\mathcal{R}|$-th order derivatives
Repeat
  While $\mathcal{R} \neq (\mathcal{E} \circ \mathcal{D})\mathcal{R}$ **repeat** $\mathcal{R} := (\mathcal{E} \circ \mathcal{D})\mathcal{R}$
  Calculate $I = \text{IC}\,(\mathcal{R})$
  If $I \overset{\mathcal{R}}{\not\to} 0$ **then** $\mathcal{R} := \mathcal{D}^{|\mathcal{R}|}\mathcal{R}$
until $I \overset{\mathcal{R}}{\to} 0$ **and** $\mathcal{R} = (\mathcal{E} \circ \mathcal{D})\mathcal{R}$
Output $\mathcal{R}$

Consider the case where the $R = 0$ is a system of differential polynomials with rational coefficients, which is linear in its highest derivatives (if not prolonging once will give an equivalent system with this property). The above algorithm is a variation of Rust's *rif'* algorithm [47]. Specifically a ranking of derivatives is chosen which ranks derivatives of higher order lexicographically greater than those of lower order, and reduction and integrability conditions are as defined in [47]. At any stage of the application of the algorithm, the current form of the system is partitioned into those of order $|\mathcal{R}|$ (Rust's set $\mathcal{L}$) and the remaining equations of lower order (Rust's set $\mathcal{N}$). As discussed earlier, we assume that a generating set for the radical of the ideal generated by $\mathcal{R}$ is chosen. Rust's work [47] rigorously justifies termination, and gives an existence and uniqueness theorem for formal power series solutions which is independent of the Cartan-Kuranishi theory. The output is a formally integrable (but not necessarily involutive) system.

Consider the system, $R = 0$ given $u_{xx} - yu = 0, u_{yy} = 0$, of Example 13. Initially $|\mathcal{R}| = 2$, and $\mathcal{R} = (\mathcal{E} \circ \mathcal{D})\mathcal{R}$, so we calculate $\mathrm{IC}(R)$, as $I = D_{yy}(-yu) - D_{xx}(0) = -yu_{yy} - 2yu_y$. Since $u_{yy} = 0$, $I \xrightarrow{\mathcal{R}} -2yu_y \neq 0$, so $R = 0$ is prolonged to order $2|\mathcal{R}| = 4$. Iterating the loop $\mathcal{R} := (\mathcal{E} \circ \mathcal{D})\mathcal{R}$, the algorithm terminates with output system $u = u_x = u_y = u_{xx} = ... = u_{xyyy} = u_{yyyy} = 0$.

Partly as a consequence of our insistence that these algorithms consist of coordinate independent steps, they often produce large output systems of equations, including apparently redundant equations.

The following algorithm transforms a formally integrable system to an equivalent formally integrable system of lower order.

Algorithm 5: Make Minimal Formally Integrable System

Input A formally integrable system $\mathcal{R}$
while $\mathcal{R} = (\mathcal{D} \circ \mathcal{E})\mathcal{R}$ **repeat** $\mathcal{R} := \mathcal{E}\mathcal{R}$
Output $\mathcal{R}$

Application of Algorithm 5 to the output of the CK algorithm and Algorithm 4 for the system of Example 13, yields $u = 0$ in both cases.

For comparison we apply a coordinate dependent approach to Example 13. For this linear system, the coordinate dependent approach of Mansfield [30] (or alternatively [39]) yields the following calculations. No prolongations need to be made and the coordinate dependent approach directly seeks the integrability condition $I = D_{yy}(-yu) - D_{xx}(0) = -yu_{yy} - 2yu_y$ which reduces to $u_y = 0$. Now $u_{yy} = 0$ is a consequence of $u_y = 0$ so the new system is $u_{xx} - yu = 0, u_y = 0$. The integrability condition in this case is $D_y(-yu) - D_{xx}(0)$ which reduces to $u = 0$. Since $u_{xx} - yu = 0, u_y = 0$ are a consequence of $u = 0$, the coordinate dependent algorithm terminates with output $u = 0$.

The simplicity of the above calculation illustrates the importance of making geometric approaches more efficient by incorporating aspects of the coordinate dependent approaches. See Seiler [50] for progress in this direction for the

CK algorithm and also [41, 42] for a coordinate dependent approach which incorporates aspects of geometric approaches.

Suppose that the sequence of systems produced during the application of an algorithm are $R^{(0)} = 0, R^{(1)} = 0, R^{(2)} = 0, \ldots$. If we apply a change of coordinates φ to $R^{(0)} = 0$ and then apply the same algorithm, then another sequence of systems arises: $\varphi(R^{(0)}) = \tilde{R}^{(0)} = 0, \tilde{R}^{(1)} = 0, \ldots$. The algorithm is absolutely geometric if: $\varphi^{-1}(\tilde{\mathcal{R}}^{(i)}) = \mathcal{R}^{(i)}$ for all i. Algorithm 4 and the CK algorithm (provided that the system is adjusted to be delta-regular at each step) are both of this type, and will output systems of the same order. If non-geometric criteria are used to determine the length of prolongation, then generally the order of the output systems will be coordinate dependent. This may be desirable for efficiency reasons, for example, if a delta-regularity test is not used at every iteration, or a redundancy criterion for integrability conditions is used in Algorithm 4, to try to reduce the number of prolongations.

Another alternative is to apply the algorithm which produces a minimal formally integrable system to the output of a coordinate dependent approach. This minimal formally integrable system (and its order), will be coordinate independent. One may then apply Mansfield's Prolongation Theorem [30, 31] to such a minimal formally integrable system to predict how far it should be prolonged to become involutive.

Captions for Figures

Figure 1
The bold planar curve shows the graph of a solution of $D_x u(x) - (3x^2 - 2x + 1)/\cosh(u(x)) = 0$. Its prolongation in J^1 is the bold non-planar curve. The other planar curves represent the family of graphs of solutions of $D_x u(x) - (3x^2 - 2x + 1)/\cosh(u(x)) = 0$ and the nonplanar curves their prolongations. The shaded surface represents the ODE's solution variety in J^1.

Figure 2
The jet varieties of the system of PDE $u_x - 1 = 0$, $(u_y)^2 - u = 0$, $u_x = 0$ in J^1 are respectively: the plane surface on the left, the curved surface and the dark shaded plane surface on the right. The variety of the first two equations is the bold curve closest to the viewer. The three varieties have no common intersection point. Thus this system has no solutions.

Figure 3
Solutions of $xu_x^2 - 3uu_x + 9x^2 = 0$. Thin curves are representatives from the family constituting the general solution. The thick curve is the singular (envelop) solution $u^2 - 4x^3 = 0$.

Figure 4
Prolongations of graphs of the solutions of $xu_x^2 - 3uu_x + 9x^2 = 0$ in J^1. Thin curves are solutions from the family of general solutions and the thick curve is

the prolongation of the singular solution $u^2 - 4x^3 = 0$.

References

[1] Becker, E. and R. Neuhaus. 1993. In: Computation of real radicals of polynomial ideals. Computational Algebraic Geometry, Progress in Math., **109** (Birkhäuser). 1–20.

[2] Becker, T. and V. Weispfenning. 1993. Gröbner bases: a computational approach to commutative algebra. (Springer Verlag, New York)

[3] Bryant, R.L., S.S. Chern, R.B. Gardner, H.L. Goldschmidt and P.A. Griffiths. 1991. Exterior Differential Systems, Mathematical Sciences Research Institute Publications **18** (Springer Verlag, New York).

[4] Boulier, F., D. Lazard, F. Ollivier and M. Petitot. 1995. Representation for the radical of a finitely generated differential ideal. Proc. ISSAC 1995. ACM Press. 158–166.

[5] Brenan, K., S. Campbell and L. Petzold. 1989. Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations. (North-Holland).

[6] Buchberger, B. 1965. An Algorithm for Finding a Basis for the Residue Class Ring of a Zero-Dimensional Polynomial Ideal (German), PhD. Thesis, Univ. of Innsbruck, Math. Inst.

[7] Carrà-Ferro, G. 1987. *Gröbner Bases and Differential Algebra,* Lecture Notes in Comp. Sci. **356** 128–140.

[8] Cartan E. 1946. *Les Systèmes Différentiels Extérieurs et leurs Applications Géométriques* (Hermann, Paris).

[9] Dirac, P.A.M. 1958. *Generalized Hamiltonian Mechanics*, Proc. Roy. Soc. A. 246: 326-332.

[10] Ehresmann, C. 1950. *Les connexions infinitésimales dans un espace fibré différentiable*, Colloque de topologie (espaces fibrés) 29-55, Bruxelles.

[11] Eisenbud, D., Huneke, C. and Vasconcelos, W. 1992. *Direct methods for primary decomposition*, Invent. Math. **110** 210–235.

[12] Galligo, A. and N Vorobjov. 1995. *Complexity of finding irreducible components of a semi-algebraic set.* J. Complexity **11** 174–193.

[13] Gear, C.W. 1990. *Differential algebraic equations, indices, and integral algebraic equations*, SIAM J. Numer. Anal. **27** 1527-1534.

[14] Gear, C.W. and L. Petzold. 1984. *ODE methods for the solution of differential/algebraic systems*, SIAM J. Numer. Anal. **21** 716-728.

[15] Gianni, P., B. Trager and G. Zacharias. 1988. *Gröbner Bases and Primary Decomposition of Polynomial Ideals*, J. Symbolic Computation. **6** 149–166.

[16] Gianni, P., B. Trager and G. Zacharias. 1989. *Gröbner Bases and Primary Decomposition of Polynomial Ideals*, In: Computational Aspects of Commutative Algebra. Edited by L. Robbiano. 15–33. (Academic Press, New York).

[17] Goldschmidt, H. 1967. *Integrability Criteria for Systems of Partial Differential Equations*, J. Diff. Geom. **1** 269–307.

[18] E. Hairer and G. Wanner. 1991. *Solving Ordinary Differential Equations II.* (Springer-Verlag).

[19] Hairer, E., Ch. Lubich and M. Roche. 1989. *The Numerical Solution of Differential-Algebraic Systems by Runge-Kutta Methods*, **1409**. (Springer-Verlag, New York).

[20] Hereman, W. 1994. *Review of symbolic software for the computation of Lie symmetries of differential equations.* Euromath Bull., **1** 45–79.

[21] Hubert, E. 1996. *The General Solution of an Ordinary Differential Equation*, Proc. ISSAC 1996. ACM Press. 189–195.

[22] Janet, M. 1920. *Sur les systèmes d'équations aux dérivées partielles*, J. de Math **3** 65–151.

[23] Kähler, E. 1934. Einführung in die Theorie der Systeme von Differentialgleichungen. (B. G. Teubner, Leipzig).

[24] Kovalevskaya, S. 1875. *Zur Theorie der Partiellen Differentialgleichungen*, J. Reine Agnew. Math. **80** 1–32.

[25] Kevorkian, K. and J.Cole. 1981. *Perturbation methods in applied mathematics*, Springer-Verlag, New York, 1981.

[26] Kredel, H. and V. Weispfenning. 1988. *Computing dimension and independent sets for polynomial ideals*, J. Symb. Comp. **6** 231-247.

[27] Kuranishi, M. 1957. *On E. Cartan's prolongation theorem of exterior differential systems*, Amer. J. Math., **79** 1–47.

[28] Lewy, H. 1957. *An example of a smooth linear partial differential equation without solution*, Ann. Math. **66** 155–158.

[29] Lin, P. *A sequential regularization method for time-dependent incompressible Navier-Stokes equations*, SIAM J. Numer. Anal. (to appear).

[30] Mansfield, E. 1991. *Differential Gröbner Bases*, Ph.D Thesis, University of Sydney.

[31] Mansfield, E. 1996. *A simple criterion for involutivity*, J. London Math. Soc. **2** 54, 323–345.

[32] Mattsson, S.E. and G. Söderlind. 1993. *Index reduction in differential-algebraic equations using dummy derivatives*, SIAM J. Sci. Comp. **14** 677-692.

[33] Olver, P.J. 1986. Application of Lie groups to differential equations. (Springer Verlag, New York).

[34] Pommaret, J. F. 1978. Systems of Partial Differential Equations and Lie Pseudogroups. (Gordon and Breach science publishers, Inc.).

[35] Quéré, M.P. and G. Villard. *An algorithm for the reduction of linear DAE*, In: Proc. ISSAC 1995. ACM Press. 223–231.

[36] Rabier, P.J. and W.C. Rheinboldt. 1991. *A general existence and uniqueness theory for implicit equations*, Differential and Integral Equations, **4** 563-582.

[37] Rabier, P.J. and W.C. Rheinboldt. 1994. *A geometric treatment of implicit differential-algebraic equations*, Journal of Differential Equations, **109** 110-146.

[38] Reich, S. 1989. *Beitrag zur Theorie der Algebrodifferentialgleichungen*, Ph. D. dissertation in Engineering, T.U. Dresden.

[39] Reid, G.J. 1991. *Algorithms for reducing a system of PDEs to standard form, determining the dimension of its solution space and calculating its Taylor series solution*, Euro. J. Appl. Math. **2** 293–318.

[40] Reid, G.J. and A. D. Wittkopf. 1993. The long guide to the Standard Form package. Programs and documentation available by anonymous ftp at `www.cecm.sfu.ca/~reid`

[41] Reid, G.J., A. D. Wittkopf and A. Boulton. 1994. *Reduction of systems of nonlinear partial differential equations to simplified involutive forms.* Institute of Applied Mathematics Report 1994-14, Univ. of British Columbia.

[42] Reid, G.J., A. D. Wittkopf and A. Boulton. 1996. *Reduction of systems of nonlinear partial differential equations to simplified involutive forms*, Eur. J. of Appl. Math. **7** 604 - 635.

[43] Ritt, J.F. 1950. *Differential Algebra.* Colloquium Publications. Volume XXXIII, Amer. Math. Soc.

[44] Rheinboldt, W.C. 1984. *Differential-algebraic systems as differential equations on manifolds*, Math. Comp., **43** 473-482.

[45] Riquier, C. 1910. Les Systèmes d'Équations aux Dérivées Partielles. (Gauthier-Villars, Paris).

[46] Roy, M.-F. and N Vorobjov. 1996. *Computing the complexification of a semialgebraic sets.* Proc. ISSAC 1996. ACM Press. 26–34.

[47] Rust, C. Ph.D Thesis. In preparation (University of Chicago).

[48] Saunders, D.J. 1989. *The Geometry of Jet Bundles*, London Mathematical Society Lecture Notes Series **142**, Cambridge University Press, Cambridge.

[49] Schü, J., W.M. Seiler, and J. Calmet. 1992. *Algorithmic Methods for Lie Pseudogroups*, In Proc. Modern Group Analysis. Acireale.

[50] Seiler, W.M. 1994. *Analysis and application of the formal theory of partial differential equations*, Ph. D. thesis, Lancaster University.

[51] Seiler, W.M. 1995. *Involution and Constrained Dynamics II: The Faddeev-Jackiw Approach*, J. Phys. A. **28** 7315-7331.

[52] Seiler, W.M. and R.W. Tucker. 1995. *Involution and Constrained Dynamics I: The Dirac Approach*, J. Phys. A. **28** 4431–4451.

[53] Spencer, D. 1969. *Overdetermined Systems of Linear Differential Equations*, Bull. A.M.S. **75** 179–239.

[54] Thomas, G. 1996. *Symbolic Computation of the Index of Quasilinear Differential-Algebraic Equations*, In: Proc. ISSAC 1996. ACM Press. 196–203.

[55] Thomas, G. 1997. *Contributions Théoretiques et Algorithmiques à l'Étude des Equations Différentielles-Algébraiques. Approche par le calcul formel. PhD Thesis*, Institut National Polytechnique de Grenoble.

[56] Tresse, A. 1894. *Sur les invariants différentiels des groupes continus de transformations*, Acta Mathematica **18** 1–88.

[57] Wang, D. 1989. *Characteristic sets and zero structures of polynomial sets.* Preprint, RISC-LINZ.

[58] Wu, W.T. 1984. *Basic Principles of mechanical theorem proving in elementary geometries*, J. of System Science and Mathematical Science. **4**(3). 207–235.