Local error control for general index-1 and index-2 differential-algebraic equations

Jan Sieber, Berlin

Abstract

This paper presents an error test function usable for the local error control and the automatic stepsize selection in the numerical integration of general index-1 and index-2 differential-algebraic equations (DAEs). This test function makes a compromise between a good approximation of the error arising per step by the discretization (local error) and the order and smoothness assumptions made by the stepsize selection schemes, that are widely used in present codes. Its computation is of moderate costs and does not require additional information about the structure of the DAE from outside.

1 Introduction

One basic task in the analysis of differential-algebraic equations (DAEs)

$$f(x'(t), x(t), t) = 0 \tag{1.1}$$

with singular leading Jacobian $f'_{x'}$ is its numerical integration. The "Backward Differentiation Formula" (BDF)

$$f(\sum_{j=0}^{k} \frac{\alpha_{j,l}}{h_l} x_{l-j}, x_l, t_l) = 0 \tag{1.2}$$

has proved its value in practical computations. Convergence and perturbation results have been developed for index-1 and important classes of index-2 tractable DAEs of the general form (1.1) in [Mä92, Tis96, Fre95].
However, some problems limit the efficiency of popular numerical integration codes using the BDF:

1. The design of the local error control and automatic stepsize selection is identical with that used for explicit ODEs.

2. The BDF is weakly instable in the index-2 case. Defects arising in the solution of the nonlinear equation (1.2) are amplified by h_l^{-1}. Furthermore, the condition of the iteration matrix used in the Newton iteration applied to (1.2) is $\sim O(h_l^{-2})$ in the index-2 case.

3. Consistent initial values have to be computed before starting the integration.

This paper explores problem 1 for DAEs of index 1 and 2. Problem 2 has to be considered in the Newton iteration and is not in the scope of this paper.
Why is point 1 a "problem"? First of all we will roughly describe the stepsize selection algorithm of a typical implementation (e. g. `DASSL` by L. R. Petzold and `LSODI` by A. C. Hindmarsh):

estimate: After the computation of the new value x_l compare an estimate $\hat{\vartheta}_l$ of the *truncation error* ϑ_l (defined later) with the tolerance `TOL` given by the user.

error test: If $\|\hat{\vartheta}_l\| > \mathtt{TOL}$, reject the step, otherwise accept it.

new stepsize: Anyway, the new stepsize is

$$h_{\text{new}} := c \cdot \sqrt[k+1]{\frac{\mathtt{TOL}}{\|\hat{\vartheta}_l\|}} \cdot h_{\text{old}}. \tag{1.3}$$

k is the nominal order of the BDF used in this step. $c < 1$ is a safety factor. If the step was rejected, retry it with h_{new}, otherwise use h_{new} in the next step.

Questions and problems arising here are:

1. Does the truncation error ϑ_l used in the local error test really represent the error caused by the discretization of (1.1) with (1.2)?
2. The choice of the new stepsize assumes that

$$\hat{\vartheta}_l = \phi \cdot h_l^{k+1} + O(h_l^{k+2}) \sim O(h_l^{k+1}) \tag{1.4}$$

with negligible changes of ϕ per step.

Practical computations have shown that some components of $\hat{\vartheta}_l$ do not fulfil assumption (1.4) in the index-2 case. Integration turns out to be feasible only if these components have been excluded from error control. At least question 1 remains open in the index-1 case.
We try to develop an error indicator $\hat{S}_l$ replacing $\hat{\vartheta}_l$ in the local error test which

- catches the error made by the discretization of (1.1) with (1.2) as well as possible,
- fulfils the assumption (1.4) for index 1 and 2,
- does not increase the integration costs essentially (no additional function or Jacobian evaluations or decompositions) and
- does not require user-supplied information about the subspace structure of the DAE

The paper has the following structure:

Section 2: Introduction of basic notions and facts of linear constant coefficient DAEs.

Section 3: Generalization to the nonlinear case and essential suppositions.

Section 4: Various error definitions and their relations.

Section 5: Usual error estimates and definition of an alternative test function.

Section 6: Stepsize control with respect to the Newton iteration.

Section 7: Some numerical tests to illustrate the effects of the various test functions.

2 Linear constant coefficient DAEs

Since most of the tractability index concepts are derived from reflections on matrix pencils (A, B) and linear DAEs

$$Ax' + Bx + q(t) = 0 \tag{2.1}$$

with constant coefficients, we start with a consideration of (2.1).
Denote the nullspace of A by N:

$$N := \ker A.$$

We define projectors P and Q such that $\operatorname{im} Q = N$, $Q^2 = Q$ and $P = Id - Q$.
The space of possible solutions is

$$C^1_N(\mathcal{I}; \mathbb{R}^n) := \{x \in C(\mathcal{I}; \mathbb{R}^n) : Px \in C^1(\mathcal{I}; \mathbb{R}^n)\}.$$

Now consider the following matrix and subspace:

$$\begin{aligned} G_1 &:= A + B \cdot Q \\ S &:= \{z \in \mathbb{R}^n : B \cdot z \in \operatorname{im} A\}. \end{aligned}$$

If $N \cap S = \{0\}$ or ,equivalently, G_1 is regular, the problem will have (tractability) index 1 by definition.
Furthermore we treat index-2 problems. DAE (2.1) will have index 2 iff

- $N \cap S$ has nonzero dimension (or equivalently $\operatorname{rk} G_1 =: n_1 < n$) and
- the subspaces

$$\begin{aligned} N_1 &:= \ker G_1 \\ S_1 &:= \{z \in \mathbb{R}^n : BPz \in \operatorname{im} G_1\} \end{aligned}$$

  intersect trivially:

$$N_1 \cap S_1 = \{0\}. \tag{2.2}$$

For (2.2) there are projectors Q_1 onto N_1 along S_1 and $P_1 := \mathrm{Id} - Q_1$. This choice of Q_1 is called *canonical*, it has the property

$$Q_1 Q = 0. \tag{2.3}$$

The products PP_1, PQ_1 and QP_1 become projectors for canonical Q_1. The regularity of the matrix

$$G_2 := G_1 + BPQ_1$$

is equivalent to condition (2.2).

We decouple (2.1) to characterize the different parts of the DAE by the definitions given above. Index-1 DAEs (2.1) are decoupled by multiplication with PG_1^{-1} and QG_1^{-1}. Then the system

$$\begin{aligned} Px' + PG_1^{-1}BPx + PG_1^{-1}q &= 0 && (2.4)\\ Qx + QG_1^{-1}BPx + QG_1^{-1}q &= 0 && (2.5) \end{aligned}$$

is equivalent to (2.1). It consists of a regular ODE (2.4) and an algebraic equation (2.5) (an assignment to Qx). If (2.1) has index 2, it will be split by $PP_1G_2^{-1}$, $QP_1G_2^{-1}$ and $Q_1G_2^{-1}$ into

$$\begin{aligned} PP_1x' + PP_1G_2^{-1}BPP_1x + PP_1G_2^{-1}q &= 0 && (2.6)\\ -QQ_1x' + Qx + QP_1G_2^{-1}BPP_1x + QP_1G_2^{-1}q &= 0 && (2.7)\\ Q_1x + Q_1G_2^{-1}q &= 0 && (2.8) \end{aligned}$$

(2.6) is a regular ODE, (2.8) is an algebraic equation and (2.7) inherits a differentiation of $(P)Q_1x$ and (due to (2.8)) of $(P)Q_1G_2^{-1}q$.

Using the decoupled systems we can easily describe the *canonical projectors* Π_{can} onto the dynamic part of the DAE (2.1) (the subspace associated with the finite spectrum of the matrix pencil (A, B)) along the algebraic part (the subspace associated with the infinite spectrum of (A, B)).

$$\begin{aligned} \Pi_{\mathrm{can}(1)} &= (\mathrm{Id} - QG_1^{-1}B)P && \text{if } \mathrm{ind}(A,B) = 1 \text{ and}\\ \Pi_{\mathrm{can}(2)} &= (\mathrm{Id} - QP_1G_2^{-1}B)PP_1 && \text{if } \mathrm{ind}(A,B) = 2. \end{aligned}$$

We look at the decoupled index-2 DAE in more detail now. As mentioned above, (2.7) inherits a differentiation *and* an algebraic equation. We want to split it: Let N_C be a complement of $N \cap S$ in N: $N = (N \cap S) \oplus N_C$. $\mathbb{R}^n$ can be split into $\mathbb{R}^n = N \oplus N_1 \oplus \mathrm{im}\,\Pi_{\mathrm{can}(2)} = (N \cap S) \oplus N_C \oplus N_1 \oplus \mathrm{im}\,\Pi_{\mathrm{can}(2)}$. Introduce projectors V onto $N \cap S$ and U onto N_C such that $\mathbb{R}^n$ is split by $\Pi_{\mathrm{can}(2)}$, Q_1, U and V.

We rewrite the system as four equations in four subspaces by multiplying (2.6) by $\Pi_{\mathrm{can}(2)}$ and splitting (2.7) by U and V :

$$\begin{aligned} \Pi_{\mathrm{can}(2)}x' + \Pi_{\mathrm{can}(2)}G_2^{-1}BPP_1\Pi_{\mathrm{can}(2)}x + \Pi_{\mathrm{can}(2)}G_2^{-1}q &= 0 && (2.9)\\ Ux + UQP_1G_2^{-1}q &= 0 && (2.10)\\ Vx - (V\cdot)QQ_1x' + (V\cdot)QQ_1x + VQP_1G_2^{-1}q &= 0 && (2.11)\\ Q_1x + Q_1G_2^{-1}q &= 0. && (2.12) \end{aligned}$$

It is obvious that the part of x in $N \cap S = \operatorname{im} V$ is the result of a differentiation of $(P)Q_1 x$ mapped by Q.

We refer to [GM86, Mä92] for detailed introduction into matrix chains and projectors of regular matrix pencils.

The projectors Π_{can} are of interest for error control because the error in the dynamic part should be controlled as known from explicit ODEs. Theorem 2.1 shows a way to approximate Π_{can}.

Theorem 2.1 Let (A, B) be a regular matrix pencil of index μ, Π_{can} its canonical projector, P as defined above and $\eta > 0$ be sufficiently small. Then the following equation asymptotic in η holds:

$$\left[\left(\frac{A}{\eta} + B\right)^{-1} \frac{A}{\eta}\right]^{\mu} = \Pi_{can} + \Pi_{can} O(\eta) P \tag{2.13}$$

PROOF: (2.13) can be easily proved for arbitrary index μ using the transformation to Kronecker Normal form. Additionally the statement is derived using projectors for index 1 and 2 later. q. e. d.

Remark: The matrices A and $(\eta^{-1}A + B)^{-1})$ can be directly accessed in practical computations with $\eta = \alpha_{0,l}^{-1} \cdot h_l$. $\eta^{-1}A + B$ is evaluated and decomposed while processing the implicit BDF equation (1.2). Thus, (2.13) turns out to be of practical use.

Below we express the left-hand side of (2.13) without the exponent μ

$$\left(\frac{A}{\eta} + B\right)^{-1} \frac{A}{\eta}$$

using the projectors defined above for index 1 and 2. The statement of theorem 2.1 applied to index 1 and 2 follows directly.

Index 1: $G_1 = A + BQ$ is regular. Substituting

$$\frac{A}{\eta} + B = (G + \eta BP) \cdot \left(\frac{P}{\eta} + Q\right),$$

the left-hand side of (2.13) reads

$$\begin{aligned}
&\left(\frac{A}{\eta} + B\right)^{-1} \frac{A}{\eta} = \\
&= \quad (\eta P + Q) \cdot (G_1 + \eta BP)^{-1} \frac{A}{\eta} \\
&= \quad (\eta P + Q) \cdot \left(Id + \eta G_1^{-1} BP\right)^{-1} \frac{P}{\eta}
\end{aligned}$$

$$
\begin{aligned}
&= (\eta P + Q) \cdot \left(\sum_{j=0}^{\infty} (-\eta G_1^{-1} B P)^j \right) \frac{P}{\eta} \\
&= P - Q G_1^{-1} B P + (P - Q G_1^{-1} B P) \left(\sum_{j=1}^{\infty} (-\eta G_1^{-1} B P)^j \right) \cdot P \\
&= \Pi_{can(1)} + \Pi_{can(1)} O(\eta) P.
\end{aligned}
$$

Index 2: $G_2 = A + BQ + BPQ_1$ is regular. We substitute

$$
\frac{A}{\eta} + B = G_2 \cdot (Id + \eta G_2^{-1} B P P_1) \cdot (P_1 + \eta Q_1) \cdot \left(\frac{P}{\eta} + Q \right)
$$

in the left-hand side of (2.13) and take $G_2^{-1} A = P_1 P$, $Q P_1 P = -Q Q_1$ and $Q_1 G_2^{-1} B P P_1 = 0$ into account:

$$
\begin{aligned}
&\left(\frac{A}{\eta} + B \right)^{-1} \frac{A}{\eta} = \\
&= (\eta P + Q) \left(P_1 + \frac{Q_1}{\eta} \right) \left(\sum_{j=0}^{\infty} (-\eta G_2^{-1} B P P_1)^j \right) P_1 P \\
&= \frac{-Q Q_1}{\eta} + P P_1 - Q P_1 G_2^{-1} P P_1 + \\
&\quad + (P P_1 - Q P_1 G_2^{-1} P P_1) \cdot \left(\sum_{j=1}^{\infty} (-\eta G_2^{-1} B P P_1)^j \right) P P_1 (\cdot P) \\
&= \frac{-Q Q_1}{\eta} + \Pi_{can(2)} + \Pi_{can(2)} O(\eta) P P_1 (\cdot P). \qquad (2.14)
\end{aligned}
$$

Representation (2.14) will be used later. Keeping in mind that $Q_1 \Pi_{can(2)} = 0$, the product of two terms of form (2.14) is

$$
\left[\left(\frac{A}{\eta} + B \right)^{-1} \frac{A}{\eta} \right]^2 = \Pi_{can(2)} + \Pi_{can(2)} O(\eta) P P_1 (\cdot P). \qquad (2.15)
$$

3 Notions and suppositions in the nonlinear case

Let $f : \mathcal{G} \times \mathcal{I} \subseteq \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$ in (1.1) be continuous and continuously differentiable with respect to x' and x. Its partial derivatives are denoted by

$$
A(y, x, t) := \frac{\partial f}{\partial x'}(y, x, t) \qquad (3.1)
$$

$$
B(y, x, t) := \frac{\partial f}{\partial x}(y, x, t). \qquad (3.2)
$$

We use the pointwise linearizations, i. e. the definitions associated with the pointwise matrix pencils $(A, B)(y, x, t)$, in the nonlinear case:

The nullspace N of the leading Jacobian A is supposed to depend only continuously differentiable on t in the index-1 case and to be constant in the index-2 case. The definitions of S, N_1, S_1, G_1, Q_1 and G_2 can be applied pointwise to the matrix pencil $(A, B)(y, x, t)$ with these restrictions. These matrices and the canonical projectors Π_{can} depend continuously on (y, x, t) in the nonlinear case. Thus, (1.1) will be of

index 1 iff G_1 is regular on $\mathcal{G} \times \mathcal{I}$ and of

index 2 iff G_1 has constant rank $n_1 < n$ and G_2 is regular on $\mathcal{G} \times \mathcal{I}$.

Structural index-2 conditions to keep rk $G_1 = n_1 < n$ around a given solution are introduced and analysed e. g. in [Mä95, Tis96]. The *tractability index* definition given here matches the index definitions denoted by *differentiation index* or *perturbation index* (using the suppositions, that are necessary to define the various concepts, accordingly).

Later we use linearizations along short intervals:

The *mean values* $A[\xi_1, \xi_2]$ and $B[\xi_1, \xi_2]$ between ξ_1 and $\xi_2 \in \mathcal{G} \times \mathcal{I}$ are defined as

$$\begin{aligned} A[\xi_1, \xi_2] &:= \int_0^1 A(s \cdot \xi_1 + (1-s) \cdot \xi_2) ds \\ B[\xi_1, \xi_2] &:= \int_0^1 B(s \cdot \xi_1 + (1-s) \cdot \xi_2) ds. \end{aligned}$$

The definitions of all matrices G_i and projectors Q_i can be applied to $(A[\xi_1, \xi_2], B[\xi_1, \xi_2])$ as well. If the distance $[\xi_1, \xi_2]$ is sufficiently small, $(A[\xi_1, \xi_2], B[\xi_1, \xi_2])$ will be a small perturbation of the pointwise linearizations in ξ_1 or ξ_2.

Because of continuity, if the problem has index 1, the mean values $[\tilde{A}, \tilde{B}]$ along short paths will have it, too.

We are not able to reason in this way in the index-2 case. Even if rk $G_1(\xi) = n_1$ on $\mathcal{G} \times \mathcal{I}$ and the path between ξ_1 and ξ_2 is arbitrarily short, rk $G_1[\xi_1, \xi_2]$ can be greater than n_1. At least, there is no continuity argument.

Thus, we explicitly demand that

$$\text{rk } G_1[(y_1, x_1, t), (y_2, x_2, t)] = n_1 \tag{3.3}$$

along sufficiently short paths. Now the projector Q_1 of this mean value becomes a small perturbation of Q_1 of a near pointwise linearization. Any structural condition to ensure that rk $G_1 = n_1$ pointwise around a solution should keep the rank of mean values of kind (3.3) along sufficiently short paths n_1, too. The conditions introduced in [Mä95, Tis96] do so.

4 The basic concepts of error control

Let the interval $\mathcal{I}$ be discretized by a grid $\pi := t_0 < t_1 < \ldots t_N$ and denote by $x_* \in C^1_N(\mathcal{I}; \mathbb{R}^n)$ the true solution of $f(x', x, t) = 0$.

The function controlled in the error test (see section 1) does not represent the global error, i. e., the difference between the true solution of the initial value problem and the computed approximation, but an error made by one step, a "local error". It is defined (as in [HNW87]) in the true solution x_* or as the "error arising after the first step starting from exact initial values":

Definition 4.1 The *local error* Θ_l is the difference between $x_*(t_l)$ and the exact solution x_l^1 of

$$f(\sum \frac{\alpha_{j,l}}{h_l} x_{l-j}, x_l, t_l) = 0 \tag{4.1}$$

with the assumption that $x_{l-j} = x_*(t_{l-j})$ for $j = 1 \ldots k$:

$$\Theta_l := x_l^1 - x_*(t_l).$$

(4.1) is assumed to be uniquely solvable for sufficiently small h_l.

The *truncation error* usually estimated and controlled by integration routines is defined as the error of the backward difference quotient scaled by h_l in the true solution x_*:

Definition 4.2 The *truncation error* ϑ_l is defined as

$$\vartheta_l := h_l x_*'(t_l) - \sum_{j=0}^{k} \alpha_{j,l} x_*(t_l).$$

ϑ_l does not depend on the kind of equation (i. e. the index), but it is a property of the function x_*. Smoothness assumptions and Taylor expansion of x_* lead to

Lemma 4.3 Assume x_* to be sufficiently smooth, i. e. $x_* \in C^{k+2}(\mathcal{I}; \mathbb{R}^n)$, and the magnitude of $h_{l-j} = O(h_l)$ for $j = 1 \ldots k$. Denote the interpolation polynomial of degree k with nodes $(t_{l-k}, x_*(t_{l-k})), \ldots, (t_l, x_*(t_l))$ by q_l. Then there exist grid dependent constants $K_{1,l}$ and $K_{2,l}$ such that

$$\vartheta_l = K_{1,l} \cdot x_*^{k+1}(t_l) h_l^{k+1} + O(h_l^{k+2}) \tag{4.2}$$

$$\vartheta_l = K_{2,l} \cdot (q_l(t_l) - q_{l-1}(t_l)) + O(h_l^{k+2}). \tag{4.3}$$

q_{l-1} is the interpolation polynomial of degree k with the $k+1$ nodes $(t_{l-k-1}, x_*(t_{l-k-1}))$, $\ldots, (t_{l-1}, x_*(t_{l-1}))$ accordingly.

The grid dependent constants are

$$K_{2,l} = \frac{h_l}{t_l - t_{l-k-1}}$$

$$K_{1,l} = \frac{1}{(k+1)!} \prod_{i=1}^{k} \frac{t_l - t_{l-i}}{h_l}.$$

(4.2) shows that for sufficiently smooth f the truncation error is of the magnitude expected by the stepsize selection algorithm ($O(h_l^{k+1})$).
The relation between ϑ_l and Θ_l is expressed by the following theorem:

Theorem 4.4 x_l^1 is supposed to exist for sufficiently small h_l as defined in 4.1. If we denote the BDF approximation of the derivative associated with x_l^1 by y_l^1, i. e.

$$y_l^1 = \frac{1}{h_l} \sum_{j=0}^{k} \alpha_{j,l} x_{l-j}$$

with $x_l = x_l^1$ and $x_{l-j} = x_*(t_{l-j})$ for $j = 1 \ldots k$, we will call $\tilde{A}$ and $\tilde{B}$ the mean values of A and ,respectively, B between the first BDF solution (y_l^1, x_l^1, t_l) and $(x_*'(t_l), x_*(t_l), t_l)$. Then Θ_l and ϑ_l fulfil the relation

$$\left(\frac{\alpha_{0,l}}{h_l} \cdot \tilde{A} + \tilde{B} \right) \cdot \Theta_l = \frac{\tilde{A}}{h_l} \cdot \vartheta_l. \tag{4.4}$$

PROOF: $f(x_*'(t_l), x_*(t_l), t_l) = 0$ and $f(y_l^1, x_l^1, t_l) = 0$ by definition of x_* and (y_l^1, x_l^1, t_l). We define

$$y_{*,l} = \frac{1}{h_l} \sum_{j=0}^{k} \alpha_{j,l} x_*(t_{l-j})$$

and obtain

$$\begin{aligned} y_l^1 - y_{*,l} &= \frac{\alpha_{0,l}}{h_l}(x_l^1 - x_*(t_l)) \\ \vartheta_l &= h_l \cdot x_*'(t_l) - h_l \cdot y_{*,l}. \end{aligned}$$

Thus,

$$\begin{aligned} 0 &= f(y_l^1, x_l^1, t_l) - f(x_*'(t_l), x_*(t_l), t_l) \\ &= \tilde{A} \cdot (y_l^1 - x_*'(t_l)) + \tilde{B} \cdot (x_l^1 - x_*(t_l)) \\ &= \tilde{A} \cdot (y_l^1 - y_{*,l}) + \tilde{A} \cdot (y_{*,l} - x_*'(t_l)) + \\ &\quad + \tilde{B} \cdot (x_l^1 - x_*(t_l)) \\ &= \left(\frac{\alpha_{0,l}}{h_l} \tilde{A} + \tilde{B} \right) \Theta_l - \frac{\tilde{A}}{h_l} \vartheta_l. \end{aligned}$$

q. e. d.

Remarks:

- The matrix $\tilde{\Phi} := \alpha_{0,l} h_l^{-1} \tilde{A} + \tilde{B}$ with sufficiently small h is regular for regular pencils $(\tilde{A}, \tilde{B})$, and equation (4.4) can be multiplied by $\tilde{\Phi}^{-1}$:

$$\Theta_l = \left(\frac{\alpha_{0,l}}{h_l} \tilde{A} + \tilde{B} \right)^{-1} \cdot \frac{\tilde{A}}{h_l} \cdot \vartheta_l. \tag{4.5}$$

- (4.5) shows that only differential components $P \cdot \vartheta_l$ of the truncation error ϑ_l contain useful information about the local error.

Although relation (4.5) holds for $(\tilde{A}, \tilde{B})$ of arbitrary index, its representation depends strongly the index of the problem. We will analyse (4.5) in more detail with the aid of theorem 2.1 for index 1 and 2 and use the linear constant coefficient case ($\tilde{A} = A$ and $\tilde{B} = B$) to give some interpretation about the origin of the different terms. We mark projectors and matrices associated to $(\tilde{A}, \tilde{B})$ by a tilde.

Representation of Θ_l for index-1 DAEs: The pencil $(\tilde{A}, \tilde{B})$ will have index 1 if h_l is sufficiently small. Using theorem 2.1 the representation

$$\Theta_l = (\alpha_{0,l}^{-1}\tilde{\Pi}_{can(1)} + \tilde{\Pi}_{can(1)}O(h_l)P) \cdot (P\cdot)\vartheta_l \tag{4.6}$$

of the local error follows from (4.5).

If the DAE is linear with constant coefficients, it will be equivalent to system (2.4), (2.5). Considering ((2.4), (2.5)) a discretization error $\Theta_{I,l}$ which is asymptotically equal to $\alpha_{0,l}^{-1}P\vartheta_l$ occurs in the regular ODE (2.4) and is propagated in (2.5). The sum of $\Theta_{I,l}$ and its propagation is asymptotically equal to $(P - QG_1^{-1}BP)\alpha_{0,l}^{-1}\vartheta_l$. $\Pi_{can(1)} = P - QG_1^{-1}BP$ is just the projection along N onto S, the space filled with possible solution curves of the homogeneous equation.

If a nonlinear index-1 DAE has constant leading nullspace N, $\Pi_{can(1)}$ of any linearization in a point (y, x, t) with (x, t) in the restriction manifold $\mathcal{M}_1 \subseteq \mathbb{R}^n \times \mathcal{I}$ will be the projection along N onto the tangent subspace T of $\mathcal{M}_1$ in (x, t). Thus, the local error Θ_l is asymptotically the projection of $\alpha_{0,l}^{-1}\vartheta_l$ onto T. (Θ_l is a small perturbation of $\Pi_{can(1)}\alpha_{0,l}^{-1}\vartheta_l$.)

Representation of Θ_l in the index-2 case: $(\tilde{A}, \tilde{B})$ is an index-2 matrix pencil. Regarding (2.14) we express

$$\Theta_l = \left(\frac{-Q\tilde{Q}_1}{h_l} + \alpha_{0,l}^{-1}\tilde{\Pi}_{can(2)} + \tilde{\Pi}_{can(2)}O(h_l)P\tilde{P}_1\right) \cdot (P\cdot)\vartheta_l. \tag{4.7}$$

$\tilde{\Pi}_{can(2)}$, $\tilde{Q}_1$ and $\tilde{P}_1$ are small perturbations of the corresponding projectors from pointwise partial derivatives in the neighbourhood.

If the DAE is linear with constant coefficients, it will be equivalent to system (2.9), (2.10), (2.11), (2.12). Two discretization errors arise in that system. The discretization of the regular ODE (2.9) causes a local error Θ_I asymptotically equal to $\alpha_0^{-1}\vartheta_I = \alpha_0^{-1}\Pi_{can(2)}\vartheta$.

The discretization of $-QQ_1x'$ and its assignment in (2.11) cause another discretization error Θ_D. Θ_D is the difference between the derivative and the backward difference quotient mapped by $-QQ_1$:

$$\Theta_D = -QQ_1(P) \cdot h^{-1}\vartheta.$$

Θ_D is one order lower than Θ_I, but it is not propagated in the next steps. The overall local error Θ is the sum of Θ_I and Θ_D. Thus, the terms of (4.7) are explained in the linear constant coefficient case.

Remark: (4.4) provides the asymptotic conformance of Θ_l and $\alpha_{0,l}^{-1}\vartheta_l$ in the special case of **explicit ODEs** ($A \equiv \mathrm{Id}$):

$$(\mathrm{Id} + \alpha_{0,l}^{-1} h_l \tilde{B})^{-1} \cdot \alpha_{0,l}^{-1}\vartheta_l = \Theta_l \tag{4.8}$$

However, if the ODE is stiff, i. e., the magnitude of an eigenvalue of $\tilde{B}$ is large, the term $\mathrm{Id} + \alpha_{0,l}^{-1} h_l \tilde{B}$ cannot be considered to be a small perturbation of Id. Thus, the asymptotic conformance is useless for stiff equations.

5 Error estimates

Most implementations of the BDF use an estimate $\hat{\vartheta}_l$ of the truncation error ϑ_l in the local error test (see section 1). If we want to estimate and control the local error, we will have to rely on $\hat{\vartheta}_l$ because of relation (4.4), too.

Let $[x_l]_{l=1}^N$ be the computed approximation of the solution on the grid π. ϑ_l is defined as (see definition 4.2)

$$\vartheta_l := h_l x'_*(t_l) - \sum_{j=0}^{k} \alpha_{j,l} x_*(t_l). \tag{5.1}$$

As the true solution x_* is not known, the estimate $\hat{\vartheta}_l$ replaces it by an interpolation polynomial with the nodes (t_{l-j}, x_{l-j}). Since the backward difference quotient is exact for polynomials of degree less than $k+1$, the interpolation polynomial p_l^{k+1} of degree $k+1$ with the $k+2$ nodes $(t_{l-k-1}, x_{l-k-1}), \ldots, (t_l, x_l)$ should be used:

$$\hat{\vartheta}_l := (p_l^{k+1})'(t_l) - \sum_{j=0}^{k} \alpha_{j,l} x_{l-j}. \tag{5.2}$$

A skilful way to compute $\hat{\vartheta}_l$ is supplied by (4.3) in lemma 4.3 applied to p_l^{k+1}. The interpolation polynomial p_{l-1} with nodes $(t_{l-k-1}, x_{l-k-1}), \ldots, (t_{l-1}, x_{l-1})$ is often used as a predictor: Then $p_{l-1}(t_l)$ is the starting point of the newton iteration applied to the nonlinear system (1.2). If we denote the new interpolation polynomial by p_l, then $p_l(t_l) - p_{l-1}(t_l)$ will be the correction added up in the Newton iteration. The calculation of $\hat{\vartheta}_l$ by (4.3) is called "Milnes Device":

$$\hat{\vartheta}_l = \frac{h_l}{t_l - t_{l-k-1}} \cdot (p_l(t_l) - p_{l-1}(t_l)). \tag{5.3}$$

What about the reliability of $\hat{\vartheta}_l$?

Obviously there will be no relation $\hat{\vartheta}_l = \vartheta_l + O(h_l^{k+2})$ even if the BDF integration scheme is convergent for the given problem.

To get an idea about the reliability of $\hat{\vartheta}_l$ consider the relation between $\hat{\vartheta}_l$ and ϑ_l with exact x_{l-j} for $j \geq 1$. (To mark this assumption we will denote the estimate by $\hat{\vartheta}_l^1$.) With the notions introduced in section 4 x_l corresponds to x_l^1. The relation between $\hat{\vartheta}_l^1$ and ϑ_l can be expressed using Θ:

$$\hat{\vartheta}_l^1 = \vartheta_l + K_{2,l} \cdot \Theta_l + O(h_l^{k+2})$$

The term $O(h_l^{k+2})$ assumes that x_* is sufficiently smooth. Now the problem dependent factor between $\hat{\vartheta}_l^1$ and ϑ_l can be computed. We obtain

$$\hat{\vartheta}_l^1 = \left[\mathrm{Id} + K_{2,l} \cdot \left(\frac{\alpha_{0,l}}{h_l} \tilde{A} + \tilde{B} \right)^{-1} \frac{\tilde{A}}{h_l} \right] \cdot \vartheta_l + O(h_l^{k+2}) \tag{5.4}$$

using theorem 4.4. This factor is invertible at least for index 1 and 2, but it depends strongly on the conformance $x_{l-j} = x_*(t_{l-j})$ for $j \geq 1$. Thus, it should not be used in practical computations. The factor can be evaluated depending on the index of the DAE.

explicit ODE: Even for explicit ODEs it is obviously different from Id: $\hat{\vartheta}_l^1 = [\alpha_{0,l}^{k+1} \cdot (\alpha_{0,l}^k)^{-1} + O(h_l)] \cdot \vartheta_l$

index 1: $\hat{\vartheta}_l^1 = [(\mathrm{Id} - \tilde{\Pi}_{\mathrm{can}(1)}) + \alpha_{0,l}^{k+1} \cdot (\alpha_{0,l}^k)^{-1} \tilde{\Pi}_{\mathrm{can}(1)} + O(h_l)] \cdot \vartheta_l$
Hence $\hat{\vartheta}_l^1$ behaves similar to the explicit case.

index 2:

$$\hat{\vartheta}_l^1 = \left[(\mathrm{Id} - \tilde{\Pi}_{\mathrm{can}(2)}) + \frac{\alpha_{0,l}^{k+1}}{\alpha_{0,l}^k} \tilde{\Pi}_{\mathrm{can}(2)} + \right. \\ \left. + K_{2,l} \cdot \frac{-Q\tilde{Q}_1}{h_l} + O(h_l) \right] \cdot \vartheta_l$$

The components of $\hat{\vartheta}_l^1$ in $N \cap S = \mathrm{im}\, QQ_1$ are not of order $O(h_l^{k+1})$, but the differential components behave like $\hat{\vartheta}_l^1$ in the explicit case.

The notion $\alpha_{0,l}^k$ means $\alpha_{0,l}$ of the BDF of nominal order k.
Independently we show:

Lemma 5.1 Integrating a linear constant coefficient index-2 DAE with varying stepsize the part of $\hat{\vartheta}_l$ in $N \cap S$ will not have magnitude $O(h_l^{k+1})$ even if the right-hand side $q(t)$ is arbitrarily smooth.

ϑ_l itself has magnitude $O(h_l^{k+1})$ with sufficiently smooth $q(t)$ and solution x_* (see lemma 4.3 and (4.2)). The order assumption $O(h_l^{k+1})$ is necessary for the stepsize selection to work (see (1.4)).

PROOF: Replacing QQ_1x' by the backward difference quotient the equation (2.11) reads

$$Vx_l - (V\cdot)QQ_1 \sum_{j=0}^{k} \frac{\alpha_{j,l}}{h_l} x_{l-j} + (V\cdot)QQ_1x_l + VQP_1G_2^{-1}q(t_l) = 0. \tag{5.5}$$

$\hat{\vartheta}_l$ evaluates to

$$\hat{\vartheta}_l = \sum_{i=0}^{k+1} \alpha_{i,l}^{k+1} x_{l-i} - \sum_{i=0}^{k} \alpha_{i,l}^k x_{l-i}. \tag{5.6}$$

Denoting $q_{l-j} = q(t_{l-j})$, substituting $Q_1 x_{l-j} = -Q_1 G_2^{-1} q_{l-j}$ and using

$$\sum_{j=0}^{k} \frac{\alpha_{j,l-i}^{k}}{h_{l-i}} q(t_{l-i-j}) = q'(t_{l-i}) - K_{1,l-i} \cdot h_{l-i}^{k} q^{(k+1)}(t_{l-i}) + O(h_l^{k+1}),$$

the $N \cap S$ part of $\hat{\vartheta}_l$ is

$$\begin{aligned} V\hat{\vartheta}_l &= -QQ_1 G_2^{-1} \left(\left[\sum_{i=0}^{k+1} \alpha_{i,l}^{k+1} q'(t_{l-i}) - \sum_{i=0}^{k} \alpha_{i,l}^{k} q'(t_{l-i}) \right] - \right. \\ &\quad \left. - \left[\sum_{i=0}^{k+1} \alpha_{i,l}^{k+1} K_{1,l-i} h_{l-i}^{k} q_{l-i}^{(k+1)} - \sum_{i=0}^{k} \alpha_{i,l}^{k} K_{1,l-i} h_{l-i}^{k} q_{l-i}^{(k+1)} \right] \right) \\ &\quad + O(h_l^{k+1}) \\ &= -QQ_1 G_2^{-1} K_{1,l} h_l^{k+1} q^{(k+2)}(t_l) + O(h_l^{k+1}) + QQ_1 G_2^{-1} \cdot K_{1,l} h_l^{k} \cdot \\ &\quad \left[\sum_{i=0}^{k+1} \alpha_{i,l}^{k+1} \frac{K_{1,l-i} h_{l-i}^{k}}{K_{1,l} h_l^{k}} q_{l-i}^{(k+1)} - \sum_{i=0}^{k} \alpha_{i,l}^{k} \frac{K_{1,l-i} h_{l-i}^{k}}{K_{1,l} h_l^{k}} q_{l-i}^{(k+1)} \right]. \end{aligned} \tag{5.7}$$

If the stepsize is constant, $K_{1,l} \equiv K_1$ will be constant, and the term (5.7) will be the truncation error of the smooth function $q^{(k+1)}$. However, if the stepsize varies, the quotients

$$\gamma_{l,i} = \frac{K_{1,l-i} h_{l-i}^{k}}{K_{1,l} h_l^{k}}$$

will not converge to 1 for $h \to 0$. Thus, the term (5.7) does not have magnitude $O(h_l)$ and $V\hat{\vartheta}_l \not\sim O(h_l^{k+1})$. q. e. d.

Summing up the reliability discussion: Although we do not know any relation between $\hat{\vartheta}_l$ and ϑ_l, we have to rely on $\hat{\vartheta}_l$ in the index-1 case and on $P\hat{\vartheta}_l$ in the index-2 case.

A way often used to avoid problems with the components of $\hat{\vartheta}_l$ in $N \cap S$ is to exclude them from error control. If $N \cap S$ is not known, $P\hat{\vartheta}_l$ can be controlled alternatively. Control of $P\hat{\vartheta}_l$ seems to be even more skilful than excluding only $N \cap S$, because the algebraic components of ϑ_l do not contain any information about the local error (see theorem 4.4).

Estimates of the local error Theorem 4.4 suggests a simple way to estimate the local error Θ_l. The Newton iteration of the BDF method needs a Jacobian

$$\Phi := \frac{\alpha}{h} A + B$$

to be evaluated at some points and decomposed to LU factors. A is often known explicitly, e. g. if $f(y, x, t) = A(x, t) \cdot y + g(x, t)$.

Definition 5.2 Let $\xi = (P(t)y, x, t) \in \mathcal{G} \times \mathcal{I}$ be the last point at which the partial derivatives A and B have been evaluated and the iteration matrix Φ decomposed. Denote the leading BDF

coefficient by α and the stepsize used to form Φ by h. $\hat{\vartheta}_l$ is the estimate of the truncation error ϑ_l defined in (5.2). The estimate $\hat{\Theta}_l$ of the local error Θ_l is

$$\hat{\Theta}_l := \frac{\alpha}{\alpha_{0,l}} \Phi^{-1} \frac{A}{h} \hat{\vartheta}_l. \tag{5.8}$$

Denote $\xi^* = (P(t_l)x'_*(t_l), x_*(t_l), t_l)$ and assume that $\|\xi - \xi^*\| = O(h)$ and $O(h_l) = O(h)$. Using the representation of Θ_l in (4.6) and theorem 2.1 we can estimate in the index-1 case that

$$\begin{aligned} \hat{\Theta}_l &= (\alpha_{0,l}^{-1} \Pi_{can(1)}(\xi) + O(h)) \cdot \hat{\vartheta}_l \qquad (5.9) \\ \|\Theta_l - \hat{\Theta}_l\| &\le C_1 \cdot \left(\|\Pi_{can(1)}(\xi^*) - \Pi_{can(1)}(\xi)\| + O(h) \right) \cdot \|\vartheta_l\| + \\ &\quad + C_2 \cdot \|\hat{\vartheta}_l - \vartheta_l\|. \end{aligned}$$

If A and B are locally Lipschitz continuous, the projector $\Pi_{can(1)}$ associated to (A, B) will be so, too. Hence, the difference $\|\Pi_{can(1)}(\xi^*) - \Pi_{can(1)}(\xi)\|$ is of magnitude $O(h)$. $\hat{\Theta}_l$ is an estimate of Θ_l as reliable as $\hat{\vartheta}_l$ of ϑ_l and it is of order $O(h_l^{k+1})$ in the index-1 case.

$\hat{\Theta}_l$ is not suitable for error control in the index-2 case. Let (A, B) be an index-2 matrix pencil now. We denote its projectors by Q_1, P_1 and $\Pi_{can(2)}$. Taking (2.14) into account, $\hat{\Theta}_l$ has the representation

$$\hat{\Theta}_l = \left(\frac{-\alpha Q Q_1}{\alpha_{0,l} h} + \alpha_{0,l}^{-1} \Pi_{can(2)} + \Pi_{can(2)} O(h) P P_1 \right) \cdot (P\cdot)\hat{\vartheta}_l. \tag{5.10}$$

Thus, $\hat{\Theta}_l$ does not have magnitude $O(h_l^{k+1})$ in some components ($N \cap S$), although $P \cdot \hat{\vartheta}_l$ has. The local error Θ_l itself has a similar representation (4.7) and therefore it is one order too low. Moreover, Q_1 will be a $O(h)$ perturbation of $\tilde{Q}_1$ (as defined in (4.7)) if the problem is nonlinear. This perturbation is amplified by h^{-1} in (5.10). Consequently, we cannot estimate Θ_l using theorem 4.4. We cannot control Θ_l itself.

How to overcome these difficulties in the index-2 case?

Suggestion: Θ_l as well as $\hat{\Theta}_l$ inherit a part of the form $-h^{-1} Q Q_1$ multiplied by $P\vartheta_l$ or $P\hat{\vartheta}_l$, respectively. Q_1 is associated either to the pointwise linearization in ξ or to the mean value $(\tilde{A}, \tilde{B})$ (see section 4). This part has to be scaled by a factor of order $O(h)$ to obtain a test function of order $O(h^{k+1})$. After scaling we do not control the full local error anymore.

However, we feel more comfortable thinking of the geometric interpretation of the parts of Θ_l in the linear constant coefficient case in section 4:

The part $-h^{-1} Q Q_1 \vartheta_l$ represents the error of the inherent differentiation task. It is not propagated during the next steps. The error that occurred in the discretization of the inherent ODE is not affected by this scaling. (This interpretation is only true for linear constant coefficient DAEs of course.)

The estimate of the full local error should be controlled in the index-1 case.

The most convenient way to apply this scaling is the use of theorem 2.1. We do not need to

evaluate Q_1, Π_{can}, P or the index explicitly:

$$S_l := \tilde{\Phi}^{-1}\tilde{A} \cdot \left(\kappa \mathrm{Id} + \frac{\alpha_{0,l}}{h_l^2}\tilde{\Phi}^{-1}\tilde{A}\right)\vartheta_l \tag{5.11}$$

$$\hat{S}_l := \Phi^{-1}A \cdot \left(\kappa \mathrm{Id} + \frac{\alpha^2}{\alpha_{0,l}h^2}\Phi^{-1}A\right)\hat{\vartheta}_l. \tag{5.12}$$

The matrices marked by a tilde correspond to the pencil $(\tilde{A}, \tilde{B})$ as introduced after theorem 4.4. α is the BDF coefficient and h the stepsize used in the iteration matrix evaluated last and decomposed at time t. κ is a user-supplied factor to weigh the differentiation. (Scaling by h corresponds to the interval length 1. Thus, κ shall be "interval length"$^{-1}$ if it differs from 1 significantly.) Using theorem 2.1 and equation (2.14) S_l and $\hat{S}_l$ have the asymptotical representation

$$S_l = \left(-\kappa Q\tilde{Q}_1 + \alpha_{0,l}\tilde{\Pi}_{can(2)} + \tilde{\Pi}_{can(2)}O(h)P\right) \cdot (P\cdot)\vartheta_l \tag{5.13}$$

$$= \left(-\kappa QQ_1(\xi^*) + \alpha_{0,l}\Pi_{can(2)}(\xi^*) + O(h)P\right) \cdot (P\cdot)\vartheta_l \tag{5.14}$$

$$\hat{S}_l = \left(-\kappa QQ_1(\xi) + \alpha_{0,l}\Pi_{can(2)}(\xi) + \Pi_{can(2)}(\xi)O(h)P\right) \cdot (P\cdot)\hat{\vartheta}_l \tag{5.15}$$

for index 2 and

$$S_l = \left(\alpha_{0,l}\tilde{\Pi}_{can(1)} + \tilde{\Pi}_{can(1)}O(h)P(t_l)\right) \cdot (P(t_l)\cdot)\vartheta_l \tag{5.16}$$

$$= \left(\alpha_{0,l}\Pi_{can(1)}(\xi^*) + O(h)P(t_l)\right) \cdot (P(t_l)\cdot)\vartheta_l \tag{5.17}$$

$$\hat{S}_l = \left(\alpha_{0,l}\Pi_{can(1)}(\xi) + \Pi_{can(1)}(\xi)O(h)P(t)\right) \cdot (P(t)\cdot)\hat{\vartheta}_l \tag{5.18}$$

for index 1. Recall $\tilde{Q}_1$ and $\tilde{\Pi}_{can}$ to be small perturbations of their corresponding projectors evaluated in ξ^*. If the partial derivatives A and B are locally Lipschitz and $\xi - \xi^* = O(h)$, $\hat{S}_l$ will be a good estimate of S_l. S_l is asymptotically equal to the local error Θ_l for index 1 and to Θ_l with scaled part $-h_l^{-1}Q\tilde{Q}_1$ for index 2. Hence, $\hat{S}_l$ works according to the suggestion.

Remarks:

1. Even if the geometric interpretation of the origin of the parts of local error does not fit (due to nonlinearities or time-dependent coefficients), the estimate $\hat{S}_l$ will have order $O(h_l^{k+1})$ and fulfil assumption (1.4), which is expected by the stepsize selection algorithm: It uses only $P \cdot \hat{\vartheta}_l$, and the factor in front of $P \cdot \hat{\vartheta}_l$ is bounded for $h \to 0$ and index 1 and 2.

   $\hat{S}_l$ uses the subspace structure of the pointwise linearization in ξ for nonlinear problems.

2. The numerical computation **costs** of $\hat{S}_l$ are moderate:

   If we consider function calls of f and Jacobian evaluations and decompositions as *essential* costs of a numerical integration process, these essential costs will not increase. The main part of the computation of $\hat{S}_l$ are two back substitutions using the decomposed matrix Φ.

   However, the additional computational effort could be noticeable for problems of low dimension with a low-cost function f and a great number of steps. However, problems cannot become infeasible because of these additional costs.

The only additional information required from the user is A. A pure difference approximation would double the costs: Two matrices A and B would have to be computed by differences instead of Φ. But A is often known explicitly. In particular, we do not need to know the problem-dependent subspaces.

3. We should examine in numerical tests whether the behaviour of $\hat{S}_l$ depends on the strategy for a recomputation of Φ. Maybe $\hat{S}_l$ changes strongly after each new evaluation of Φ and A.

4. Using the notation of [BCP89] the modification of the usual control function $\hat{\vartheta}_l$ by a matrix factor is a "filter". The filter defined in 5.2 is described in [BCP89]. Accordingly, $\hat{S}_l$ is a "stronger filter" applicable to general DAEs of index 1 and 2.

5. **Order control** is an important part of the stepsize control. A variable order BDF has to choose the future order after each successful step. We denote $\hat{S}_l$ and $\hat{\vartheta}_l$ (defined above) by $\hat{S}_l^k$ and $\hat{\vartheta}_l^k$, respectively. Usual implementations introduce

$$\hat{\vartheta}_l^{k+1} := \sum_{i=0}^{k+2} \alpha_{i,l}^{k+2} x_{l-i} - \sum_{i=0}^{k+1} \alpha_{i,l}^{k+1} x_{l-i} \tag{5.19}$$

$$\hat{\vartheta}_l^{k-1} := \sum_{i=0}^{k} \alpha_{i,l}^{k} x_{l-i} - \sum_{i=0}^{k-1} \alpha_{i,l}^{k-1} x_{l-i} \tag{5.20}$$

for suitable k, equivalently to (5.6). (The practical formulas may differ from (5.19) and (5.20), but should be equivalent.) Then $\hat{\vartheta}_l^k$, $\hat{\vartheta}_l^{k+1}$ and $\hat{\vartheta}_l^{k-1}$ are compared with each other to choose the optimal order. Sometimes formula (1.3) with k, $k+1$ and $k-1$ is used to choose the order such that the next stepsize is as large as possible.
$\hat{\vartheta}_l^{k+1}$ and $\hat{\vartheta}_l^{k-1}$ can be modified to obtain $\hat{S}_l^{k+1}$ and $\hat{S}_l^{k-1}$ for comparison to $\hat{S}_l^k$ (as defined in (5.12)):

$$\hat{S}_l^{k+1} := \Phi^{-1}A \cdot \left(\kappa \mathrm{Id} + \frac{\alpha^2}{\alpha_{0,l}^{k+1} h^2} \Phi^{-1} A \right) \hat{\vartheta}_l^{k+1} \tag{5.21}$$

$$\hat{S}_l^{k+1} := \Phi^{-1}A \cdot \left(\kappa \mathrm{Id} + \frac{\alpha^2}{\alpha_{0,l}^{k-1} h^2} \Phi^{-1} A \right) \hat{\vartheta}_l^{k-1}. \tag{5.22}$$

Now $\hat{S}_l^k$, $\hat{S}_l^{k+1}$ and $\hat{S}_l^{k-1}$ can be used for choice of optimal order. (We recommend the usage of the "filtered" functions for order control, too. This is in contrast to [BCP89].)

6. This modification of the interpolation error and its estimate is not specific to the BDF. A similar modification seems to be possible using the trapezoidal rule as implemented in SPICE2 and treated in [Den88]. The only BDF specific theorem is theorem 4.4. A **general implicit linear multistep method** for DAEs with constant N looks like

$$0 = P \cdot \left(\sum_{j=0}^{k} \beta_{j,l} y_{l-j} - \frac{1}{h_l} \sum_{j=0}^{k} \alpha_{j,l} x_{l-j} \right) + Q \cdot y_l \tag{5.23}$$

$$0 = f(y_l, x_l, t_l). \tag{5.24}$$

with nonzero $\alpha_{0,l}$ and $\beta_{0,l}$. ($\alpha_{0,l} = 1$, $\alpha_{1,l} = -1$ and $\beta_{0,l} = \beta_{1,l} = \frac{1}{2}$ for the trapezoidal rule) (5.23) is replaced by

$$0 = \sum_{j=0}^{k} \beta_{j,l} y_{l-j} - \frac{1}{h_l} \sum_{j=0}^{k} \alpha_{j,l} x_{l-j} \tag{5.25}$$

in practical implementations and only the x_l part of the solution is controlled. The iteration matrix is

$$\Phi := \frac{\alpha_{0,l}}{\beta_{0,l} h_l} A + B, \tag{5.26}$$

the truncation error is

$$\vartheta_l := h_l \sum_{j=0}^{k} \frac{\beta_{j,l}}{\beta_{0,l}} x_*'(t_{l-j}) - \sum_{j=0}^{k} \frac{\alpha_{j,l}}{\beta_{0,l}} x_*(t_{l-j}) \tag{5.27}$$

and the local error is

$$\Theta_l := x_l^1 - x_*(t_l) \tag{5.28}$$

if (y_l^1, x_l^1) was the solution of (5.25), (5.24) with $x_{l-j} = x_*(t_{l-j})$ and $y_{l-j} = x_*'(t_{l-j})$ for $j \geq 1$. Using these denotations the generalization of theorem 4.4 has the form

$$\left(\frac{\alpha_{0,l}}{\beta_{0,l} h_l} \tilde{A} + \tilde{B} \right) \cdot \Theta_l = \frac{\tilde{A}}{h_l} \vartheta_l. \tag{5.29}$$

$\tilde{A}$ and $\tilde{B}$ are the mean values between (y_l^1, x_l^1, t_l) and $(x_*'(t_l), x_*(t_l), t_l)$. The only method specific estimate is the choice of $\hat{\vartheta}_l$. This depends on the nominal order of the method. See [Den88] for choices and formulas of $\hat{\vartheta}_l$ implemented in the trapezoidal rule of SPICE2.

However, only linear methods with $\beta_{j,l} = 0$ for $j \geq 2$ (i. e. the BDF methods) approximate the set of points (y, x, t) passed by solutions of $f(x', x, t) = 0$ in the index-2 case (see [Rhe84] for the geometric concept of the index):
The defect of the full set of constraints in (y_l, x_l, t_l) is of the magnitude $h_l^{-1} \sum_{j=0}^{k} \alpha_{j,l} \delta_{l-j} + O(h_l^k)$ (if $f(y_{l-j}, x_{l-j}, t_{l-j}) = \delta_{l-j}$ were the last defects of the BDF solutions) and depends only on the last k steps. ([HW91] implemented IRK methods like RADAU5 with similar properties for index-2 DAEs.)

6 Stepsize control with respect to the Newton iteration

Another task of the stepsize control is: How to choose the next stepsize such that we can expect convergence of the BDF inherent Newton iteration with moderate effort? Obviously, most of the performance parameters of the iteration are stepsize dependent.

Remind, the stricter the control the smaller the stepsize, and ,particularly, the stepsize control for index-2 DAEs cannot choose stepsize arbitrary small due to the instability of the BDF and the ill-conditioned iteration matrices.

The Newton iteration is only locally convergent. Thus, first of all we have to observe the presumable length of correction added up in the iteration process. Other performance parameters (contraction rate of defects or corrections) depend on this difference between the starting point and the solution of the iteration. The difference at the grid point t_l is denoted by $p_l(t_l) - p_{l-1}(t_l)$ in section 5. It coincides asymptotically with $K_{2,l}^{-1}\hat{\vartheta}_l$. The grid dependent factor $K_{2,l}$ was introduced in lemma 4.3 (see (5.3)).

If we denote the length of correction by $z_l := p_l(t_l) - p_{l-1}(t_l)$ and aim at $\|z_l\| \leq z_{\text{ref}}$ in the next step, h_l has to be

$$h_l \leq h_{l-1} \sqrt[k+1]{\frac{z_{\text{ref}}}{\|z_{l-1}\|}}. \tag{6.1}$$

The final choice of the new h_l has to be the minimum of the suggestions made by error control and by (6.1). Restriction (6.1) of h_l assumes z_l to be of order $O(h_l^{k+1})$. As mentioned in section 4, if the DAE has index 2, z_l will not have magnitude $O(h_l^{k+1})$. Thus, restriction (6.1) represents only a rough upper bound for h_l with respect to the correction length. More restrictive stepsize control strategies are not considered to be useful because of the ill-posedness of the index-2 DAEs, and since they are not in the scope of this paper.

7 Numerical tests

We have tested the error control with $\hat{S}_l$ with several examples arising in the simulation of constraint multibody systems and of electrical circuits and compared it with the usual alternatives. To do so we have included the computation of $\hat{S}_l$ and its control into the code `LSODI` from `ODEPACK` by A. C. Hindmarsh as an option (for experimental purposes only). Other slight changes of the code were:

1. Conversion from fixed to variable coefficient BDF.
2. Disabling the scaling of all equations by h_l and enabling the code to treat fully implicit equations.
3. Separation of the tolerances `tolnew` of the Newton iteration stopping criterion from the tolerances `tol` of error control.
4. Adding arguments (A and Q) to the subroutine. Q is only needed for a comparison to the control of $P\hat{\vartheta}$.
5. `LSODI` computes the iteration matrix Φ if the coefficient $\alpha_{0,l}h_l^{-1}$ changed considerably or after each `msbp` steps. Two optional (more expensive) updating strategies were included: Optionally the subroutine computes Φ once in each step or once in each Newton iteration.

Sources of some DAE problems were the test suite of "Centrum voor Wiskunde en Informatica" Amsterdam

`http://www.cwi.nl/cwi/projects/IVPtestset.shtml`

and [Tis96]. The reference solutions of index-2 examples suggested by that test suite were found out by analysing the problem and excluding the critical index-2 components from error control. A description in postscript format, FORTRAN sources and a reference solution at the end of the interval came with each problem of the test suite. We could compare the error control using $\hat{S}_l$ versus $\hat{\vartheta}_l$ and $P\hat{\vartheta}_l$ for index-1 problems and the control using $\hat{S}_l$ versus $P\hat{\vartheta}_l$ and $\hat{\vartheta}_{l,1}$ (that means: excluding index-2 components) for index-2 problems.

Rough summary of those experiments:

index 1: The error control using $\hat{S}_l$ was considerably better than $\hat{\vartheta}_l$ applied to examples which were not treated well by $\hat{\vartheta}_l$. (See example 7.1 for possible explanations for difficulties of the $\hat{\vartheta}_l$ control.)

index 2: Control of $P\hat{\vartheta}_l$ will not be satisfying if there are very few differential components or if they are of less interest. $\hat{S}_l$ promises to treat those problems better (e. g. NAND gate of [Tis96]). See example 7.2 for another difficulty, possibly arising in the case of $P\hat{\vartheta}_l$ control.

We want to present in this paper small specially constructed examples for index 1 and for index 2.

7.1 Illustration of the difference between the truncation error and the local error in the Index-1 case

It is characteristic of general index-1 equations to inherit an implicit assignment to the algebraic components Qx depending on the differential components Px:

$$Qx = \zeta_{\text{implicit}}(Px, t).$$

The example presented below is a decoupled system of index 1. It is constructed to visualize the behaviour of the step size control based on $\hat{\vartheta}_l$, $P\hat{\vartheta}_l$ and $\hat{S}_l$ depending on the magnitude of $\frac{\partial \zeta}{\partial Px}$. The parameter c in (7.3) represents this partial derivative.

$$\begin{aligned} 0 &= x_1' - x_2 && (7.1)\\ 0 &= x_2' + x_1 && (7.2)\\ 0 &= \exp(x_3 - c \cdot [x_1 - \sin(t)] - \sin(t)) - 1 && (7.3) \end{aligned}$$

The equations ((7.1), (7.2)) constitute an explicit ODE for x_1 and x_2 with a solution $(\sin(t), \cos(t))$. The algebraic component x_3 is determined by $c \cdot x_1$ (and adjusted to give an exact solution $\sin(t)$). We write $\exp(\dots) - 1 = 0$ instead of $\dots = 0$, so the BDF equation is not solved exactly by the Newton iteration. The global error of x_3 is c times global error of x_1. We observe the behaviour of the error control of $\hat{\vartheta}_l$, $P\hat{\vartheta}_l$ (i. e. the x_1 and x_2 part of $\hat{\vartheta}_l$) and $\hat{S}_l$ with respect to varying c in the figure 1. The choice of `tolnew` is `tol`.

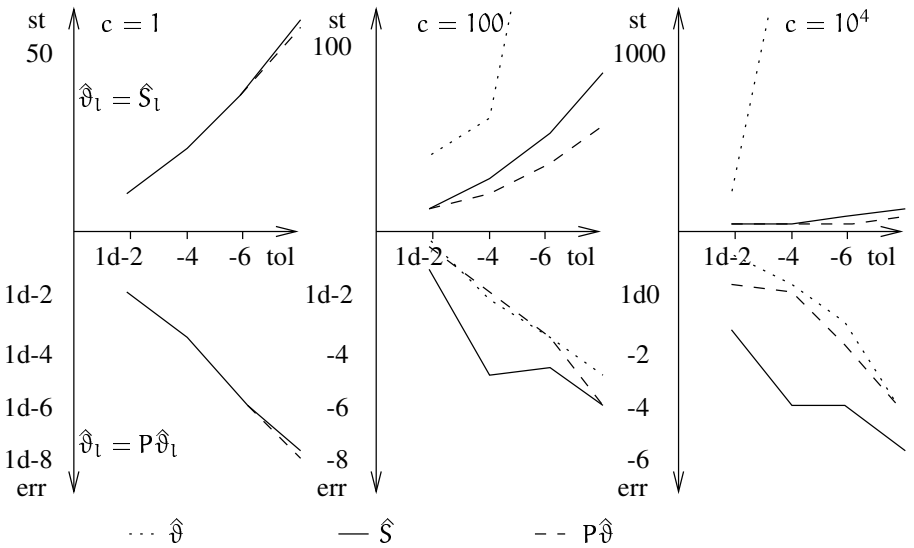

Figure 1: global error of x_3 vs. tolerance and number of steps plotted over the range of tolerances `1d-2` to `1d-8`

The control of $\hat{\vartheta}_l$ does not work well with $c \gg 1$. Perturbations of the interpolation polynomial of x_3 used by $\hat{\vartheta}_l$ are amplified by c. This leads to a lack of smoothness of x_3 and so to bad performance of the solver. It chooses order 1 mostly.

$P\hat{\vartheta}_l$ does not take care of c. So the number of steps is either very large ($\hat{\vartheta}_l$) or the error is c times larger than the user expected ($P\hat{\vartheta}_l$). The control of $\hat{S}_l$ manages x_1 to be as accurate as necessary to get the desired accuracy of x_3.

7.2 Illustration of the difference between the truncation error and the local error in the Index-2 case

Example 7.1 is extended to an index-2 example by joining a differentiation of x_3 as equation (7.6).

$$0 = x_1' - x_2 \tag{7.4}$$
$$0 = x_2' + x_1 \tag{7.5}$$
$$0 = x_3' + x_4 \tag{7.6}$$
$$0 = \exp(x_3 - c \cdot [x_1 - \sin(t)] - \sin(t)) - 1 \tag{7.7}$$

This is a Hessenberg system of index 2. x_4 is the index-2 variable (in $N \cap S = N$) and a solution is $x_4 = -\cos(t)$. The performance of the error control of $P\hat{\vartheta}_l$ and of $\hat{S}_l$ is compared. We report the number of steps and the global error of x_3 and x_4 using $c = 1, 100$ and 10^4 in figure 2 and figure 3. `tolnew` was chosen `tol`.

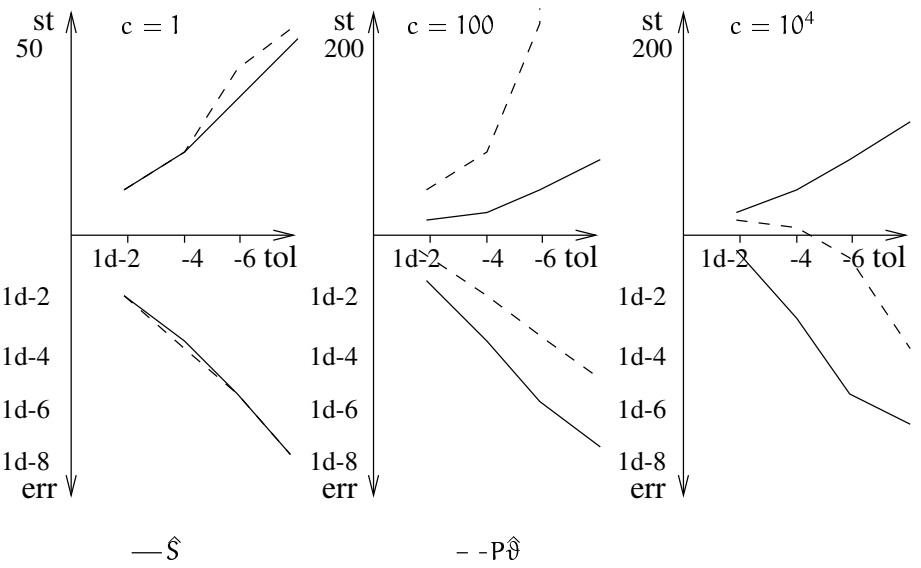

Figure 2: global error of x_3 and number of steps vs. tolerance plotted over the range of tolerances `1d-2` to `1d-8`

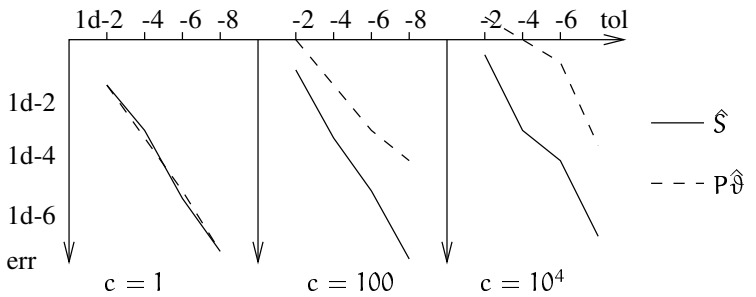

Figure 3: global error of x_4 vs. tolerance in (number of steps: see figure 2 and text) plotted over the range of tolerances `1d-2` to `1d-8`

Now the control of $P\hat{\vartheta}_l$ faces the same difficulties as the $\hat{\vartheta}_l$ control did in example 7.1. Its numbers of steps for $c = 10^4$ are 407, 5,200, 44,000 and 8,800 for the tolerances 1d-2, 1d-4, 1d-6 and 1d-8 (do not fit into the graphics of figure 2). Thus, the control of $P\hat{\vartheta}_l$ leads under certain circumstances to an unreasonable inefficient behaviour of the solver.

7.3 A Hessenberg system of index 2

Now we want to give some idea about the role of κ. We use the Hessenberg system of index 2 from [Fre95, Mä96] on the interval $[0.1, 1.5]$. (It is of interest to compare the results obtained below with [Fre95, Mä96].)

$$
\begin{aligned}
0 &= x_1' + x_5 - x_4 && (7.8)\\
0 &= x_2' + 2 \cdot \sqrt{x_4 \cdot x_5} && (7.9)\\
0 &= \sin(t) \cdot x_3' - 5 \cdot \sin(t) && (7.10)\\
0 &= 25 \sin(\arcsin^3(x_1)) - 75 \sin(\frac{1}{375} x_3^3) + 100 \sin^3(\frac{1}{3} t^3) && (7.11)\\
0 &= 2 \cdot x_1 \cdot x_2 - \sin(\frac{2}{5} \cdot x_3), && (7.12)
\end{aligned}
$$

which has a solution $x_* = (\sin(t), \cos(t), 5t, \cos^2(\frac{1}{2}t), \sin^2(\frac{1}{2}t))$. The dimension of the inherent regular ODE is 1 and (7.10) is a regular ODE that is solved exactly by the BDF. Thus, the discretization error contains only a $N \cap S = N$ part. The global error of the P components x_1, x_2 and x_3 measured in practical computations is less than or equal to the user given tolerance `tol`. It is caused only by numerical perturbations (the inaccurate solution of the BDF equation with `tolnew=tol` and roundoff errors). The graphics do not show this error.

We focus on the error of the index-2 variables, i. e. of the Q components x_4 and x_5. It is influenced only by the discretization error occurring in the inherent differentiation task and the numerical perturbations. Therefore increasing weights κ are expected to decrease the errors of x_4 and x_5, accordingly.

Firstly, we show the results obtained using $\hat{S}_l$ with $\kappa = 1d0$ (see (5.12)) and $P\hat{\vartheta}_l$ with various tolerances (`rtol=atol`) in the diagrams of figure 4.

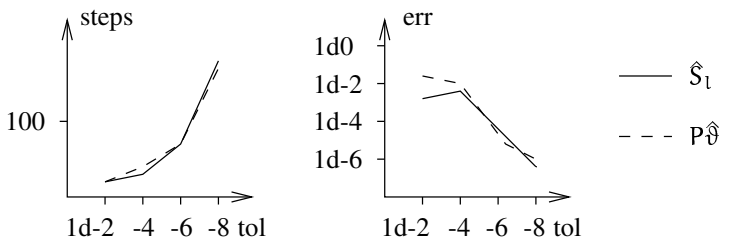

Figure 4: number of steps (left) and maximum norm of Q· global error at $t = 1.5$ for the system (7.8) – (7.12) plotted over the range of tolerances `1d-2` to `1d-8`

The difference between the results obtained with control of $\hat{S}_l$ (with $\kappa = 1$) and $P\hat{\vartheta}_l$ is immaterial. However, the comparison with [Mä96, Fre95] illustrates a considerably better performance of the standard variable order BDF codes used here. We emphasize again, that the error control using $\hat{\vartheta}_l$ does not work well. (Particularly, the code fails for $\texttt{tol} \leq 1d-3$ and chooses low orders for rough tolerances.)

We fix `tol=tolnew=1d-5` and vary κ between 0 and `1d7` to obtain figure 5. This diagram should illustrate the role of κ as a "weight of the error occuring in the inherent differentiation" (see (5.15)). Since the discretization is exact for the components x_1, x_2 and x_3 (unless the numerical perturbations), the main part (in orders of h) of S_l contains only the term $-\kappa Q Q_1 \cdot \vartheta_l$. Thus, κ weighs directly the accuracy of x_4 and x_5. On the other hand, the accuracy of x_4 and x_5 is limited by the precision of the solution of the nonlinear system. The perturbations (`tolnew=1d-5`) are amplified by h_l^{-1}. Therefore, it turns out to be inefficient to improve the accuracy of the index-2 components by weighing with $\kappa \gg$ "interval length"$^{-1}$.

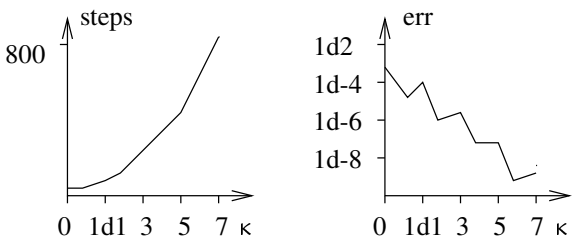

Figure 5: Global error of x_4 and x_5 over κ

7.4 Example of index 1 with moving geometry

This example (introduced by J. Wensch (Halle)) is intended to show the limitations of the numerical methods (like the BDF) and the pointwise linearizations applied to nonlinear or time dependent linear DAEs. The formulas and its convergence statements and estimates assume a negligible change of the geometry per step. The example system reads

$$\begin{pmatrix} \delta - 1 & \delta t \\ 0 & 0 \end{pmatrix} \cdot x' + \mu \cdot \begin{pmatrix} \delta - 1 & \delta t \\ \delta - 1 & \delta t - 1 \end{pmatrix} \cdot x = 0. \tag{7.13}$$

A solution of this linear time dependent system is

$$x(t) = \begin{pmatrix} (\delta - 1)^{-1} \delta t e^{(\delta - \mu)t} \\ e^{(\delta - \mu)t} \end{pmatrix}$$

Its leading Nullspace is time dependent:

$$N = \mathcal{L} \begin{pmatrix} \frac{\delta t}{\delta - 1} \\ 1 \end{pmatrix}.$$

The finite spectrum of the pointwise matrix pencil is

$$\sigma(A(t), B(t)) = \{-\mu\}.$$

Thus, the inherent ODE of the pointwise linearization is stiff for $\mu \gg 0$. The magnitude of the solution decreases exponentially for $\delta < \mu$ with the rate $\delta - \mu$. However, the solution of the

Implicit Euler Method (BDF(1)) with stepsize h decreases iff

$$\left|\frac{1+h\delta}{1+h\mu}\right| < 1,$$

i. e. the region of A-stability does not coincide with the region of stable solutions of the DAE. (It is bounded by the line $\{1+h\mu = -(1+h\delta)\}$ in the $(h\delta, h\mu)$ plane.) Consequently, the numerical method cannot work efficiently for $\delta < \mu$ with $|\delta| \gg |\mu|$. The stepsize control has to choose the stepsize $h <\sim |\delta/\mu|$. Some results obtained by the code described above are:

	δ	-1,000	-100,000	-10,000,000
$\hat{\vartheta}_1$	# steps	278	$\approx$ 24,000	$\approx$ 2,400,000
	# rejected steps	77	$\approx$ 7,300	$\approx$ 730,000
$\hat{S}_1$	# steps	344	$\approx$ 29,600	$\approx$ 2,900,000
	# rejected steps	94	$\approx$ 9,800	985,476

Table 1: Number of steps with $\mu = 1$ and `atol=rtol=1d-4` for Newton iteration and error test

The order chosen by the order control was 1 (Implicit Euler Method).

References

[BCP89] K. E. Brenan, S. L. Campbell, L. R. Petzold: *Numerical solution of initial-value problems in differential-algebraic equations*. North-Holland, New York, 1989

[Den88] G. Denk: *Die numerische Integration von Algebro-Differentialgleichungen bei der Simulation elektrischer Schaltkreise mit SPICE2*. Mathematisches Institut TU München, Rep. TUM-M8903, 1989

[Fre95] St. Freude: *Projizierende Defektkorrektur für Algebro-Differentialgleichungen mit dem Index 2*. Master Thesis, HU Berlin 1995

[GM86] E. Griepentrog, R. März: *Differential-algebraic equations and their numerical treatment*. Teubner Texte zur Mathematik 88, Leipzig, 1986

[HNW87] E. Hairer, S. P. Nørsett, G. Wanner: *Solving Ordinary Differential Equations I: Non-stiff Problems*. Springer Series in Computational Mathematics, 1987

[HW91] E. Hairer, G. Wanner: *Solving Ordinary Differential Equations II: Stiff and differential-algebraic Systems*. Springer Series in Computational Mathematics, 1991

[Hin83] A. C. Hindmarsh: *ODEPACK, a systematized collection of ODE solvers*. In Scientific Computing, R. S. Stepleman et al. (eds.), North-Holland, Amsterdam, 1983

[Mä92] R. März: *Numerical Methods for differential algebraic equations*. Acta Numerica. 1992, p. 141-198

[Mä95] R. März: *On linear differential-algebraic equations and linearizations*. Applied Numerical Mathematics 18, 1995, 267-292

[Mä96] R. März: *Managing the drift-off in numerical index-2 differential algebraic equations by projected defect corrections*. Preprint 96-32, HU Berlin 1996

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

[Sie97] J. Sieber: *Fehlerkontrolle und Schrittweitensteuerung bei der numerischen Integration von Anfangswertaufgaben mit der BDF*. Master Thesis, HU Berlin 1997

[Tis96] C. Tischendorf: *Solution of index-2 differential-algebraic equations and its application in circuit simulation*. PhD Thesis, HU Berlin 1996