Helsinki University of Technology Institute of Mathematics Research Reports

Teknillisen korkeakoulun matematiikan laitoksen tutkimusraporttisarja

Espoo 2002 A448

Numerical Solution and Structural Analysis of Differential-Algebraic Equations

Teijo Arponen

TEKNILLINEN KORKEAKOULU
TEKNISKA HÖGSKOLAN
HELSINKI UNIVERSITY OF TECHNOLOGY
TECHNISCHE UNIVERSITÄT HELSINKI
UNIVERSITE DE TECHNOLOGIE D'HELSINKI

Helsinki University of Technology Institute of Mathematics Research Reports
Teknillisen korkeakoulun matematiikan laitoksen tutkimusraporttisarja
Espoo 2002 A448

Numerical Solution and Structural Analysis of Differential-Algebraic Equations

Teijo Arponen

Dissertation for the degree of Doctor of Science in Technology to be presented with due permission of the Department of Engineering Physics and Mathematics, for public examination and debate in Lecture Hall G at Helsinki University of Technology (Espoo, Finland) on the 18th of June, 2002, at 12 noon.

Helsinki University of Technology
Department of Engineering Physics and Mathematics
Institute of Mathematics

Teijo Arponen: *Numerical Solution and Structural Analysis of Differential-Algebraic Equations*; Helsinki University of Technology Institute of Mathematics Research Reports A448 (2002).

Abstract: *In the last two decades differential-algebraic equations (DAEs) have become an important branch in numerical analysis. In this Thesis we study them from a new, geometric point of view. The DAE is interpreted as a subset of a jet bundle and its solution are induced by the Cartan distribution on the jet bundle. We also introduce a method to examine and define the structure of a general, polynomial, DAE whose locus is not necessarily a fibred manifold. Also it is shown how some singularities of multibody systems are removed by using the algebraic techniques used in this approach.*

AMS subject classifications: 34A26,65L05,70B15,70E99,34A09,58A20

Keywords: symbolic computation, Runge-Kutta methods, index reduction, overdetermined differential equations

Teijo.Arponen@hut.fi

ISBN 951-22-5953-2
ISSN 0784-3143
Espoo, 2002

Helsinki University of Technology
Department of Engineering Physics and Mathematics
Institute of Mathematics
P.O. Box 1100, 02015 HUT, Finland
email:math@hut.fi http://www.math.hut.fi/

Acknowledgements

I gratefully acknowledge the funding during years 1998-2001 from the following resources: Vilho, Yrjö ja Kalle Väisälän rahasto (Finnish Academy of Sciences and Letters), Alfred Kordelinin säätiö and Teknillisen Korkeakoulun tutkijankoulutusstipendi (Helsinki University of Technology).

I am deeply grateful to my advisor professor Jukka Tuomela, Joensuun Yliopisto, for discussions, motivation, encouragement and patience. I am also very thankful to my supervisor professor Olavi Nevanlinna, and to doctors Liz Mansfield and Werner Seiler for reviewing the manuscript. And, all the staff in the Institute of Mathematics at Helsinki University of Technology know I have enjoyed the working environment there!

However, my deepest gratitude goes to my wife, Migi, for her love, support and patience during my visits abroad, and to my sons Joni, Niko, Oskari, for just being there.

Publications

This Thesis consists of the present introduction and the following papers:

[A] J. Tuomela and T. Arponen. On the numerical solution of involutive ordinary differential systems. *IMA J. Numer. Anal.*, 20:561–599, 2000.

[B] J. Tuomela and T. Arponen. On the numerical solution of involutive ordinary differential systems: higher order methods. *BIT*, 41:599–628, 2001.

[C] T. Arponen. Regularization of constraint singularities in multibody systems. *Multibody Systems Dynamics*, 6:355–375, 2001.

[D] T. Arponen. The complete form of a differential algebraic equation. Revised version of Technical Report A438, Helsinki University of Technology, 2001. Submitted.

Author's contribution: in [A] the author designed and programmed the implementation of the algorithms with Mathematica. He also ran numerical tests and wrote corresponding parts in sections 6 and 7 of [A]. In [B] the author planned and wrote the preprint [TA99] which consists half of [B], and designed and programmed the implementation of its algorithms with Maple. Papers [C] and [D] the author produced individually.

Contents

1 Introduction

Many phenomenons in nature are modelled by *differential equations*. For example, if one wants to know the motion of a satellite around the Earth, one constructs, according to the dynamical laws of Isaac Newton, equations modelling the motion of satellite. Then, the equations have to be solved. Sometimes the solution can be found exactly, but usually one needs to use approximative methods, i.e. to construct a *numerical solution*. A numerical solution is, in general, not completely accurate but an approximation to the actual solution.

The subject of this Thesis is differential-algebraic equations, for which we will from now on use the shorthand notation DAE. The name DAE refers to a system with differential and algebraic equations. Here 'algebraic' refers to any non-differential equations.

1.1 On the history of DAEs

The DAEs arose originally in engineering problems, mainly in electrical circuits and multibody systems. In the circuit case, the laws of Kirchhoff (the sum of the voltage drops around any closed loop is zero; at any point of a circuit, the sum of inflowing currents is equal to the sum of outflowing currents) produce, with the modeling of the connections between current and voltage in the circuit elements, equations for the currents.

In multibody systems, the differential equations come from the dynamical laws governing the motion of bodies and the constraint equations come from the rigidity of the system. For example, in the pendulum (see section 1.3) case the constraint equation is the constant length of the bar.

The name "multibody systems" includes for example attitude control of satellites and space vehicles, movement of robots and ground vehicles, e.g. a railway wheelset. See for example [RS88, HD90].

How to solve differential equations numerically? This has been studied since Euler at 18^{th} century, but since the advent of computers, the interest to this question has grown enormously, enough to become a branch of mathematics on its own.

In 1960's engineers working on electrical circuits or multibody systems realized that solving a differential equation with constraints is more involved than solving one without constraints; that is, the constrained case can not in general be reduced to the unconstrained case by some standard trick. The first paper which introduced a way to attack these problems was written by C. W. Gear [Gea71] in 1971. There also the name "differential-algebraic equation" was introduced. A rapid development of numerical methods for DAEs begun with [Pet82] in the beginning of 1980's. Petzold's code *DASSL* is nowadays still widely used e.g. in problems of electrical or chemical engineering.

There are also several concepts of a so called *index*, which is an integer measuring how difficult a DAE is to solve numerically. In conventional DAE research the concepts of different indices sometimes even dominate the discussion. For example, the *DASSL* code typically works reliably only when the (differential) index is at most one. There are many papers suggesting suitable numerical schemes for systems with index two or three, but there are always some additional requirements for the system to fulfill.

Conventionally, when the (differential) index is more than one, the DAE is called a 'higher index' problem and considered as something that should be avoided. There are also *index reduction* techniques, we will discuss these in section 2.2. Recent surveys of different concepts of indices and relations between them are e.g. [CG95a, Man96, Sei99].

1.2 Geometrical approaches to DAEs

There has evolved also several geometrical approaches to DAEs, beginning in 1984 by Rheinboldt [Rhe84] and continuing by Reich [Rei90, Rei91] and [RR91, RR94a]. Another version is done by Szatkowski [Sza92]. We will return to these shortly. At this point we remark that these geometrical approaches are more intrinsic but less constructive than those introduced in the previous section.

In 1950's Ehresmann introduced the concept of a *jet space* in differential geometry. These were later used as a basic building block in constructing the *formal theory of* partial *differential equations* (which we call just *formal theory* from now on), see the introduction in [A] for references and more information. The jet spaces were first applied to analyze DAEs in 1993 in [PT93] and [LV94], independently.

The equation is considered as a locus in a suitable jet space. Since DAEs are a special case of (nonlinear) PDEs one might think that the theory developed in [A] is just a special case of the formal theory, however, this is not so: in formal theory everything is based on fibered submanifolds of a jet space, but in our approach the manifold is not supposed to be fibered. Also, the solution is defined as an integral manifold of a distribution which is induced by natural conditions.

The reason for the first difference, namely dropping the requirement of a fibration of the locus, stems (perhaps surprisingly) from numerical point of view: the method for computing a solution (that is, an integral manifold) is based on the idea that we are allowed to move away from the locus and then project orthogonally back to it. This procedure does not respect any fibration of the locus.

The reason for the second difference in our approach is that it is natural and also more general: a distribution is a more general object than a vector field. First of all this makes regular some situations which classically are considered as singularities. Secondly, an interesting property of distributions is that while a singularity of a vector field is generically zero dimensional, that of a distribution might have generically positive dimension, see also [Tuo97].

The approach of Rheinboldt et al. is based on a reduction process, which is shown in [Kor97, RLW01], independently, to be equivalent with a special case of the formal theory.

There is also an interesting application: impasse points. These are studied for example by Rabier and Rheinboldt in [RR94b, RR94c]. Their results are generalized and proofs quite dramatically simplified in [Tuo97] by using distributions. However, in this Thesis we will not consider impasse points, an interested reader may consult the abovementioned papers and references therein.

Remark 1.1. There is an interesting concept, *extended jet space*, introduced by P. Olver in [Olv86]. Now we can interprete our solution, see [A, def. 3.7], as a section in the extended jet space. However, we do not gain much from this, since the tools of formal theory also need the fibration of the manifold.

Remark 1.2. Also [Sza92] considers distributions as we do, but his point of view is quite abstract; he considers ($\infty-$dimensional) Banach manifolds and defines for them similar reduction process as in [Rhe84]. He also notes that the reduction process might lead to an infinite loop.

Remark 1.3. There is also an interesting approach [Pry01] based on Taylor series. The paper is related to our approach in the sense that [Pry01] also does not transform it to a first order equation, but studies the highest derivatives to find structure of the system. Here we note that it is not clear how to *define* what is meant by a structure of a DAE. Our paper [D] offers one view point to this.

1.3 The canonical example: pendulum

Undoubtedly the most famous example of a DAE is the frictionless pendulum, see figure 1: a massless rod and an infinitesimal ball with unit mass at the end of the rod. Using cartesian coordinates x, y for the center of mass, its motion can be modelled by the following equations:

$$\begin{cases} x'' + \lambda x &= 0 \\ y'' + \lambda y + g &= 0 \\ x^2 + y^2 - L^2 &= 0 \end{cases} \tag{1}$$

where g is the gravitational constant, L is the length of the pendulum and (x, y) is the location in cartesian coordinates. To use jet notation, put $y^1 := x$, $y^2 := y$, $y^3 := \lambda$, hence $x'' \mapsto y^1_2$ and $y'' \mapsto y^2_2$. For convenience, also put the constants $g = 1$ and $L = 1$. In this notation:

$$f := \begin{cases} y^1_2 + y^1 y^3 &= 0 \\ y^2_2 + y^2 y^3 + 1 &= 0 \\ (y^1)^2 + (y^2)^2 - 1 &= 0 \end{cases} \tag{2}$$

Here the constraint equation is the constant length of the bar (the dotted arc in figure 1):

$$(y^1)^2 + (y^2)^2 - 1 = 0 \tag{3}$$

For later reference, we formulate a system which is (2) except that (3) is replaced by its derivative (and the common factor 2 is canceled):

$$g := \begin{cases} y_2^1 + y^1\, y^3 & = 0 \\ y_2^2 + y^2\, y^3 + 1 & = 0 \\ y^1\, y_1^1 + y^2\, y_1^2 & = 0 \end{cases} \tag{4}$$

Now any solution of g should also be a solution of f, provided we start with an initial point which satisfies also (3). Now, computing with g might give something like in figure 2, where the movement of the ball is described by the dashed arcs. There one can see that (3) is not fulfilled, i.e. the ball does not stay on the dotted arc. A corollary of this is that this (numerical) solution is not a solution of (2)! This is an example of phenomena called *drift-off*: the numerical solution is "drifting" off the constraint equation (3).

We will come back to this example in several occasions.

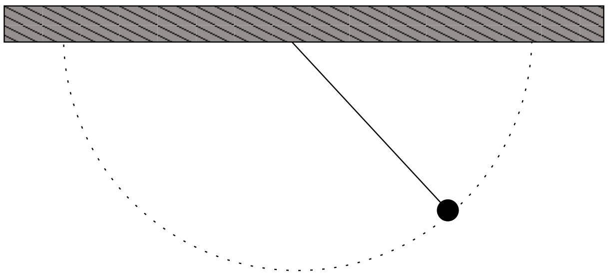

Figure 1: The pendulum. The dotted arc shows the area where the ball will move.

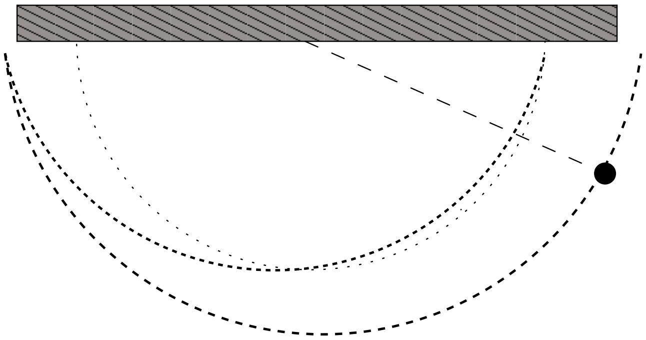

Figure 2: A drift-off with the pendulum. The computed solution (dashed arcs) does not stay in the dotted arc.

1.4 Basic tools

In this section we give a very brief description of what mathematical tools we have used. For rigorous definitions, see [A, sections 2.1 and 3.2], [B, sections 2.1, 6 and 8], [C, sections 3.1-3.3], [D, section 3].

>From the numerics of ODEs we take the Runge-Kutta methods and modify it with geometrical ideas. The theorem of existence and uniqueness of a solution is the one dimensional version of the Frobenius theorem, well known in differential geometry. From commutative algebra we use the concept of ideals, especially their prime decomposition. This decomposition has a correspondence in varieties, yet in this case the varieties have additional structure.

The basic ideas (or, lines of thought) come from the formal theory of PDEs. The main concepts (e.g. involutivity, prolongation $J_q \to J_{q+1}$ and surjectivity of the projection $J_{q+1} \to J_q$) are geometrical, hence intrinsic. However, when we need to compute something, we need algebra. Now, it is not always clear what algebraic concepts (if any) correspond to the geometrical concepts. For example in case of involutivity, see section 3.4.

2 DAEs

As usual, we shall use word 'equation' both for a system consisting of a single equation, and for a system of several equations. It will be clear from the context which one is meant.

Let us start by putting a canonical system right here:

$$\begin{cases} f^1(t, y, y_1, y_2, \ldots, y_q) & = 0 \\ f^2(t, y, y_1, y_2, \ldots, y_q) & = 0 \\ & \vdots \\ f^k(t, y, y_1, y_2, \ldots, y_q) & = 0 \end{cases} \tag{5}$$

where $k \geq n$, $y = \left(y^1, \ldots, y^n\right)$ and each f^i is a smooth enough real valued function.

2.1 On the definitions of indices

We will not even try to do a survey on several definitions of the indices, for that the reader is referred to [CG95a] and references therein. The purpose of this section is to point out that there are two main approaches when defining a differential index: a geometrical one [RR91, Rei90] and a 'derivative array' [BCP89, CG95b] which is based on handling the defining equations. Both of these definitions include constant rank conditions, which are slightly different.

The first is intrinsic but more difficult to construct as it requires finding suitable coordinates, on the other hand the latter is less intrinsic but rather straightforward to construct, although *checking* the constant rank conditions might cause problems, see the abovementioned references. In the view point of formal theory, the intrinsic meaning of the first one is: index equals the number of steps of the Cartan-Kuranishi algorithm needed to reach an involutive form, see [RLW01].

Under suitable conditions these definitions coincide. Any attempt to formulate precisely what these 'suitable conditions' would be, becomes easily rather messy and uninformative.

Remark 2.1. Between these approaches is that of Kunkel and Mehrmann [KM98] whose 'strangeness index' is defined by using a mixture of these two approaches. Their approach is more general than that of [CG95b] but requires finding certain diffeomorphisms, that is, suitable coordinate transformations.

Remark 2.2. This is example 6 in [CG95a]. The following DAE, where we have denoted x, y, z instead of y^1, y^2, y^3,

$$\sin(y')y + x = 0 \tag{6}$$

$$\sin(z')z + y = 0 \tag{7}$$

$$z = 0 \tag{8}$$

has index 3, yet it is clearly equivalent with $x = y = z = 0$, which has index zero. Hence the concept of the differential index is not intrinsic but depends on the chosen representation.

2.2 Index reduction

As noted before, in conventional DAE approaches the index should be as low as possible, in practice this means one or zero, for the system to be suitable for numerics. Therefore, there has evolved some techniques which transform the given system to a new one which has lower index. A common name for these techniques is *index reduction*, see e.g. [Gea88].

Now there are at least three different kind of approaches here: first, simply replacing certain equations of the system by their derivatives, for example replacing (2) by (4). Second, so called Baumgarte stabilization, which replaces an equation by a linear combination of the same equation and its derivatives up to some order. Third, a more intrinsic approach: in [Rei90, RR94a] the system is interpreted as a locus in suitable space and through a differentiation-elimination process (which is defined in an intrinsic way, by geometrical terms, see [RR94a] or [RLW01] for details) the system is replaced by another one.

This latter one has been shown [Kor97, RLW01] (independently) to be equivalent with the Cartan-Kuranishi algorithm. This requires certain constant rank conditions but we think this is still a correct way to transform the system, in the sense that the structure of the system is not changed. However, this is not always constructive, since it requires finding suitable coordinates (which certainly exist).

The second approach, in spite of its name, might make the system numerically quite instable. A geometrical interpretation for this approach can be given by the pendulum example: a vector field *around* the locus of the last row of (1) is defined such that it points towards the locus. The problem is: how do you choose the coefficients of the linear combination? A bad choice can make the system quite stiff. Continuing with the pendulum example, a geometrical interpretation for the stiffness is that the vector field around the locus is too steep towards the locus.

Remark 2.3. Another kind of index reduction technique, which seems to be between the 'second' and 'third' above, is described in [Sei95]. It is applied to quasilinear first order Lagrangians and called symplectic index reduction, due to the fact that it maintains the symplectic structure.

The first approach is, in our opinion, throwing away information of the system and therefore might even destroy the structure of the system, hence should be avoided. An effect of this is the drift-off of the length of our pendulum, see section 1.3.

2.3 Numerics

In conventional DAE numerics, one is always considering systems which have n equations, where n is also the number of dependent variables: $y = (y^1, \ldots, y^n)$. Systems with $k > n$ equations are called overdetermined. We note that it seems like any interesting (that is, a "higher index") DAE generically has an involutive form (in the sense of [A], see also section 3.4) which is overdetermined.

Overdetermined systems are usually considered to be ill-posed since a small perturbation could make the system nonsolvable. In conventional DAE numerics, when an overdetermined system is encountered, some least squares solution is used. This has two side-effects: the first one is the drift-off, which we encountered already in section 1.3. On the other hand, this drifting can be eliminated by some kind of projection, which forces the computed points to satisfy the constraints. Yet if this is done in an arbitrary manner, it leads to another problem, which is the second side-effect: instability. See also [CM95] and references therein for a discussion.

New view points

So, our approach usually is looking at a overdetermined system, in the conventional language. Does this make it worthless by the comments above?

The first thing to note is that since we are considering the locus of the equation as a subset $\mathcal{R}_q$ *in a jet space*, we have a few more (that is, $nq + n$ in case of (5)) dependent variables, so a situation $k > n$ is actually *not* overdetermined. One immediately argues that the y', y'' etc. depend on the y and hence it has only n dependent variables. However, the whole point of the jet approach is to put y *and* y', $\ldots$, $y^{(q)}$ to an equal setting. Their interdependence is represented by the *Cartan distribution* which is the $\mathcal{C}_p$ in [A, (3.3)]. Now our locus is not overdetermined but of dimension $nq+n+1-k$.

The only place where we meet 'overdeterminacy' in our method is when we evaluate our distribution $\mathcal{D}_p$, it is the kernel of an overdetermined (numerical) matrix. Now one could say that this is ill-posed: numerical round-off errors make the kernel vanish. But it actually is not, because we read (a spanning vector of) $\mathcal{D}_p$ from the singular value decomposition of the corresponding matrix, see [A, section 6.1]. This task is stable.

As already mentioned, our method includes also a projection: this also is a stable task when it is defined to be orthogonal. We have implemented it as a classical newton iteration.

The worst side-effect of our approach is that due to the bigger dimension of the J_q it is more costly. Especially, the newton iteration in the projection is the dominant part of the algorithm. One could say that having undrifting *and* stability simultaneously has a high price.

Remark 2.4. Since we do not have a restriction like $k = n$ for the number of equations, we can easily maintain all such invariants which are formulated as equations. It is interesting to note that the more equations we have, the smaller dimension our locus has. Hence, the *more* equations, the *simpler* the system.

Remark 2.5. Another view point which seems to be forbidden in conventional numerical approaches, is that the value of $y'(t)$ is not necessarily uniquely defined by the point (t, y). However, such a situation can already be found from a very simple geometrical problem, see [A, example 2.3.2]. Let us emphasize that in such a situation it is natural to consider y' as another (dependent) variable.

Remark 2.6. One immediate way to reduce the cost of computation would be to (locally) parametrize $\mathcal{R}_q$; then we could avoid the projection since $p + h\,V_p$ (V_p is the direction of the solution at p, see [A] and [B]) will stay on the parametrization domain (which is $\mathbb{R}^{nq+n+1-k}$, i.e. a linear space). However, this does not really make things intrinsic: the choice of the parametrization chart affects the 'effect of h'. How can you then do step size control? How "close" to each other are $\pi(p + h\,V_p)$ and $\varphi^{-1}(\varphi(p) + h(\varphi_*(V))$ (where φ is the parametrization, φ_* its tangent map and π is our orthogonal projection to $\mathcal{R}_q$), when "close" should be considered by the metric of J_q ?

Besides, constructing a parametrization, for example by choosing suitable jet coordinates as is done in [KM98] (although they do not mention jets), is not necessarily a cheap operation; it requires several rank evaluations of the jacobian. Not to mention, all objects included in the computation should be re-evaluated through the parametrization. Although, we admit that if the parametrization could be done globally, it might be reasonable. However, we have not studied this aspect.

Remark 2.7. Sometimes it is considered of finding consistent initial values (see e.g. [CG95b] and references therein). In our approach it is clear that the answer to the question "is the given initial value consistent?" is positive if and only if the initial value fulfills the involutive form (or complete form, in a nonsmooth, general polynomial case). We do not consider this question (of consistency) as an important one, since it is a corollary of a more important question: find involutive form.

As a convenient side effect(!) of our approach, we do not need to give the (consistent) initial value with multiple decimals, as is done in conventional

DAE approaches, see e.g. [HW91]. Using involutive form with our projection ensures that the (projected) initial value will be consistent.

2.4 Symbolics

By symbolics, we mean the symbolic computation, that is computation with symbolic variables and exact numbers. Let us note that this has a synonym "computer algebra", which seems to have become more widely used. In our approach we first do the symbolic computation to get a form suitable for numerical computation, and only then do the actual numerics, i.e. produce lots of those marvelous floating point numbers! We have in [D, section 5.2] a discussion of this, so we will keep this section short.

In [A] and [B] we were counting on Cartan-Kuranishi; that is, prolong and project, until it stabilizes. However, a reasonable approach is to consider arbitrary polynomials, and then we noted [D, remark 2.2] that there is a need to reconsider the algebraic point of view to involutivity, taking into account our definition of a solution. See also remark 3.4.

It is worth noting that a lot of equations which are not differential polynomials, can be transformed to such by introducing more variables.

We give some comments on the algorithms of *Maple* which seems to have the most up-to-date computer algebra packages relevant to our study:

- **diffalg** which is based on the Rosenfeld-Gröbner algorithm, see [BLOP].
- **DEtools** which is large package based on the article [CTvB95]. **DEtools** includes also as a subpackage the **rifsimp** where "rif" stands for reduced involutive form, see [RWB96].
- in **pdetools** the "casesplit" command which is implemented to *Maple6*, is a user interface which automatizes the usage of both **diffalg** and **rifsimp**. See remark 3.3.
- **diffgrob2** which is based on differential Gröbner bases, see [MC97].

Remark 2.8. There is also a concept of *characteristic sets*, which we have not considered. That might be useful also in our context. However, we did not find it useful enough to our purposes, hence we discarded it. An interested reader should consult [Mis93, Rit50, ALMM99].

3 Comments on publications [A-D]

3.1 Comments on [A]

The paper [A] is based on the preprint [TA98]. There is presented the fundamental ideas of our approach, and analyzed the local error of some numerical basic methods. These numerical methods are (low order) Runge-Kutta methods with an extra property: an orthogonal projection to the locus $f^{-1}(0)$.

Now there is a concept of "Runge-Kutta with projections" in the conventional DAE analysis, see [HW91]. However, their projection is not the same as our approach: first, they project onto the locus *in* $\mathcal{E}$ while we project onto the locus in $J_q(\mathcal{E})$. Second, their projection is not necessarily orthogonal.

We emphasize that the orthogonality of our projection makes the methods stable, but there is a price to pay: the projection is clearly the dominating part of our numerical method and makes it very slow, although in our opinion more reliable, compared to the conventional ones. The future developments of our numerical solver will attack to the projection.

The methods are explicit euler, implicit euler, and implicit midpoint rule. Their order coincides with their classical versions. In fact, in [B] it is shown that any Runge-Kutta method of classical order less or equal to four sustains its order when we make it a projected version.

We have also made numerical testing of the methods to well-known examples from literature. A surprising effect is that the result is qualitatively different depending on whether or not we take in account the constant energy of the (Hamiltonian) system. This holds true even in the case of the midpoint method, which keeps the energy almost constant (when computing without the equation energy=constant), in other words, keeps the drift-off quite small.

Remark 3.1. We note that the results of [A] seem to be in conflict with the well-known principle that solving DAEs is not the same as solving ODEs, which can be seen also in the title of [Pet82].

Now, the apparent conflict is resolved by noting that terminology "ODE" in the title of [Pet82] refers to an equation of the form $y'(t) = f(t, y)$, whereas in our geometrical approach there is no need to such a specification. See [A, remark 3.6] for more details.

3.1.1 The involutive form of the pendulum

As mentioned at the end of section 3.1 of [A], we want to transform a system to an involutive form. As an example we show how [A, (4.1)] (that is, (2) of this introduction) transforms to [A, (4.2)]. For a quick review of the Cartan-Kuranishi algorithm, see [D, sec. 2.3].
Step 1: now $q = 2$ and we get

$$B = \begin{pmatrix} 1 & & \\ & 1 & \\ & & 0 \end{pmatrix}, \qquad \tilde{f} = \begin{pmatrix} y^1_1\, y^3 + y^1\, y^3_1 \\ y^2_1\, y^3 + y^2\, y^3_1 \\ 2y^1\, y^1_1 + 2y^2\, y^2_1 \end{pmatrix} \tag{9}$$

Step 2: B is now constant, hence $\ker(B^T)$ is too, and clearly $v := (0, 0, 1)$ is a basis for $\ker(B^T)$.
Step 3: now $\nu = 1$ and we have the equation

$$0 = 2y^1\, y^1_1 + 2y^2\, y^2_1 = y^1\, y^1_1 + y^2\, y^2_1 \tag{10}$$

which is algebraically independent of (2).

Step 4: append (10) to f to get:

$$f := \begin{cases} y_2^1 + y^1\, y^3 & = 0 \\ y_2^2 + y^2\, y^3 + 1 & = 0 \\ (y^1)^2 + (y^2)^2 - 1 & = 0 \\ y^1\, y_1^1 + y^2\, y_1^2 & = 0 \end{cases} \tag{11}$$

then step 1 again:

$$B = \begin{pmatrix} 1 & & & \\ & 1 & & \\ & & 0 & \\ & & & 0 \end{pmatrix}, \qquad \tilde{f} = \begin{pmatrix} y_1^1\, y^3 + y^1\, y_1^3 \\ y_1^2\, y^3 + y^2\, y_1^3 \\ 2y^1\, y_1^1 + 2y^2\, y_1^2 \\ 2y^1\, y_2^1 + 2(y_1^1)^2 + 2y^2\, y_2^2 + 2(y_1^2)^2 \end{pmatrix} \tag{12}$$

Step 2: $v^1 := (0,0,1,0)$, $v^2 := (0,0,0,1)$ span $\ker(B^T)$.
Step 3: now $\nu = 2$ and we get the equations

$$\begin{cases} 0 & = 2y^1\, y_1^1 + 2y^2\, y_1^2 \\ 0 & = 2y^1\, y_2^1 + 2(y_1^1)^2 + 2y^2\, y_2^2 + 2(y_1^2)^2 \end{cases} \tag{13}$$

where the latter one is algebraically independent of (11), and the reduction with respect to (11) gives

$$\begin{aligned} 0 &= y^1\,(-y^1\, y^3) + (y_1^1)^2 + y^2\,(-y^2\, y^3 - 1) + (y_1^2)^2 \\ &= (y_1^1)^2 + (y_1^2)^2 - y^2 + (-(y^1)^2 - (y^2)^2)\, y^3 \\ &= (y_1^1)^2 + (y_1^2)^2 - y^2 - y^3 \end{aligned} \tag{14}$$

hence the new f:

$$f := \begin{cases} y_2^1 + y^1\, y^3 & = 0 \\ y_2^2 + y^2\, y^3 + 1 & = 0 \\ (y^1)^2 + (y^2)^2 - 1 & = 0 \\ y^1\, y_1^1 + y^2\, y_1^2 & = 0 \\ (y_1^1)^2 + (y_1^2)^2 - y^2 - y^3 & = 0 \end{cases} \tag{15}$$

and step 1 gives

$$B = \begin{pmatrix} 1 & & & & \\ & 1 & & & \\ & & 0 & & \\ & & & 0 & \\ & & & & 0 \end{pmatrix}, \qquad \tilde{f} = \begin{pmatrix} y_1^1\, y^3 + y^1\, y_1^3 \\ y_1^2\, y^3 + y^2\, y_1^3 \\ 2y^1\, y_1^1 + 2y^2\, y_1^2 \\ 2y^1\, y_2^1 + 2(y_1^1)^2 + 2y^2\, y_2^2 + 2(y_1^2)^2 \\ 2y_1^1\, y_2^1 + 2y_1^2\, y_2^2 - y_1^2 - y_1^3 \end{pmatrix} \tag{16}$$

step 2: $v^1 := (0,0,1,0,0)$, $v^2 := (0,0,0,1,0)$, $v^3 := (0,0,0,0,1)$.
step 3: now $\nu = 3$, immediately we see from previous round that $(v^1)^T\tilde{f}$ and $(v^2)^T\tilde{f}$ are dependent (they directly correspond to (10) and (14), respectively)

of (15). Hence we need only to check $(v^3)^T\tilde{f}$ which indeed gives a new, independent equation:

$$\begin{aligned}
0 &= 2y_1^1\,y_2^1 + 2y_1^2\,y_2^2 - y_1^2 - y_1^3 \\
&= 2y_1^1\,(-y^1\,y^3) + 2y_1^2\,(-y^2\,y^3 - 1) - y_1^2 - y_1^3 \\
&= -2(y^1\,y_1^1 + y^2\,y_1^2)\,y^3 - 2y_1^2 - y_1^2 - y_1^3 \\
&= -(3y_1^2 + y_1^3)
\end{aligned} \tag{17}$$

hence the new f:

$$f := \begin{cases} y_2^1 + y^1\,y^3 & = 0 \\ y_2^2 + y^2\,y^3 + 1 & = 0 \\ (y^1)^2 + (y^2)^2 - 1 & = 0 \\ y^1\,y_1^1 + y^2\,y_1^2 & = 0 \\ (y_1^1)^2 + (y_1^2)^2 - y^2 - y^3 & = 0 \\ 3y_1^2 + y_1^3 & = 0 \end{cases} \tag{18}$$

step 1 gives:

$$B = \begin{pmatrix} 1 & & \\ & 1 & \\ & & 0 \\ & & 0 \\ & & 0 \\ & & 0 \end{pmatrix}, \qquad \tilde{f} = \begin{pmatrix} y_1^1\,y^3 + y^1\,y_1^3 \\ y_1^2\,y^3 + y^2\,y_1^3 \\ 2y^1\,y_1^1 + 2y^2\,y_1^2 \\ 2y^1\,y_2^1 + 2(y_1^1)^2 + 2y^2\,y_2^2 + 2(y_1^2)^2 \\ 2y_1^1\,y_2^1 + 2y_1^2\,y_2^2 - y_1^2 - y_1^3 \\ 3y_2^2 + y_2^3 \end{pmatrix} \tag{19}$$

step 2: $v^1 := (0,0,1,0,0,0)$, $v^2 := (0,0,0,1,0,0)$, $v^3 := (0,0,0,0,1,0)$ $v^4 := (0,0,0,0,0,1)$. Now $\nu = 4$ but, as before, we know from previous round that v^1, v^2, v^3 do not give new equations, hence we check only $(v^4)^T\,\tilde{f}$ which indeed is algebraically independent of (18). The new f:

$$f := \begin{cases} y_2^1 + y^1\,y^3 & = 0 \\ y_2^2 + y^2\,y^3 + 1 & = 0 \\ (y^1)^2 + (y^2)^2 - 1 & = 0 \\ y^1\,y_1^1 + y^2\,y_1^2 & = 0 \\ (y_1^1)^2 + (y_1^2)^2 - y^2 - y^3 & = 0 \\ 3y_1^2 + y_1^3 & = 0 \\ 3y_2^2 + y_2^3 & = 0 \end{cases} \tag{20}$$

step 1 gives

$$B = \begin{pmatrix} 1 & & \\ & 1 & \\ & & 0 \\ & & 0 \\ & & 0 \\ & & 0 \\ 0 & 3 & 1 \end{pmatrix}, \qquad \tilde{f} = \begin{pmatrix} y_1^1\,y^3 + y^1\,y_1^3 \\ y_1^2\,y^3 + y^2\,y_1^3 \\ 2y^1\,y_1^1 + 2y^2\,y_1^2 \\ 2y^1\,y_2^1 + 2(y_1^1)^2 + 2y^2\,y_2^2 + 2(y_1^2)^2 \\ 2y_1^1\,y_2^1 + 2y_1^2\,y_2^2 - y_1^2 - y_1^3 \\ 3y_2^2 + y_2^3 \\ 0 \end{pmatrix} \tag{21}$$

step 2: $v^1 := (0,0,1,0,0,0,0)$, $v^2 := (0,0,0,1,0,0,0)$, $v^3 := (0,0,0,0,1,0,0)$, $v^4 := (0,0,0,0,0,1,0)$. Now $\nu = 4$ but again we know from previous round that none of $(v^j)^T \tilde{f}$ give new equations. Hence we are done and an involutive form of f is (20), which is exactly [A, (4.2)].

3.1.2 Erratum on [A]

In [A], the last line of (7.6) should read

$$V = \big(1, y_1^1, y_1^2, -(y^1 + 2y^1y^2), -(y^2 + (y^1)^2 - (y^2)^2)\big)$$

However, the numerical computations were done with the correct V, above.

3.2 Comments on [B]

The paper [B] consists of the preprints [Tuo98, TA99]. The theory of [A] is extended to Runge-Kutta methods up to order four. Also Taylor type methods are analyzed, but they are not implemented. Multistep methods we did not even consider, since they are based on using linear combinations of previously computed steps, and our manifold is, in general, certainly not a linear space.

We implemented the methods: classical RK4, Fehlberg 4(5) and Dormand-Prince 5(4). Now any numerical method with a nonconstant stepsize needs some kind of strategy to adjust the stepsize. Now this seems to include always some heuristics, and the conventional Runge-Kutta methods (that is, when the manifold is $\mathbb{R}^n$ and there is no projections) have quite well-tested strategies, see [HNW87]. However, our numerical testing indicates that these need some kind of revision in our case. This task we have not (yet) considered.

Another goal, which we were unable to reach, is to generalize (or disprove) the result on the order of our projected Runge-Kutta methods. That is, to answer to conjecture 4.1 of [B].

3.3 Comments on [C]

In [C] we apply the ideal decomposition to regularize certain singularities appearing in multibody systems. This is a bit side track compared to [A,B,D] which are very closely, even sequentially, related.

The main idea is that sometimes the models contain singularities which are due to a 'bad' choice of the model. Now ideal decomposition might be helpful in here, as has been shown in the examples. The decomposition gives several (but finitely many) systems, from which one can hopefully find a nonsingular one with desired geometrical properties.

There are two natural questions to the decomposition: first, how do we know what movements (that is, varieties of ideals), and why, are non-physical? Second, how do we know what ideal to choose? Both of these have been answered in [C, remark 3.2].

Other approaches to solve this problem of a 'bad' model is to somehow discover what is causing the singularity and to add (or multiply) suitable 'compensating' terms to the model. But that is based on ad hoc methods and requires quite a lot of experience to find such a trick. On the contrary, our method is based on prime decomposition which is constructive.

Remark 3.2. On the concluding remarks we mention the possibility to apply this to also some other kinds of singularity, and the reader might wonder what kinds of singularity we think. The reason for this is that the method, prime decomposition, works for any polynomial ideal. That is, any kind of polynomial singularity. What might be the corresponding *physical* singularities, is hard to say in advance. It depends on the designer of the model.

3.3.1 Errata on [C]

- We have mentioned that the decomposition is done over $\mathbb{C}$ and used *Singular* to compute the decomposition. Now *Singular* computes the decomposition over $\mathbb{Q}$ instead of $\mathbb{C}$. However, the results are correct also over $\mathbb{C}$.

- We mentioned that the prime decomposition is implemented on several computer algebra systems, including *Maple* and *Mathematica*. However, it seems that although on both of these have been implemented a computation of the Gröbner bases, the actual decompositions are not.

3.4 Comments on [D]

In [D] we expand our approach to arbitrary polynomial systems: then we need to look at the components of the system and also drop the requirement of the constant rank of the jacobian. In [A] the examples we considered were in *involutive* form, by which we have meant that there exist equations which globally describe the geometrical situation, that is, the (geometrical) involutivity of the system.

Now involutivity and constant rank conditions (i.e. a certain symbol involved is a vector bundle) implies [Pom83, prop. 4.34] that the system is a prime differential ideal. Hence it is natural to consider prime differential ideals to be "the" (components of) structure of the system, see also remark 1.3 of this introduction. However, as noted in [D, rem. 2.2], we cannot use differential ideals, which in formal theory is the most natural tool to define "the" components of f. Instead, [D, rem. 2.3] has a discussion about the definition of the structure.

It is important to note that the prime decomposition, hence also our concept of *structure*, depends on the chosen ground field, which we have chosen to be $\mathbb{Q}$. With a larger field we might get more components. However, we find this choice to be the most reasonable, for two reasons: first, since we are working with characteristic zero, we can not have smaller than $\mathbb{Q}$. Second, since we are aiming to a numerical computation, at the end what we have is a collection of floating point numbers, representation of which certainly does not need any field bigger than $\mathbb{Q}$.

Remark 3.3. To compare the packages of *Maple* (see section 2.4), it is reasonable to look at the “casesplit” procedure: it splits the system in to different cases and it is natural to ask whether this is related to our splitting (in [D]). These cases are represented by pairs (E, I) where E is a set of equations and I a set of inequations, that is, $I = \{g_1 \neq 0, \ldots, g_r \neq 0\}$. Now, in [D], each (component of the) system is a pair of (bases of) ideals (A, B) which means a point set $\mathcal{V}(A) - \mathcal{V}(B)$, that is, at least one element of B is $\neq 0$, *but not necessarily all*. So, our pair (A, B) is not used in the same way as in the “casesplit” procedure, which produces cases based on the pivot elements.

Remark 3.4. There is also a recent, quite interesting approach to define involutivity in algebraic terms: [GB97a, GB97b]. Their work is based on that of Janet; they consider *involutive divisions* of the variables, yet this is not in the scope of this Thesis.

4 Conclusions

One could say that there is a dilemma when studying (numerically) the DAEs: given a DAE, one wants to (1) know its structure and (2) compute approximations to solutions. Now, the goal (1) is best achieved when one finds intrinsic properties of the DAE, but these are usually defined in a nonconstructive way, which prevents performing (2). Or, if (1) is successfully done, it often makes (2) very laborious. On the other hand, forgetting (1) makes it possible to perform (2) in a relatively efficient way, but the quality of approximations rests on heuristics.

In [D] we have attempted to define 'structure' of a polynomial DAE in a constructive way. The construction is done by computer algebra, that is, symbolic computation. This is, however, very time-consuming and so far restricted to systems of moderate size. Of course the situation will become better since the computer algebra techniques are evolving rapidly.

The goal (2) seems to be done, for systems in practical applications, best by BDF (backward differentiation formula) methods. But what is 'reasonable' in approximation? There seems to be no other answer except a human eye, that is, heuristics: one runs several computations varying the initial point a little and hoping that the computed numerical solution will also vary only a little, until one gets convinced that the computed solution(s) is (are) a reasonable approximation to a correct one. Unfortunately, these fast methods depend on the chosen representation of the DAE.

We propose that when numerically solving DAEs, one should first use symbolic computation to find out the structure of the DAE, although it is at this point of research limited to moderate size systems, and *only then* use a numerical method which is based on geometrical ideas, hence does not depend on the chosen representation of the system. Especially, the numerical solvers should not be based on the representation of the system but instead on the intrinsic properties of DAEs.

References

[ALMM99] P. Aubry, D. Lazard, and M. Moreno Maza. On the theories of triangular sets. *J. Symb. Comp.*, 28:105–124, 1999.

[BCP89] K.E. Brenan, S.L. Campbell, and L.R. Petzold. *Numerical Solution of Initial-Value Problems in DAEs.* North-Holland, 1989.

[BLOP] F. Boulier, D. Lazard, F. Ollivier, and M. Petitot. Representation for the radical of a finitely generated differential ideal. In *Proc. of ISSAC 1995.*

[CG95a] S.L. Campbell and C.W. Gear. The index of general nonlinear DAEs. *Numer. Math.*, 72:173–196, 1995.

[CG95b] S.L. Campbell and E. Griepentrog. Solvability of general differential algebraic equations. *SIAM J. Sci. Comp*, 16(2):257–270, 1995.

[CM95] S.L. Campbell and E. Moore. Constraint preserving integrators for general nonlinear higher index DAEs. *Numer. Math.*, 69:383–399, 1995.

[CTvB95] E.S. Cheb-Terrab and K. von Bülow. Analytical solving of PDEs using symbolic computing. Technical Report 20, Univ. do Estado do Rio de Janeiro, Inst. de Física, 1995.

[GB97a] V. Gerdt and Y. Blinkov. Involutive bases of polynomial ideals. Preprint E5-3, JINR, Dubna, 1997.

[GB97b] V. Gerdt and Y. Blinkov. Minimal involutive bases. Preprint E5-4, JINR, Dubna, 1997.

[Gea71] C.W. Gear. Simultaneous numerical solution of DAEs. *IEEE Trans. Circ. Th.*, 18(1):89–95, 1971.

[Gea88] C.W. Gear. DAE index transformations. *SIAM J. Sci. Stat. Comput.*, 9(1), 1988.

[HD90] E. Haug and R. Deyo, editors. *Real-Time Integration Methods for Mechanical System Simulation*, volume F 69 of *NATO ASI Series.* Springer, 1990.

[HNW87] E. Hairer, S. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I: nonstiff problems*, volume 8 of *Computational Mathematics.* Springer, 1987.

[HW91] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II: stiff and differential-algebraic problems*, volume 14 of *Computational Mathematics*. Springer, 1991.

[KM98] P. Kunkel and V. Mehrmann. Regular solutions of nonlinear DAEs and their numerical determination. *Numer. Math.*, 79:581–600, 1998.

[Kor97] T. Korvola. On the formal integrability of differential algebraic systems. Technical Report A381, Helsinki Univ. of Tech, April 1997.

[LV94] G. Le Vey. Differential algebraic equations - a new look at the index. Technical report, Institut de Recherche en Informatique et Systemes Aleatoires, 1994.

[Man96] E. Mansfield. A simple criterion for involutivity. *J. London Math. Soc.*, 2(54):323–345, 1996.

[MC97] E. Mansfield and P. Clarkson. Applications of the differential algebra package diffgrob2 to classical symmetries of differential equations. *J. Symb. Comp.*, 23(5):517–533, 1997.

[Mis93] B. Mishra. *Algorithmic Algebra*. Texts and Monographs in Computer Science. Springer, 1993.

[Olv86] P.J. Olver. *Applications of Lie Groups to Differential Equations*, volume 107 of *Graduate Texts in Mathematics*. Springer, 1986.

[Pet82] L. Petzold. DAEs are not ODEs. *SIAM J. Sci. Stat. Comp.*, 3(3), September 1982.

[Pom83] J. F. Pommaret. *Differential Galois Theory*, volume 15 of *Mathematics and Its Applications*. Gordon and Breach Science Publishers, 1983.

[Pry01] J. Pryce. A simple structural analysis method for DAEs. *BIT*, 41(2):101–131, 2001.

[PT93] O-P. Piirilä and J. Tuomela. Differential-algebraic systems and formal integrability. Technical Report A326, Helsinki Univ. of Tech., november 1993.

[Rei90] S. Reich. On a geometrical interpretation of DAEs. *Circuits Systems Signal Process*, 9(4), 1990.

[Rei91] S. Reich. On an existence and uniqueness theory for nonlinear DAEs. *Circuits Systems Signal Process*, 10(3), 1991.

[Rhe84] W.C. Rheinboldt. Differential-algebraic systems as differential equations on manifolds. *Math. Comp.*, 43(168):473–482, 1984.

[Rit50] J. Ritt. *Differential Algebra.* Dover, 1950.

[RLW01] G. Reid, P. Lin, and A. Wittkopf. Differential elimination-completion algorithms for DAE and PDAE. *Stud. Appl. Math.*, 106:1–45, 2001.

[RR91] P.J. Rabier and W. Rheinboldt. A general existence and uniqueness theory for implicit DAEs. *Diff. Int. Eqns*, 4:563–582, 1991.

[RR94a] P.J. Rabier and W. Rheinboldt. A geometric treatment of implicit DAEs. *J. Diff. Eqns*, 109:110–146, 1994.

[RR94b] P.J. Rabier and W.C. Rheinboldt. On impasse points of quasilinear DAEs. *J. of Math. Anal. and Appl.*, 181:429–454, 1994.

[RR94c] P.J. Rabier and W.C. Rheinboldt. On the computation of impasse points of quasilinear DAEs. *Math. Comp.*, 62:133–154, 1994.

[RS88] R. Roberson and R. Schwertassek. *Dynamics of Multibody Systems.* Springer, 1988.

[RWB96] G.J. Reid, A.D. Wittkopf, and A. Boulton. Reduction of systems of nonlinear PDEs to simplified involutive forms. *Eur. J. Appl. Math.*, 7(6):635–666, 1996.

[Sei95] W. Seiler. Involution and constrained dynamics II: the Faddeev-Jackiw approach. *J. Phys. A: Math. Gen.*, 28:7315–7331, 1995.

[Sei99] W. Seiler. Indices and solvability for general systems of differential equations. In V. Ghanza, E. Mayr, and E. Vorzthsov, editors, *Computer Algebra in Scientific Computing – CASC 99*, pages 355–385. Springer, 1999.

[Sza92] A. Szatkowski. Geometric characterization of singular DAEs. *Int. J. Systems Sci*, 23(2):167–186, 1992.

[TA98] J. Tuomela and T. Arponen. On the numerical solution of involutive ordinary differential systems: introduction and numerical examples. Technical Report A393, Helsinki University of Technology, 1998.

[TA99] J. Tuomela and T. Arponen. On the numerical solution of involutive ordinary differential systems: higher order methods 2. Technical Report A409, Helsinki Univ. of Tech., 1999.

[Tuo97] J. Tuomela. On singular points of quasilinear differential and differential-algebraic equations. *BIT*, 37:966–975, 1997.

[Tuo98] J. Tuomela. On the numerical solution of involutive ordinary differential systems: higher order methods. Technical Report A397, Helsinki University of Technology, 1998.

(continued from the back cover)

A441 Jukka Liukkonen
Data Reduction and Domain Truncation in Electromagnetic Obstacle Scattering, Oct 2001

A440 Ville Turunen
Pseudodifferential calculus on compact Lie groups and homogeneous spaces, Sep 2001

A439 Jyrki Piila and Juhani Pitkäranta
On corner irregularities that arise in hyperbolic shell membrane theory, Jul 2001

A438 Teijo Arponen
The complete form of a differential algebraic equation, Jul 2001

A437 Viking Högnäs
Nonnegative operators and the method of sums, Jun 2001

A436 Ville Turunen
Pseudodifferential calculus on compact homogeneous spaces, Jun 2001

A435 Timo Salin
On Quenching with Logarithmic Singularity, May 2001

A434 Ville Havu
An Analysis of Asymptotic Consistency Error in a Parameter Dependent Model Problem, Mar 2001

A433 Ville Havu and Juhani Pitkäranta
Analysis of a bilinear finite element for shallow shells I: Consistency error, Jan 2001

A432 Tuomas Hytönen
Optimal Wrap-Around Network Simulation, Dec 2000

A431 Ville Turunen
Pseudodifferential calculus on compact Lie groups, Jun 2001

A430 Ville Havu and Juhani Pitkäranta
Analysis of a bilinear finite element for shallow shells I: Approximation of Inextensional Deformations, May 2000

A428 Jarmo Malinen
Discrete Time H^∞ Algebraic Riccati Equations, Mar 2000

HELSINKI UNIVERSITY OF TECHNOLOGY INSTITUTE OF MATHEMATICS
RESEARCH REPORTS

The list of reports is continued inside. Electronical versions of the reports are available at *http://www.math.hut.fi/reports/* .

A446 Tuomas Hytönen
R-Boundedness is Necessary for Multipliers on H^1, Feb 2002

A445 Philippe Clément, Stig-Olof Londen, Gieri Simonett
Quasilinear Evolutionary Equations and Continuous Interpolation Spaces, Mars 2002

A444 Tuomas Hytönen
Convolutions, Multipliers and Maximal Regularity on Vector-Valued Hardy Spaces, Dec 2001

A443 Tuomas Hytönen
Existence and Regularity of Solutions of the Korteweg - de Vries Equations and Generalizations, Dec 2001

A442 Ville Havu
Analysis of Reduced Finite Element Schemes in Parameter Dependent Elliptic Problems, Dec 2001

ISBN 951-22-5953-2
ISSN 0784-3143
Espoo, 2002