

# Polynomial Cost for Solving IVP for High-Index DAE

ROBERT M. CORLESS\* and SILVANA ILIE†

Ontario Research Centre for Computer Algebra and  
Department of Applied Mathematics, University of Western Ontario  
London, Ontario, Canada, N6A 5B7  
e-mails: rcorless | silvana@uwo.ca

## Abstract

We show that the cost of solving initial value problems for high-index differential algebraic equations is polynomial in the number of digits of accuracy requested. The algorithm analysed is built on a Taylor series method developed by Pryce for solving a general class of differential algebraic equations. The problem may be fully implicit, of arbitrarily high fixed index and contain derivatives of any order. We also show that adaptive meshes are never more expensive than non-adaptive meshes. Finally, we construct sufficiently smooth interpolants of the discrete solution.

*Key words:* differential algebraic equations, initial value problems, adaptive step-size control, Taylor series, structural analysis, automatic differentiation

*AMS subject classification:* 34A09, 65L80, 68Q25

## 1 Introduction

Many physical problems of interest today are modelled as differential algebraic equations (DAE). Therefore, the need for efficient numerical methods of solving them has grown rapidly [2, 23]. Among these numerical methods, Taylor series methods have recently proved to be an attractive choice on many interesting problems arising in practice.

---

\*Research supported by a grant from the Natural Sciences and Engineering Research Council of Canada

†Research supported by an Ontario Graduate Scholarship

The Taylor series method, due to Pryce [34, 35], is based on the structural analysis of Pryce [35] and Pantelides [33]. While Pantelides' structural analysis relies on solving an algorithm for iteratively determining the hidden constraints, Pryce's analysis is based on solving a priori an assignment problem, allowing, in addition, higher order derivatives. Chang and Corliss [14] showed how to compute Taylor series for particular cases of DAE.

The Taylor series coefficients can be efficiently computed with automatic differentiation. The idea of automatically generating recursion formulae for efficient computation of Taylor series goes back at least to Moore [30]. Full recurrence formulae for Taylor coefficients are attributed to Miller [29]. Applications of automatic differentiation to solving differential equations was considered in Corliss and Chang [13, 14].

In two previous papers, we studied the theoretical (asymptotic) complexity, as the accuracy of the computed residual,  $\varepsilon$ , gets better, of the numerical solution of IVP for ODE in [11] and for index-1 DAE in [26]. The main results of those papers are that the asymptotic cost of the numerical solution of such initial value problems is polynomial in the number of digits of accuracy,  $\mathcal{B} = \lceil \log_2(1/\varepsilon) \rceil$ . These results improved on the results of the standard theory of information based complexity (see, e.g., [39]), which predicted exponential cost of solving IVP for ODE. We believe that our model, assuming more smoothness of the problem and solution, is more realistic.

A second result is that adaptive step-size, under modest realistic assumptions, is never worse than fixed stepsizes and can be an unbounded factor better.

In the present paper, we extend the results to IVP for higher index DAE. We analyse an algorithm based on Pryce's method [34, 35] for solving numerically a general class of DAE that can be high-index, fully implicit and may contain derivatives of arbitrary order. The error control is in terms of defect (residual error), due to its low cost and straightforward interpretation [16, 18]. However, a similar analysis can be obtained for forward error.

The analysis of the Taylor series method for DAE (as discussed by Pryce [35] and Nedialkov & Pryce [32]) is complex. Another difficulty is an appropriate (smooth enough) choice of interpolant to facilitate the analysis. In the end, the analysis of this paper requires more effort than the index-1 case, as might be expected.

The paper assumes that the index of the problem does not change throughout the integration, and that singularities and discontinuities are isolated and may be located at negligible cost and that solutions are otherwise smooth. We also assume that the DAE is suitable for automatic differentiation [36, 21]. Under these assumptions, we show that the numerical solution of IVP for index- $n$  DAE is again of cost polynomial in the number of digits of accuracy.

The study of asymptotic cost for solving differential equations sheds light on finite precision and finite order methods. It is hoped that this theoretical analysis may be of use in the design and analysis of practical codes for highly accurate solutions of high-index differential algebraic equations.

## 2 Pryce's structural analysis method for DAEs

We consider an autonomous initial value problem for a differential algebraic equation in  $m$  dependent variables  $x_j = x_j(t)$ , with  $t$ , the scalar independent variable, in an interval  $[a, b]$ . The DAE has the general form

$$f_i(x_j, \text{derivatives of } x_j) = 0 \quad \text{for } i = 1 \dots m. \quad (1)$$

As is well-known, nonautonomous problems can be put in an autonomous form by increasing the dimension by one. The functions  $f_i$  are assumed piecewise analytic in their variables. We allow higher order derivatives of the dependent variables and nonlinear expressions in them.

We assume that if we add to the DAE extra ODEs that describe standard functions used in the functions  $f_i$ , then the problem is converted to a larger DAE whose new functions involve only the four basic arithmetic operations. The functions  $f_i$  belong thus to the class of functions to which automatic differentiation applies.

We assume that the initial condition satisfies a set of constraints which is sufficient for it to be a consistent point for (1).

A *consistent point* of the DAE (1) is defined as a scalar  $t^*$  together with a set of scalars  $\eta_{jl}$ , where  $(j, l)$  are all the elements of a finite set  $S_J$ , so that there exists a solution to the DAE (1) in a neighbourhood of  $t = t^*$  with  $x_j^{(l)}(t^*) = \eta_{jl}$  for all  $(j, l) \in S_J$  and the solution is unique. The set  $S_J$  need not be minimal.

In [35, 34] Pryce develops an algorithm for solving such a system by expanding the solution in Taylor series. The method is based on a pre-analysis stage which consists in solving an assignment problem. The assignment problem gives the *offsets*, that is a set of nonnegative integers  $c_i \geq 0$ ,  $d_j \geq 0$ , with  $c_i$  being the number of times the equation  $f_i$  has to be differentiated to reduce the DAE to an ODE and  $d_j$  is the highest order to which variable  $x_j$  appears in the corresponding ODE. The offsets also indicate which equations to solve for which unknowns, and give a systematic way of determining consistent initial conditions.

The next step of the method consists in generating the system Jacobian. Provided the Jacobian is non-singular at each integration step, the method succeeds and the Taylor series coefficients can be computed to the desired order. This is based on the fact that, at each step of integration, after some initial stages, the Tay-

lor coefficients can be obtained as solutions of some linear systems involving the system Jacobian.

## 2.1 Structural analysis

We review below the main steps of Pryce's structural analysis and the corresponding algorithm, following [35, 34].

1. Form an  $n \times n$  *signature matrix*  $\Sigma = (\sigma_{ij})$  with

$$\sigma_{ij} = \begin{cases} \text{order of derivative to which the variable } x_j \\ \text{appears in equation } f_i, \\ \text{or } -\infty \text{ if the variable does not occur.} \end{cases}$$

2. Solve an assignment problem to determine a HVT (*highest value transversal*), which is a subset of indices  $(i, j)$  describing just one element in each row and each column, such that  $\sum \sigma_{ij}$  is maximized and is finite. We make the assumption that such a HVT exists.
3. Determine the *offsets* of the problem, which are the vectors  $\mathbf{c}$ ,  $\mathbf{d}$  with  $d_i, c_j \geq 0$  the smallest such that

$$d_j - c_i \geq \sigma_{ij} \quad \text{for all } i \leq m, j \leq m$$

with equality on the HVT.

The *structural index* is then defined as

$$\nu = \max_i c_i + \begin{cases} 0 & \text{if all } d_j > 0 \\ 1 & \text{if some } d_j = 0. \end{cases}$$

The structural index is no less than the differential index on first order DAE [35].

4. Form the *system Jacobian*  $J_f$ , given by

$$J_{f_{ij}} = \begin{cases} \frac{\partial f_i}{\partial x_j^{(d_j - c_i)}} & \text{if } x_j^{(d_j - c_i)} \text{ appears in } f_i, \\ 0 & \text{otherwise.} \end{cases}$$

5. Choose a consistent point. If  $J_f$  is non-singular at that point, then the solution can be computed with Taylor series in a neighbourhood of that point.

We assume that

$$x_j(t) = \sum_{\ell \geq 0} \frac{1}{\ell!} x_{j,\ell} (t - t^*)^\ell,$$

substitute it in (1) and expand the equations in Taylor series

$$f_i(x_j(t), \text{ derivatives of } x_j(t)) = \sum_{q \geq 0} \frac{1}{q!} f_{i,q}(t - t^*)^q,$$

and solve  $f_{i,q} = 0$ . Each  $f_{i,q} = f_i^{(q)}(t^*)$  depends on a finite number of the  $x_{j,\ell} = x_j^{(\ell)}(t^*)$ .

## 2.2 Notation

We define  $k_d = -\max_j d_j$  and  $k_c = -\max_i c_i$  ( $k_d \leq k_c$ ). In this paper we consider  $k_c \leq -1$ , in which case the DAE (1) is at least of index 1. The case of index-1 DAEs with  $k_c = 0$  is considered in [26].

We follow the notation from [35] and define for  $k \geq k_d$ :

$$\begin{aligned} J_k &= \{(j, \ell) : \ell = k + d_j \geq 0, 1 \leq j \leq m\} \\ I_k &= \{(i, q) : q = k + c_i \geq 0, 1 \leq i \leq m\}. \end{aligned}$$

We consider the following notation

$$f_I = \{f_i^{(q)} : \text{the } q\text{-th derivative of } f_i, (i, q) \in I\}$$

for some set  $I$  and

$$x_J = \{x_j^{(\ell)} : \text{the } \ell\text{-th derivative of } x_j, (j, \ell) \in J\}$$

for some set  $J$ . We denote by  $I_{\leq k}$  ( $I_{<k}$ ) the union of all  $I_r$  with  $r \leq k$  ( $r < k$  respectively) and by  $J_{\leq k}$  ( $J_{<k}$ ) the corresponding unions.

We shall also use the notation  $f_{I_k}$  (and  $x_{J_k}$ ) for the values of the functions at a specific time and we shall omit to write the dependency of the time, if this is clear from the context.

## 2.3 Theoretical results

The choice of the offsets induces a specific block triangular structure on the system of equations to be solved and this structure is exploited by the algorithm. At each stage  $k$ ,  $f_{I_k}$  is a function only of the variables  $x_{J_{\leq k}}$ . The algorithm consists in solving, for each stage  $k = k_d, k_d + 1, k_d + 2, \dots$  the equations

$$f_{I_k} = f_{I_k}(x_{J_{<k}}, x_{J_k}) = 0 \quad \text{for variables } x_{J_k}. \quad (2)$$

This is equivalent to solving, for each  $k$ , the system  $f_{i,k+c_i} = 0$  for those  $i$  with  $k + c_i \geq 0$  in the unknowns  $x_{j,k+d_j}$  for those  $j$  with  $d_j + k \geq 0$ .

If we consider

$$m_k = |I_k|, \quad n_k = |J_k|,$$

where  $|I|$  represents the cardinality of set  $I$ , then  $m_k \leq n_k$  for all stages  $k$ , which means that at each stage we solve at most as many equations as unknowns (see [35]).

By permuting the variables  $x_j(t)$  so that the  $d_j$  are in decreasing order and by permuting the equations  $f_i$  so that the  $c_i$  are in decreasing order, it can be shown ([35], Proposition 4.1) that the Jacobian

$$J_{f_k} = \frac{\partial f_{I_k}}{\partial x_{J_k}} \quad (3)$$

is the leading  $m_k \times n_k$  submatrix of  $J_f$ . If  $J_f$  is non-singular, then  $J_{f_k}$  is full-row rank and thus there exists  $J_{f_k}^\dagger$ , the Moore-Penrose inverse of  $J_{f_k}$  (see, e.g., [5]).

If a point  $(t^*, x_{J \leq 0}^*) \in \mathbb{R} \times \mathbb{R}^{J \leq 0}$  satisfies (2) for all stages  $k \leq 0$  then it is a consistent point of the DAE (1). If the system Jacobian  $J_f$  is non-singular at this point and the functions  $f_i$  are analytic in a neighbourhood, then the DAE is solvable in some neighbourhood of this point and can be reduced to an ODE using this method ([35], Theorem 4.2). Alternatively, the DAE (1) can be solved by Taylor series in a neighbourhood [32, 35].

We assume the initial conditions

$$\frac{d^\ell}{dt^\ell} x_j(a) = x_{j,\ell}^0 \text{ for all } \ell \leq d_j$$

satisfy the stages  $k \leq 0$  of Pryce's algorithm at  $t = a$ ,

$$f_{I_k}(x_{J_k}^0) = 0$$

for all  $k \leq 0$  and thus is a consistent point for the DAE (1). We also assume that the system Jacobian is non-singular in a neighbourhood of the exact solution on the interval  $[a, b]$ .

We are interested in obtaining an approximate solution,  $\chi_j(t)$ , with a tolerance  $\varepsilon$  in residual errors  $\|\delta_i(t)\|$ , where

$$f_i(t, \chi_j(t), \text{derivatives of } \chi_j(t)) = \delta_i(t). \quad (4)$$

**Definition 1** *The approximate solution  $\chi_j(t)$  satisfies the DAE (1) with a tolerance  $\varepsilon$  in residual error if the residuals (4) satisfy, for all  $i \leq m$ ,*

$$\left\| \frac{d^k}{dt^k} \delta_i(t) \right\| \leq \varepsilon \quad \text{for all } 0 \leq k \leq c_i. \quad (5)$$

### 3 Numerical solution

We are interested in finding a bound for the cost of the following algorithm for obtaining a numerical solution of the DAE (1) using the structural analysis of [35]: assume we have obtained at time  $t_n$  some values  $x_{J \leq 0}^n$  which satisfy the constraints  $f_{I_{k \leq 0}} = 0$  more accurately than the desired tolerance. We generate the Taylor coefficients at  $t = t_n$  for the unknown functions up to the desired order and predict the values  $\hat{x}_{J \leq 0}^{n+1}$  at the next integration step by computing the Taylor series with a chosen stepsize  $h_n$ .

We correct these values by projecting the approximate solution back on the constraint manifold. We do this by applying only one Newton projection at each stage  $k \leq 0$ . As suggested by numerical experiments, one Newton iteration is sufficient for satisfying sufficiently accurately the constraint corresponding to that stage, see, e.g., [3]. Due to the use of a high order method, the approximate Taylor solution at one step stays very close to the exact solution at tight accuracies and the convergence of Newton iteration is assured. The advantage is that the cost of the algorithm is not increased unnecessarily.

The step is accepted if a given measure of the residual errors satisfy the tolerance (and thus guaranteeing, as shown later, that a continuous extension of the discrete solution satisfies (5)).

#### 3.1 Predictor step

We consider the truncated Taylor solution at time  $t = t_n + h$

$$\hat{x}_j(t) = \sum_{\ell=0}^{p+d_j-1} \frac{1}{\ell!} x_{j,\ell}^n (t - t_n)^\ell \quad \text{for } j \leq m. \quad (6)$$

Substitute it in the equations and expand in Taylor series:

$$f_i(\hat{x}_j, \text{derivatives of } \hat{x}_j) = \sum_{q \geq 0} \frac{1}{q!} f_{i,q}(x_{j,\ell}^n) (t - t_n)^q \quad \text{for } i \leq m.$$

We solve the systems  $f_{i,k+c_i}(x_{J \leq k}^n) = 0$  (for  $k + c_i \geq 0$ ) in the unknowns  $x_{j,k+d_j}^n$  (for  $k + d_j \geq 0$ ) for each  $k = k_d, k_d + 1, \dots, p - 1$ .

For  $k \leq 0$ , these equations are already satisfied with a much better accuracy than the tolerance. Indeed, at the first integration step, these equations are satisfied by the initial consistent point, and at the other integration steps, they are satisfied since the values are the ones projected on the constraint manifold at the previous step.

The stages  $k \geq 1$  are always linear in the corresponding unknowns and involve the Jacobian  $J_f(x_{j \leq 0}^n)$ , which is non-singular if the chosen tolerance is small enough.

On each interval  $[t_n, t_n + h_n)$ , the approximate Taylor solution (6) satisfies the problem

$$f_i(\hat{x}_j(t), \text{derivatives of } \hat{x}_j(t)) = \hat{\delta}_i(t), \quad i = 1 \dots m. \quad (7)$$

At time  $t_{n+1} = t_n + h_n$  the predicted values are

$$\hat{x}_{j,\ell}^{n+1} = \hat{x}_j^{(\ell)}(t_{n+1}^-) \quad (8)$$

for  $\ell \leq d_j, j \leq m$ .

The predicted point,  $\hat{x}_{J \leq 0}^{n+1}$  is close to a consistent point of the DAE, and its distance to the constraint manifold is given by the truncation error. So for a small  $\varepsilon$  the predicted point will be in a neighbourhood of the exact solution where the system Jacobian,  $J_f$ , is non-singular. In the same neighbourhood, each  $J_{f_k}$  given by (3) is full row-rank and there exists  $J_{f_k}^\dagger$ , the Moore-Penrose inverse of  $J_{f_k}$ .

### 3.2 Corrector step

For each  $k \leq 0$  we have at most as many equations as unknowns. We wish to satisfy these equations with a better accuracy than the one given by the truncation error. For each stage  $k \leq 0$  we compute a set of corrected values, by applying a single Newton iteration per stage. More would be superfluous.

For  $k < k_c$ , there are no constraints to satisfy and we preserve the values of the variables corresponding to that stage :  $x_{J_k} = \hat{x}_{J_k}$  for all  $k_d \leq k < k_c$  (if  $k_d < k_c$ ).

In what follows, the difference between the corrector at the end of the step,  $t_{n+1}$ , and the corresponding predictor defined by (8) will be denoted by

$$u_{J_k}^{n+1} = x_{J_k}^{n+1} - \hat{x}_{J_k}^{n+1} \quad (9)$$

for stages  $k = k_c, \dots, 0$ .

Using the notation (9), we start with the Gauss-Newton projection

$$u_{J_k}^{n+1} = -J_{f_k}^\dagger(\hat{x}_{J_k}^{n+1}) \cdot f_{I_{k_c}}(\hat{x}_{J_{k_c}}^{n+1}) \quad (10)$$

for stage  $k = k_c$ . We continue, at each stage  $k = k_c + 1, \dots, 0$ , with the following Gauss-Newton projections

$$u_{J_k}^{n+1} = -J_{f_k}^\dagger(\xi_k^{n+1}) \cdot f_{I_k}(\xi_k^{n+1}) \quad (11)$$

where

$$\xi_k^{n+1} = (x_{J_{<k}}^{n+1}, \hat{x}_{J_k}^{n+1}). \quad (12)$$

The values  $x_{J_{<k}}^{n+1}$  are already computed from the previous stages.

## 4 Error estimates

We shall analyse the errors of the Newton projections and we shall compute the residual errors over one integration step.

Since all  $f_i$  are analytic functions in their variables, the interval of integration is compact, and the system Jacobian is non-singular for the exact solution, then there exists a neighbourhood, in the space  $\mathbb{R}^{J \leq 0}$ , of the exact solution  $x_{J \leq 0}^e([a, b])$  such that the Jacobian  $J_f(z)$  is non-singular and all the Moore-Penrose inverses  $J_{f_k}^\dagger(z)$ , with  $k \leq 0$  are uniformly bounded on that neighbourhood; that is, there exists  $L > 0$  so that

$$\|J_{f_k}^\dagger(z)\| \leq L \quad \text{for all } k \leq 0. \quad (13)$$

For  $k_c + 1 \leq k$ ,  $f_{I_k}(y_{J \leq k})$  is *Lipschitz continuous* with respect to the variables  $\{y_{J_l}\}_{l=k_c, \dots, k-1}$  in some neighbourhood in  $\mathbb{R}^{J \leq 0}$  of the exact solution, if there exists  $M_k > 0$  so that

$$\begin{aligned} & \|f_{I_k}(y_{J < k_c}, y_{J_{k_c}}^1, \dots, y_{J_{k-1}}^1, y_{J_k}) - f_{I_k}(y_{J < k_c}, y_{J_{k_c}}^2, \dots, y_{J_{k-1}}^2, y_{J_k})\| \\ & \leq M_k \sum_{\ell=k_c}^{k-1} \|y_{J_\ell}^1 - y_{J_\ell}^2\| \end{aligned} \quad (14)$$

for all  $y_{J \leq k}^q = (y_{J < k_c}, y_{J_{k_c}}^q, \dots, y_{J_{k-1}}^q, y_{J_k})$ , with  $q = 1, 2$ , in that neighbourhood.

Let  $t = t_n + h$  and  $h \in [0, h_n]$ .

**Proposition 1** *If for all  $k_c + 1 \leq k \leq 0$ ,  $f_{I_k}(y_{J \leq k})$  is Lipschitz continuous with respect to the variables  $\{y_{J_l}\}_{l=k_c, \dots, k-1}$  in some neighbourhood in  $\mathbb{R}^{J \leq 0}$  of the exact solution; then, for  $\varepsilon$  small enough, the following holds:*

$$u_{J_k}^{n+1} = \mathcal{O}(h_n^{p-k}), \quad k_c \leq k \leq 0 \quad (15)$$

and

$$f_{I_k}(\hat{x}_{J \leq k}(t)) = \Phi_{I_k}^n h^{(p-k)} + \mathcal{O}(h^{p-k+1}), \quad k_c \leq k \leq 0 \quad (16)$$

$$f_{I_k}(\xi_k^{n+1}) = \Phi_{I_k}^n h_n^{(p-k)} + \mathcal{O}(h_n^{p-k+1}), \quad k_c + 1 \leq k \leq 0 \quad (17)$$

where the vectors  $\Phi_{I_k}^n$  do not depend on  $h_n$ .

**Proof.** Consider

$$\phi_i^n = \frac{1}{(p + c_i)!} \frac{d^{p+c_i}}{dt^{p+c_i}} f_i(x_{J \leq 0})(t_n).$$

These coefficients are independent of  $h$  and  $h_n$ , depending only on the value of some fixed function at  $(x_{J \leq 0}^n)$  (the beginning of the step). Then

$$f_i(\hat{x}_j(t), \text{ derivatives of } \hat{x}_j(t)) = \phi_i^n (t - t_n)^{p+c_i} + \mathcal{O}((t - t_n)^{p+c_i+1})$$

and thus, by differentiating  $(k + c_i) \geq 0$  times with respect to  $t$ , we obtain (16), where  $\Phi_{i,k+c_i}^n = ((p + c_i)! / (p - k)!) \phi_i^n$ , provided  $k \leq 0$ .

We also note that, by using (14) for  $k_c + 1 \leq k$ , we derive

$$f_{I_k}(\xi_k^{n+1}) = f_{I_k}(\hat{x}_{J_{\leq k}}^{n+1}) + \sum_{\ell=k_c}^{k-1} \mathcal{O}(\|u_{J_\ell}^{n+1}\|) \quad (18)$$

We now prove (15) for  $k$  and (17) for  $k + 1$  by induction over  $k \geq k_c$ .

- Let  $k = k_c$ . By using (10) and the uniform-boundedness condition (13), we obtain

$$\|u_{J_{k_c}}^{n+1}\| \leq L \|f_{I_{k_c}}(\hat{x}_{J_{k_c}}^{n+1})\|.$$

If we take  $k = k_c$  in (16), we derive

$$f_{I_{k_c}}(\hat{x}_{J_{k_c}}^{n+1}) = \mathcal{O}(h_n^{p-k_c})$$

and thus we obtained (15) for  $k = k_c$ .

If we consider  $k = k_c + 1$  in (18) and we apply (15) for  $k = k_c$  (obtained above), then we find :

$$f_{I_{k_c+1}}(\xi_k^{n+1}) = f_{I_{k_c+1}}(\hat{x}_{J_{\leq k_c+1}}^{n+1}) + \mathcal{O}(h_n^{p-k_c}).$$

- We now assume the hypothesis is true for all  $\ell \leq k - 1$  and show that it is true for  $k$ . By using (11) and (13), we obtain

$$\|u_{J_k}^{n+1}\| \leq L \|f(\xi_k^{n+1})\|.$$

But from the induction step at  $k - 1$  we know that

$$\|f(\xi_k^{n+1})\| = \mathcal{O}(h_n^{p-k})$$

and thus (15) is also true for  $k$ .

The induction ends by showing (17) at  $k + 1$ . By applying the induction hypothesis (15) for  $\ell \leq k$  in (18) for  $k + 1$ , we obtain

$$f_{I_{k+1}}(\xi_k^{n+1}) = f_{I_{k+1}}(\hat{x}_{J_{\leq k+1}}^{n+1}) + \mathcal{O}(h_n^{p-k}).$$

**Remark 1** In Proposition 1, we only need the Lipschitz condition (14) to hold for small right hand sides in (14). This is satisfied when each derivative  $\frac{\partial f_{I_k}}{\partial y_{J_\ell}}$  is bounded, for all  $\ell = k_c, \dots, k-1$ , on some neighbourhood of the exact solution. In particular, if  $f_i$ ,  $i = 1, \dots, m$  and the exact solution contains only analytic functions on the interval of integration, then Proposition 1 applies.

**Remark 2** According to Proposition 1, for each stage  $k = k_c, \dots, 0$ , the Taylor residuals together with the errors committed in the projections, all corresponding to the stage  $k$ , are of *the same order*,  $\mathcal{O}(h_n^{p-k})$ .

**Remark 3** For small enough  $\varepsilon$ , the Newton projections double the number of digits of accuracy in the corresponding residuals,  $f_{I_k}(x_{J_{\leq k}}^{n+1}) = \mathcal{O}(h_n^{2(p-k)})$ , for  $k \leq 0$ . Indeed, by a Taylor expansion we obtain

$$\begin{aligned} f_{I_k}(x_{J_{\leq k}}^{n+1}) &= f_{I_k}(\xi_k^{n+1}) + J_{f_k}(\xi_k^{n+1})u_{J_k}^{n+1} \\ &+ \frac{1}{2}u_{J_k}^{n+1T} \cdot G_k^{n+1}u_{J_k}^{n+1} \end{aligned}$$

where

$$G_k^{n+1} = \int_0^1 2(1-s) \frac{\partial^2 f_{I_k}}{\partial x_{J_k} \partial x_{J_k}}(x_{J_{<k}}^{n+1}, \hat{x}_{J_k}^{n+1} + su_{J_k}^{n+1}) ds.$$

The first two terms in the expansion cancel each other due to (10)–(11), and then by using (13) we derive

$$\|f_{I_k}(x_{J_{\leq k}}^{n+1})\| \leq \frac{1}{2}N_k L^2 \|f_{I_k}(\xi_k^{n+1})\|^2$$

where  $N_k = \sup \left\| \frac{\partial^2 f_{I_k}}{\partial x_{J_k} \partial x_{J_k}} \right\|$  on some neighbourhood. We end the proof by applying Proposition 1.

## 4.1 Example

We consider the example of the simple nonlinear pendulum, which is described by the following equations

$$\begin{cases} 0 = f_1 = x'' + x\lambda \\ 0 = f_2 = y'' + y\lambda - g \\ 0 = f_3 = x^2 + y^2 - L^2 \end{cases} \quad (19)$$

where  $x, y$  are the ordinary orthogonal coordinates and  $L\lambda$  is the tension in the string. The length of the pendulum,  $L$ , and the acceleration due to gravity,  $g$ , are positive constants.

The offsets of the variables  $(x, y, \lambda)$  are  $(2, 2, 0)$ , and those of the equations  $(f_1, f_2, f_3)$  are  $(0, 0, 2)$ . Since the system Jacobian is non-singular [35], then the system is solvable by Pryce's method. This is an index-3 DAE [35].

According to the algorithm, the equations to be solved are grouped as follows:

$(f_3)$  in unknowns  $(x, y)$  at stage  $k = -2$ ,

$(f'_3)$  in unknowns  $(x', y')$  at stage  $k = -1$ ,

$(f_1, f_2, f''_3)$  in unknowns  $(x'', y'', \lambda)$  at stage  $k = 0$ .

According to Remark 1, in order for Proposition 1 to apply, it suffices to check that the following are bounded:

(I). Stage  $k = -1$ :

$$\begin{bmatrix} \frac{\partial f'_3}{\partial x} & \frac{\partial f'_3}{\partial y} \end{bmatrix} = \begin{bmatrix} 2x' & 2y' \end{bmatrix}$$

(II). Stage  $k = 0$ :

$$\begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} & \frac{\partial f_1}{\partial x'} & \frac{\partial f_1}{\partial y'} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} & \frac{\partial f_2}{\partial x'} & \frac{\partial f_2}{\partial y'} \\ \frac{\partial f''_3}{\partial x} & \frac{\partial f''_3}{\partial y} & \frac{\partial f''_3}{\partial x'} & \frac{\partial f''_3}{\partial y'} \end{bmatrix} = \begin{bmatrix} \lambda & 0 & 0 & 0 \\ 0 & \lambda & 0 & 0 \\ 2x'' & 2y'' & 4x' & 4y' \end{bmatrix}$$

Indeed, with a compactness and smoothness argument, we see that all of these variables are bounded on some neighbourhood of the exact solution.

Hence Proposition 1 applies and we can derive the order of the residual errors and of the errors between the corrector and predictor for the variables at each stage. If, for a given order  $p$ , the approximate Taylor solution is computed according to formula (6), then these errors, at the initial stages, are those listed in Table 1.

## 5 Dense Output

In the present paper we are not concerned with finding the optimal interpolant (the one that minimizes some norm of the residual). We want to measure the size of the residual associated with a sufficiently accurate continuous approximation.

Table 1: The order of the Taylor residual errors and of the errors of projections, for the initial stages in the simple pendulum problem.

Stage	equations	order of residual	unknowns	order of error in projection
$k = -2$	$f_3 = 0$	$h^{p+2}$	$x, y$	$h^{p+2}$
$k = -1$	$f'_3 = 0$	$h^{p+1}$	$x', y'$	$h^{p+1}$
$k = 0$	$f_1 = 0$ $f_2 = 0$ $f''_3 = 0$	$h^p$	$x'', y'', \lambda$	$h^p$

Let  $[t_n, t_{n+1}]$  be the current interval of integration and  $t$  arbitrary in this interval. Consider  $\chi_j$  to be a piecewise-polynomial approximation of the variable  $x_j$ . On each interval  $[t_n, t_{n+1}]$ , define  $\chi_j$  to be the Hermite interpolant which satisfies the following conditions:

$$\begin{aligned} \chi_j^{(k)}(t_n) &= x_{j,k}^n \quad \text{for all } k = 0, \dots, p + d_j - 1 \\ \chi_j^{(k)}(t_{n+1}) &= x_{j,k}^{n+1} \quad \text{for all } k = 0, \dots, d_j. \end{aligned} \quad (20)$$

This interpolant has continuous derivatives up to order  $d_j$  on  $[a, b]$ .

If for each  $\chi_j$  we consider the basis

$$\begin{aligned} (t - t_n)^k & \quad \text{for } 0 \leq k \leq p + d_j - 1 \\ (t - t_n)^{p+d_j} (t - t_{n+1})^k & \quad \text{for } 0 \leq k \leq d_j \end{aligned}$$

then we find the following representation of the interpolant:

$$\chi_j(t) = \hat{x}_j(t) + \sum_{\ell=0}^{d_j} \alpha_{p+d_j+\ell}^j (t - t_{n+1})^\ell (t - t_n)^{p+d_j} \quad (21)$$

where

$$\alpha_{p+d_j+k}^j = h_n^{-(p+d_j+k)} \sum_{\ell=0}^k c_{k,\ell}^j h_n^\ell u_{j,\ell}^{n+1}$$

for all  $1 \leq j \leq m$  and  $0 \leq k \leq d_j$  and  $\hat{x}_j(t)$  is the truncated Taylor solution (6). The coefficients  $c_{k,\ell}$  are some constants depending only on  $p$  and the offsets.

Let  $R_j(t) = \chi_j(t) - \hat{x}_j(t)$ ,  $j \leq m$ , be the difference between the interpolant and the truncated Taylor solution (6). We find that

$$R_j(t) = \sum_{\ell=0}^{d_j} p_{j,\ell}(\tau) h_n^\ell u_{j,\ell}^{n+1} \quad (22)$$

where  $\tau = \frac{1}{h_n}(t - t_n)$ ,  $\tau \in [0, 1]$  and the polynomials  $p_{j,\ell}$  have coefficients depending only on  $p$  and the offsets (by convention, the polynomials corresponding to those variables which are not projected are identically zero). In fact, we can show that

$$\|R_{J_0}(t)\| \leq \alpha \sum_{k=0}^{k_c} h_n^k \|u_{J_k}^{n+1}\| \quad (23)$$

where we can choose, for example,

$$\alpha = \max |p_{j,\ell}^{(d_j)}(\tau)|$$

and the maximum is taken over all  $\tau \in [0, 1]$ , all  $j \leq m$ ,  $\ell \leq d_j$  and depends only on  $p$  and the offsets.

By using equations (10)–(11) and Proposition 1, we derive that

$$\|R_{J_0}(t)\| \leq \alpha \sum_{\ell=0}^{k_c} h_n^\ell \|J_{f_\ell}^\dagger(\xi_\ell^{n+1}) \cdot \hat{\delta}_{I_\ell}(t_{n+1}^-)\| + \mathcal{O}(h_n^{p+1}) \quad (24)$$

and is of order  $\mathcal{O}(h_n^p)$ .

## 5.1 Residual control

We are interested in measuring the magnitude of the residuals associated with the continuous approximation (21). Using Proposition 1, we obtain that at each stage  $k \leq 0$  on the interval  $[t_n, t_{n+1}]$ , the corresponding residual is of order  $\mathcal{O}(h_n^{p-k})$ , thus the dominating residual is obtained at stage  $k = 0$  and is of order  $\mathcal{O}(h_n^p)$ . This residual is given by

$$\delta_{I_0}(t) = f_{I_0}(\chi_{J \leq 0}(t)) . \quad (25)$$

By using (22) and a Taylor expansion of (25) around the predicted solution (6), we derive

$$\delta_{I_0}(t) = \hat{\delta}_{I_0}(t) + J_{f_0}(\hat{x}_{J \leq 0}(t)) \cdot R_{J_0}(t) + \mathcal{O}(h_n^{p+1}) . \quad (26)$$

Therefore, by using (23), (26) and Proposition 1, we find that the principal error term is bounded by  $(\alpha + 1)\psi_n h_n^p$  with

$$\psi_n h_n^p = \|J_{f_0}(\hat{x}_{J \leq 0}^{n+1})\| \sum_{\ell=0}^{k_c} h_n^\ell \|J_{f_\ell}^\dagger(\xi_\ell^{n+1}) \hat{\delta}_{I_\ell}(t_{n+1}^-)\| . \quad (27)$$

It is sufficient to require that  $\psi_n h_n^p \leq \frac{1}{\alpha + 1} \varepsilon$  to guarantee that the residual is below the tolerance (higher order terms are neglected).

**Remark 4** The maximum  $\alpha = \alpha(p)$  depends on  $p$  and there exists  $\mathcal{K} > 0$  so that  $[\alpha(p) + 1]$  is  $\mathcal{O}(p^{\mathcal{K}})$  ( $\mathcal{K}$  may be chosen, e.g., the highest among all degrees of the polynomials  $p_{j,\ell}^{(d_j)}$  with  $\ell = 0, \dots, d_j$ ).

## 5.2 Example

We return now to the nonlinear simple pendulum problem governed by the equations (19). We want to construct the corresponding interpolants, as described above, and estimate the residual error.

The Hermite interpolants corresponding to  $(x, y, z)$  and the step  $n$  of integration are respectively:

$$\begin{cases} \chi_1(t) &= \hat{x}(t) + p_{1,0}(\tau)u_{1,0}^{n+1} + h_n p_{1,1}(\tau)u_{1,1}^{n+1} + h_n^2 p_{1,2}(\tau)u_{1,2}^{n+1} \\ \chi_2(t) &= \hat{y}(t) + p_{2,0}(\tau)u_{2,0}^{n+1} + h_n p_{2,1}(\tau)u_{2,1}^{n+1} + h_n^2 p_{2,2}(\tau)u_{2,2}^{n+1} \\ \chi_3(t) &= \hat{\lambda}(t) + p_{3,0}(\tau)u_{3,0}^{n+1} \end{cases} .$$

Here,

$$\begin{cases} p_{1,0}(\tau) = p_{2,0}(\tau) &= \tau^{p+2}[1 - (p+2)(\tau-1) + \frac{1}{2}(p+2)(p+3)(\tau-1)^2] \\ p_{1,1}(\tau) = p_{2,1}(\tau) &= \tau^{p+2}(\tau-1)[1 - (p+2)(\tau-1)] \\ p_{1,2}(\tau) = p_{2,2}(\tau) &= \frac{1}{2}\tau^{p+2}(\tau-1)^2 \\ p_{3,0}(\tau) &= \tau^p . \end{cases}$$

and

$$\begin{cases} u_{1,j}^{n+1} &= x_j^{n+1} - \hat{x}_j^{n+1} \\ u_{2,j}^{n+1} &= y_j^{n+1} - \hat{y}_j^{n+1} \\ u_{3,\ell}^{n+1} &= \lambda_j^{n+1} - \hat{\lambda}_j^{n+1} \end{cases}$$

for  $j = 0, 1, 2$  and  $\ell = 0$ .

Using (22), we derive the errors in the interpolation at stage  $k = 0$ :

$$R_{J_0}(t) = \begin{pmatrix} \frac{1}{h_n^2} p_{1,0}^{(2)}(\tau)u_{1,0}^{n+1} + \frac{1}{h_n} p_{1,1}^{(2)}(\tau)u_{1,1}^{n+1} + p_{1,2}^{(2)}(\tau)u_{1,2}^{n+1} \\ \frac{1}{h_n^2} p_{2,0}^{(2)}(\tau)u_{2,0}^{n+1} + \frac{1}{h_n} p_{2,1}^{(2)}(\tau)u_{2,1}^{n+1} + p_{2,2}^{(2)}(\tau)u_{2,2}^{n+1} \\ p_{3,0}(\tau)u_{3,0}^{n+1} \end{pmatrix}$$

where

$$\begin{cases} p_{1,0}^{(2)}(\tau) &= p_{2,0}^{(2)}(\tau) = \frac{1}{2}(p+2)(p+3)(p+4)\tau^p(\tau-1)[(p+3)(\tau-1) + 2] \\ p_{1,1}^{(2)}(\tau) &= p_{2,1}^{(2)}(\tau) = -(p+2)(p+3)(\tau-1)[(p+4)(\tau-1) + 3] \\ p_{1,2}^{(2)}(\tau) &= p_{2,2}^{(2)}(\tau) = \frac{1}{2}(p+3)(p+4)(\tau-1)^2 + 2(p+3)(\tau-1) + 1 . \end{cases}$$

Thus, we derive  $\alpha(p) = \mathcal{O}(p^4)$ .

## 6 Polynomial cost of Pryce's algorithm

We show below that the cost of computing an approximate solution of the DAE (1) which satisfies the tolerance with the algorithm under investigation is polynomial in the number of digits of accuracy requested. Moreover, provided the local error coefficients given by (27) satisfy some regularity conditions described below, the cost of computing the solution with the same accuracy is minimized on the equidistributed mesh. In order to obtain this, we make use of a result on equidistribution from [11], which is given below.

For a vector  $\Psi = [\psi_1, \dots, \psi_N]$  we define the *s-norm* as

$$\|\Psi\|_s = \left( \sum_{i=1}^N \psi_i^s \right)^{1/s}$$

and the *Hölder s-mean* as

$$\mathcal{M}_s(\Psi) = \left( \frac{1}{N} \sum_{i=1}^N \psi_i^s \right)^{1/s}.$$

The *s-norm* is not a true norm for  $s < 1$ , when the triangle inequality is not always satisfied. We shall not need this property.

We assume that the vector of the error coefficients  $\Psi_N = [\psi_1, \dots, \psi_N]$  satisfies some regularity conditions. First, we assume that there exists a positive integer  $N_0$  such that  $\mathcal{M}_s(\Psi_{N_2}) \leq \mathcal{M}_s(\Psi_{N_1})$  for  $N_2 > N_1 \geq N_0$ . This means that if the mesh is sufficiently fine, the Hölder mean does not increase with further refinement. Second, we assume that the same property holds for the maximum norm of the local error coefficients,  $\|\Phi_N\|_\infty$ .

We evaluate the cost of an algorithm by counting the number of arithmetic operations, which is a more appropriate measure of efficiency for series methods than counting number of function evaluations [13]. Following the standard theory of computational complexity [39], we ignore memory hierarchy, overheads and interpolation costs.

The cost per step for the method investigated in this paper, for a fixed order, is the same for all steps and consists of the cost of obtaining the Taylor series plus the cost of  $[1 - k_c]$  Newton projections. Thus, the cost of the method for fixed order is directly proportional to the number of steps taken.

A mesh is optimal if it allows us to take the minimum number of steps to compute an approximate solution on the interval  $[a, b]$ , satisfying the tolerance.

**Theorem 1 (Minimax)** Given  $p, N \in \mathbb{N}$  and a vector with positive coefficients  $[\psi_i]_{1 \leq i \leq N}$ , the following inequality is true

$$\max\{\psi_i h_i^p : \sum_1^N h_i = b - a\} \geq (b - a)^p \|\Psi\|_{-1/p} = \bar{h}^p \mathcal{M}_{-1/p}(\Psi)$$

where  $\bar{h} = (b - a)/N$  is the average stepsize. Equality holds iff

$$\psi_i h_i^p = \bar{h}^p \mathcal{M}_{-1/p}(\Psi) \text{ for all } 1 \leq i \leq N.$$

A proof of this theorem based on Hölder's inequality is available in [11], and one by linearization in [26].

A first consequence of the Minimax Theorem (and the regularity condition on the Hölder mean of the local error coefficients) is that the equidistributing mesh ( $\psi_i h_i^p = \bar{h}^p \mathcal{M}_{-1/p}(\Psi) = \varepsilon$  for all  $1 \leq i \leq N$ ) is an optimal mesh for the algorithm. It is straightforward to derive that the number of steps corresponding to the equidistributing mesh is

$$N = \frac{b - a}{\bar{h}} = (b - a) \cdot (\mathcal{M}_{-1/p}(\Psi_N))^{1/p} \left(\frac{\varepsilon}{\alpha + 1}\right)^{-1/p}. \quad (28)$$

A second consequence of the Minimax Theorem (and regularity conditions on the Hölder mean and infinity norm) is that the cost of computing an approximate solution satisfying a given tolerance is never less on a fixed step mesh than on an equidistributed mesh for the above algorithm. In fact, the ratio of the two costs (the ratio of the number of steps corresponding to each mesh) is given by  $\|\Psi_{fix}\|_{\infty} / (\mathcal{M}_{-1/p}(\Psi))^{1/p}$ .

Proofs of the above results are similar to those in [11].

**Theorem 2** The minimal cost of computing the solution of (1) with Pryce's method is polynomial in the number of digits of accuracy requested and is bounded above by

$$\mathcal{C}(b - a)e^{2+(\mathcal{K}/e)} (\mathcal{M}_{-1/p}(\Psi_N))^{1/p} \mathcal{B}^4$$

where  $\mathcal{C}, \mathcal{K}$  are some constants. The minimum cost is reached on the equidistributing mesh.

**Proof.** Order- $p$  accurate solutions computed with Taylor series using automatic differentiation can be obtained in  $\mathcal{O}(p^2)$  operations (see [13]) and, if naive multiplication is used, the cost of doing arithmetics with  $\mathcal{B}$  bits of accuracy is  $\mathcal{O}(\mathcal{B}^2)$ . Thus the cost of obtaining the Taylor series at one step costs  $cp^2\mathcal{B}^2$ . The constant  $c$  depends on the dimension of the problem.

The cost of  $[1 - k_c]$  Newton projection is independent of  $p$  and costs  $\mathcal{O}(\mathcal{B}^2)$ . Therefore the cost of one step is  $C_1 p^2 \mathcal{B}^2$ .

Using (28), we obtain the total cost corresponding to the equidistributed mesh:

$$C_1(b-a)(\mathcal{M}_{-1/p}(\Psi_N))^{1/p} \mathcal{B}^2 p^2 \left(\frac{\varepsilon}{\alpha+1}\right)^{-1/p}. \quad (29)$$

Since there exist positive constants  $C_2$  and  $\mathcal{K}$  so that  $\alpha + 1 \leq C_2 p^{\mathcal{K}}$ , then

$$(\alpha + 1)^{1/p} \leq C_2 e^{\mathcal{K}/e}.$$

Choosing  $p = \lceil (\mathcal{B} \ln 2)/2 \rceil$  in (29) and  $\mathcal{C} = C_1 C_2 (\ln 2)^2/4$ , we obtain an upper bound of the minimum cost given by

$$\mathcal{C}(b-a)e^{2+(\mathcal{K}/e)}(\mathcal{M}_{-1/p}(\Psi_N))^{1/p} \mathcal{B}^4.$$

**Remark 5** The cost of computing the solution on a fixed-step mesh is also polynomial in the number of digits of accuracy. However, we note that the dimension of the problem is hidden in the constant factor in the expression of the cost.

We also remark that *very high accuracies* (of order  $10^{-128}$ ) have been very recently reported for solving DAEs with Pryce's method [3, 4]. Currently, existing software using Pryce's method may solve differential algebraic systems of hundred equations [31]. Detailed description on implementation and results of such a solver are given in [32]. These results support our theory which predicts polynomial cost in the number of bits of accuracy.

## 7 Conclusions

In this paper, we have investigated the cost of solving initial value problems for high-index differential algebraic equations depending on the number of digits of accuracy requested.

We analysed an algorithm based on a Taylor series method developed by Pryce for a general class of differential algebraic equations. We showed that the cost of computing a solution with this algorithm is polynomial in the number of digits of accuracy. We also showed that adaption is better than nonadaption and we include a sufficiently smooth dense output.

The cost of locating singularities and rank changes will be addressed in a future work.

**Acknowledgements** We thank Stephen Campbell, George Corliss, Wayne Enright, Ned Nedialkov and John Pryce for helpful discussions on DAE solving. This work was supported financially by a grant from the Natural Science and Engineering Council of Canada (Rob Corless) and by an Ontario Graduate Scholarship (Silvana Ilie).

## References

- [1] Allgower, E.L., Georg, K., *Introduction to numerical continuation methods*, SIAM Classics in Applied Mathematics, Philadelphia, 2003.
- [2] Ascher, U., Petzold, L., *Computer methods for ODE and DAE*, SIAM, Philadelphia, 1998.
- [3] Barrio, R., Performance of the Taylor series methods for ODEs/DAEs, *Applied Mathematics and Computation*, in press.
- [4] Barrio, R., Blesa, F., Lara, M., VSVO formulation of the Taylor method for the numerical solution of ODEs, *Computers & Mathematics with Applications*, to appear.
- [5] Ben-Israel, A., Greville, T.N.E., *Generalized inverses, theory and applications*, Wiley-Interscience, New York, 1974.
- [6] Brent, R.P., Fast multiple-precision evaluation of elementary functions, *J. Assoc. Comput. Mach* **23**(2) (1976) 242 – 251.
- [7] Brent, R.P., and Kung, H.T., Algorithms for composition and reversion of power series, in: J.F. Traub, ed., *Analytic Computational Theory*, Academic Press, New York, 1976, 217 – 225.
- [8] Butcher, J.C., *The numerical analysis of ordinary differential equations*, John Wiley & Sons Ltd., Chichester, 1987.
- [9] Corless, R.M., Error backward, *Chaotic Numerics, AMS Contemporary Mathematics Series* **172** (1994) 31 – 62.
- [10] Corless, R.M., An elementary solution of a minimax problem arising in algorithms for automatic mesh selection, *SIGSAM Bulletin: Commun. Comput. Algebra* **34**(4) (2001) 7 – 15.
- [11] Corless, R.M., A new view of the computational complexity of IVP for ODE, *Numerical Algorithms* **31** (2002) 115 – 124.

- [12] Corliss, G.F., Integrating ODE's in the complex plane—pole vaulting, *Mathematics of Computation* **35**(152) (1980) 1181 – 1189.
- [13] Corliss, G., Chang, Y.F., Solving ordinary differential equations using Taylor series, *ACM Transactions on Mathematical Software* **8**(2) (1982) 114 – 144.
- [14] Corliss, G., Chang, Y.F., ATOMFT: Solving ODEs and DAEs using Taylor series, *Computers and Mathematics with Applications* **28** (1994) 209 – 233.
- [15] Corliss, G.F., and Lodwick, W., Role of constraints in the validated solution of DAEs, Tech. Rep. 430, Marquette University, Milwaukee, USA, 1996.
- [16] Enright, W.H., A new error control for initial value solvers, *Appl. Math. Comput.* **31** (1989) 288 – 301.
- [17] Enright, W.H., Analysis of error control strategies for continuous Runge-Kutta methods, *SIAM J. Num. Anal.* **26**(3) (1989) 588 – 599.
- [18] Enright, W.H., Continuous numerical methods for ODEs with defect control, *J. Comp. Appl. Math* **125**(1-2) (2000) 159 – 170.
- [19] Enright, W.H., Jackson, K.R., Norsett, S.P., Thomsen, P.G., Interpolants for Runge-Kutta formulas, *ACM Transactions on Mathematical Software* **12**(3) (1986) 193 – 218.
- [20] Gladwell, I., Shampine, L.F., Baca, L.S., Brankin, R.W., Practical aspects of interpolation in Runge-Kutta codes, *SIAM J. Sci. Stat. Comput* **8**(3) (1987) 322 – 341.
- [21] Griewank, A., *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation*, Frontiers in Applied Mathematics, SIAM, Philadelphia, PA, 2000.
- [22] Hairer, E., Nørsett, S.P., Wanner, G., *Solving Ordinary Differential Equations I*, Computational Mathematics, Vol. 8, Springer, Berlin, 1987.
- [23] Hairer, E., Wanner, G., *Solving Ordinary Differential Equations II*, Computational Mathematics, Vol. 14, Springer, Berlin, 1991.
- [24] Hoefkens, J., Berz, M., and Makino, K., Efficient high-order methods for ODEs and DAEs, in: G.F. Corliss et al., eds., *Automatic Differentiation: From Simulation to Optimization*, Springer-Verlag, New York, 2001, 343 – 350.
- [25] van der Hoeven, J., Fast evaluation of holonomic functions, *Theoret. Comput. Sci.* **210** (1999) 199 – 215.

- [26] Ilie, S., Corless, R.M., Reid, G., Numerical solutions of index-1 differential algebraic equations can be computed in polynomial time, *Numerical Algorithms*, to appear.
- [27] Jackson, K.R., Nedialkov, N., Some recent advances in validated methods for IVPs for ODEs, *Appl. Numer. Math.* **42**(1) (2002) 269 – 284.
- [28] Kahan, W., Handheld calculator evaluates integrals, *Hewlett-Packard J.* **31**(8) (1980) 23 – 32.
- [29] Miller, J.C.P., *Introduction to tables of Weber parabolic cylinder functions*, National Physical Laboratory, H.M.S.O., London, 1955.
- [30] Moore, R.E., *Interval Analysis*, Prentice Hall, Englewood Cliffs, New York, 1966.
- [31] Nedialkov, N.S., private communication.
- [32] Nedialkov, N.S., Pryce, J.D. , Solving differential-algebraic equations by Taylor series (I): computing Taylor coefficients, *BIT*, accepted.
- [33] Pantelides, C., The consistent initialization of differential-algebraic systems, *SIAM J. Sci. Stat. Comput.* **9**(2) (1988) 213 – 231.
- [34] Pryce, J.D., Solving high-index DAEs by Taylor series, *Numerical Algorithms* **19** (1998) 195 – 211.
- [35] Pryce, J.D., A simple structural analysis method for DAEs, *BIT* **41**(2) (2001) 364 – 394.
- [36] Rall, L.B., *Automatic Differentiation: Techniques and Applications*, Springer-Verlag, Berlin, 1981.
- [37] Shampine, L., Solving ODEs and DDEs with residual control, *Appl. Numer. Math.* **52**(1) (2005) 113 – 127.
- [38] Söderlind, G., Automatic control and adaptive time-stepping, *Numerical Algorithms* **31** (2002) 281 – 310.
- [39] Werschulz, A.G., *The Computational Complexity of Differential and Integral Equations*, Oxford Science, Oxford, 1991.