On the structure of errors for Radau IA methods applied to index-2 DAEs

A. Aubry 1, P. Chartier *

Institut National de Recherche en Informatique et Automatique, 35042 Rennes Cedex, France

Abstract

In this contribution, we focus on the structure of local and global errors for Radau IA methods applied to index-2 problems. We show in particular that if the last step of integration is done with the Radau IIA method, the final approximation is then of order $2s - 2$. This result is also interpreted in terms of effective order and possible implications for Gauss methods are sketched.

Keywords: Differential–algebraic systems; Index-2; Radau IA methods; Radau IIA methods

1. Introduction

Differential–Algebraic Equations (DAEs) originate in the modelisation of various physical or chemical phenomena, most commonly in index one, two or three forms, and occasionally higher. Since the beginning of the eighties, the interest of working directly with the orginal system instead of reformulating it as an ordinary differential system has been acknowledged and has motivated many researches, from the pioneering work of S.L. Campbell [7,8] to the more recent analysis of index-3 semi-explicit systems conducted by L. Jay [17]. Many aspects of the numerical solution of differential–algebraic equations have been explored successfully. The first methods to be considered for solving differential–algebraic systems have been Backward Differentiation Formulae (BDFs), for which various convergence results—see [23] for linear systems or [3,11,12,19,20] for nonlinear systems—have been proven, leading to the development of the now famous code DASSL [2,21]. On the other hand, Runge–Kutta methods have been first regarded as poor competitors to multistep methods, mainly because of the order reduction phenomenon, observable for most non-stiffly accurate methods and for most DAEs. Ulterior results from [15,22] have however shown that good Runge–Kutta methods can form the basis of a competitive code, as demonstrated by the code RADAU5 from [15]. Moreover, in the context of constrained Hamiltonian systems, carefully chosen pairs of Runge–Kutta methods [17]
have proven to be capable of preserving invariants and hence to tackle difficult problems. An even more elaborate combination of Lobatto methods seems to enable a specific and thus more appropriate treatment of the different forces—component encountered in some mechanical systems [18].

In this paper, we will restrict our attention to Runge–Kutta methods applied to index-2 problems in semi-explicit form, as studied in [13,15]:

\[
\begin{align*}
\dot{y}(x) &= f(y(x), z(x)), \quad y(x_0) = y_0, \\
0 &= g(y(x)), \quad z(x_0) = z_0,
\end{align*}
\]

where \( f \) is a function from \( \mathbb{R}^M \times \mathbb{R}^N \) into \( \mathbb{R}^M \) and \( g \) a function from \( \mathbb{R}^M \) into \( \mathbb{R}^N \). For the system to have a unique solution, we assume that \( g(y_0) = 0 \), \( (\partial g/\partial y)f(y_0, z_0) = 0 \) and that

\[
(g_y f_z)(y, z) = \begin{pmatrix} \partial g/\partial y \\ \partial f/\partial z \end{pmatrix}(y, z)
\]

is invertible in a neighbourhood of the exact solution \((y(x), z(x))\). Convergence results for these systems are well-known, both for Runge–Kutta and projected Runge–Kutta methods. Projected Runge–Kutta methods have been introduced by U. Ascher and L.R. Petzold [1] as a mean to restore the usual order of convergence of Runge–Kutta methods. The underlying idea consists in projecting the numerical solution obtained after each step onto the invariant manifold \( \{ y \in \mathbb{R}^N, g(y) = 0 \} \). In the aforementioned paper, the projection step is clearly intended to be applied after each step of the numerical method. At least from a theoretical point of view and for certain Runge–Kutta methods, it seems that this additional projection does not need to be imposed at every step and can actually even be postponed to the end of the integration interval without dramatically affecting the convergence. Similarly to [4,9], the last step can also be replaced by one smoothing step of Radau IIA, giving an overall order of convergence of \( 2s - 2 \) for the \( s \)-stage Radau IIA method.

In Section 2 we will prove a general result on the influence of perturbation in the nonlinear system that has to be solved at each step. More precisely, an \( h \)-expansion of the output value in terms of the input perturbation is given. This expansion is examined in more details in Section 3 and it turns out that most terms can be eliminated from it, provided the input perturbation lies in the invariant space of a certain projection. This somehow surprising result will be confirmed by numerical results and should be compared with similar results obtained by L. Jay [16] in the more complex situation of index-3 problems. We will then draw from this local result some conclusions on the structure of the global error, confirmed by some numerical results in Section 4. Eventually, a different point of view will be briefly described in Section 5, allowing some extrapolations to different Runge–Kutta methods such as Gauss methods.

2. Influence of perturbations

In this section, we construct an expansion of the output value of a perturbed system in terms of the input perturbation. The result given here is a more precise version of [13, Theorem 4.2].

**Definition 1.** The sets of trees \( \text{SDAT}^2_y \) and \( \text{SDAT}^2_z \) are defined recursively by:

1. \( \Theta_y \in \text{SDAT}^2_y \),
2. \( [	au_x, \ldots, \tau_x, \omega]_y \in \text{SDAT}^2_y \) iff \( \omega \in \text{SDAT}^2_y \cup \text{SDAT}^2_z \),
(3) \([\tau_{x_1, \ldots, \tau_x, t}]_z \in \text{SDAT}_2\) if \(t \in \text{SDAT}_2\) and either \(m > 0\) or \(m = 0\) and \(t \notin \[u]\_y\) with \(u \in \text{SDAT}_2\).

By convention, we take \(\tau_y = \ast = \[\emptyset\]_y\) and \(\tau_z = \circ = \[\emptyset\]_z\).

Using standard notations [14,15], a \(y\)-root will be represented by a meagre vertex \(\ast\) and a \(z\)-root by a fat vertex \(\circ\). The new type of vertex \(\tau_z\) will be represented by an arrow. For sake of clarity, we will denote by \(t\), \(u\) and \(\omega\) trees of \(\text{SDAT}_2\), \(\text{SDAT}_2\) and \(\text{SDAT}_2 \cup \text{SDAT}_2\), respectively. A few trees are represented in Table 2.

**Definition 2.** Let \(\rho_0\) be an integer greater or equal to 1. The function \(\rho\) is defined recursively on \(\text{SDAT}_2 \cup \text{SDAT}_2\) by:

1. \(\rho(\emptyset_y) = \rho_0\),
2. \(\rho([\tau_{x_1, \ldots, \tau_x}, \omega]_y) = m + 1 + \rho(\omega)\),
3. \(\rho([\tau_{x_1, \ldots, \tau_x}, t]_z) = m - 1 + \rho(t)\).

\(\rho(\omega)\) is called the *order* of \(\omega\).

\(\rho(\emptyset_y)\) is taken equal to \(\rho_0\), where \(\rho_0\) will later be the order of \(\Delta y_0\). As a matter of fact, the elementary differentials associated with the trees of \(\text{SDAT}_2 \cup \text{SDAT}_2\) are eventually applied to \(\Delta y_0\).

**Definition 3.** The function \(\alpha\) is defined recursively on \(\text{SDAT}_2 \cup \text{SDAT}_2\) by:

1. \(\alpha(\emptyset_y) = 1\),
2. \(\alpha([\tau_{x_1, \ldots, \tau_x}, \omega]_y) = \left(\frac{\rho(\omega) + m}{\rho(\omega)}\right)\alpha(\omega)\),
3. \(\alpha([\tau_{x_1, \ldots, \tau_x}, t]_z) = \left(\frac{\rho(t) + m}{\rho(t)}\right)\alpha(t)\).

\(\alpha(\omega)\) is called the *number of labellings* of \(\omega\).

**Definition 4.** The function \(\gamma\) is defined recursively on \(\text{SDAT}_2 \cup \text{SDAT}_2\) by:

1. \(\gamma(\emptyset_y) = \rho_0!\),
2. \(\gamma([\tau_{x_1, \ldots, \tau_x}, \omega]_y) = \left(\frac{\rho(\omega) + m + 1}{\rho(\omega)}\right)\gamma(\omega)\),
3. \(\gamma([\tau_{x_1, \ldots, \tau_x}, t]_z) = \frac{1}{\left(\frac{\rho(t) + m}{\rho(t)}\right)\gamma(t)}\).

**Definition 5.** The function \(\Phi\) is defined recursively on \(\text{SDAT}_2 \cup \text{SDAT}_2\) by:

1. \(\Phi(\emptyset_y) = A^{-1}\),
2. \(\Phi([\tau_{x_1, \ldots, \tau_x}, \omega]_y) = C^m A\Phi(t)\),
3. \(\Phi([\tau_{x_1, \ldots, \tau_x}, u]_y) = C^m \Phi(u)\),
Table 1
Examples of trees with their associated functions

<table>
<thead>
<tr>
<th>ω</th>
<th>ρ(ω)</th>
<th>α(ω)</th>
<th>γ(ω)</th>
<th>Φ(ω)</th>
<th>F(ω)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ω0</td>
<td>ρ0</td>
<td>1</td>
<td>ρ0!</td>
<td>A⁻¹</td>
<td>I_M</td>
</tr>
<tr>
<td>ω0 + 1</td>
<td>(ρ0 + 1)!</td>
<td>1</td>
<td>A⁻¹</td>
<td>f_x(−g_y f_x)⁻¹ g_y</td>
<td></td>
</tr>
<tr>
<td>ω0 + 2</td>
<td>(ρ0 + 2)!</td>
<td>1</td>
<td>A</td>
<td>f_y f_y</td>
<td></td>
</tr>
<tr>
<td>ω0 + 1</td>
<td>(ρ0 + 1)!</td>
<td>1</td>
<td>A</td>
<td>f_x(−g_y f_x)⁻¹ g_y f_y</td>
<td></td>
</tr>
<tr>
<td>ω0 + 2</td>
<td>(ρ0 + 1)!</td>
<td>1</td>
<td>A⁻¹ C A</td>
<td>f_x(−g_y f_x)⁻¹ g_y (1) f_y</td>
<td></td>
</tr>
<tr>
<td>ω0 + 2</td>
<td>(ρ0 + 1)!</td>
<td>1</td>
<td>A⁻¹ C A</td>
<td>(−g_y f_x)⁻¹ g_y (1) f_y</td>
<td></td>
</tr>
<tr>
<td>ω0 + 2</td>
<td>(ρ0 + 1)!</td>
<td>1</td>
<td>A⁻¹ C A</td>
<td>(−g_y f_x)⁻¹ g_y (1) f_y</td>
<td></td>
</tr>
<tr>
<td>ω0 + 1</td>
<td>(ρ0 + 1)!</td>
<td>1</td>
<td>A⁻¹ C A</td>
<td>(−g_y f_x)⁻¹ g_y (1) f_x(−g_y f_x)⁻¹ g_y</td>
<td></td>
</tr>
</tbody>
</table>

(4) \( \Phi \left( [τ_x, \ldots, τ_x, t]_s \right) = A^{-1} C^m A \Phi(t), \)

with \( C = \text{diag}(c_1, \ldots, c_s) \). \( \Phi \) is called the elementary weight function.

Definition 6. The function \( F \) is defined recursively on SDAT2_y ∪ SDAT2_z by:

1. \( F(\emptyset) = I_M \),
2. \( F(\tau_x, \ldots, \tau_x, t)_y = \frac{\partial^m f_y(y(x), z(x))}{\partial x^m} \bigg|_{x_0} F(t) \),
3. \( F(\tau_x, \ldots, \tau_x, u)_y = \frac{\partial^m f_z(y(x), z(x))}{\partial x^m} \bigg|_{x_0} F(u) \),
4. \( F(\tau_x, \ldots, \tau_x, t)_z = (−g_y f_x)⁻¹ (y(x_0), z(x_0)) \frac{\partial^m g_y(y(x), z(x))}{\partial x^m} \bigg|_{x_0} F(t). \)

\( F(ω) \) is called the elementary differential associated with \( ω \).

We can now formulate the main result of this section:

Lemma 1. For an invertible \( s \times s \) matrix \( A \), let \( Y_i, Z_i \) be given by

\[
Y_i = y_0 + h \sum_{j=1}^{s} a_{ij} f(Y_j, Z_j), \quad i = 1, \ldots, s,
\]
\[
0 = g(Y_i), \quad i = 1, \ldots, s,
\]
\[
y_1 = y_0 + h \sum_{j=1}^{s} b_{ij} f(Y_j, Z_j),
\]
and consider perturbed values \( \hat{Y}_i, \hat{Z}_i \) satisfying
\[
\hat{Y}_i = \hat{y}_0 + h \sum_{j=1}^{s} a_{ij} f(\hat{Y}_j, \hat{Z}_j), \quad i = 1, \ldots, s,
\]
\[
0 = g(\hat{Y}_i), \quad i = 1, \ldots, s,
\]
\[
\hat{y}_1 = \hat{y}_0 + h \sum_{j=1}^{s} b_{ij} f(\hat{Y}_j, \hat{Z}_j).
\]

If the Runge–Kutta method \((A, b, c)\) satisfies \(C(q), q \geq 1\), and if
\[
y_0 = y(x_0) + O(h^n),
\]
\[
\hat{y}_0 = y(x_0) + O(h^n), \quad \eta \geq 2,
\]
where \((y(x), z(x))\) denotes the exact solution at abscissa \(x\), then we have
\[
(h\Delta F) = \sum_{t \in SDA_{2T}, \rho(t) \leq \mu+\eta, t \neq \rho_0} \frac{h^{\rho(t)-\eta}}{\rho(t)!} \alpha(t) \gamma(t) (\Phi(t)e \otimes F(t)\Delta y_0) + O(h^{\mu+1}\|\Delta y_0\|),
\]
\[
\Delta Z = \sum_{u \in SDA_{2T}, \rho(u) \leq \mu+\eta} \frac{h^{\rho(u)-\eta}}{\rho(u)!} \alpha(u) \gamma(u) (\Phi(u)e \otimes F(u)\Delta y_0) + O(h^{\mu}\|\Delta y_0\|),
\]
\[
\Delta y_1 = \sum_{t \in SDA_{2T}, \rho(t) \leq \mu+\eta} \frac{h^{\rho(t)-\eta}}{\rho(t)!} \alpha(t) \gamma(t) (b^T\Phi(t)e) F(t)\Delta y_0 + O(h^{\mu+1}\|\Delta y_0\|),
\]
where \(\rho_0 = \eta, e = (1, \ldots, 1)^T \in \mathbb{R}^s,\Delta y_0 = \hat{y}_0 - y_0,\Delta y_1 = \hat{y}_1 - y_1,\Delta F\) denotes the vector of \(\mathbb{R}^{sM}\) made up from the subvectors \(f(\hat{Y}_1, \hat{Z}_1) - f(Y_1, Z_1), \ldots, f(\hat{Y}_s, \hat{Z}_s) - f(Y_s, Z_s)\), \(\Delta Z\) denotes the vector of \(\mathbb{R}^{sN}\) made up from the subvectors \(\hat{Z}_1 - Z_1, \ldots, \hat{Z}_s - Z_s\) and where \(\mu = \min(\eta - 2, q - 1)\).

The techniques used in this proof are similar to [16,17]. However, our presentation differs slightly.

**Proof.** Denoting \(\Delta Y\) the vector of \(\mathbb{R}^{sM}\) made up from the subvectors \(\hat{Y}_1 - Y_1, \ldots, \hat{Y}_s - Y_s\) and considering \(f_y(y(x_0 + c_ih), z(x_0 + c_ih)), f_z(y(x_0 + c_ih), z(x_0 + c_ih))\) and \(g_y(y(x_0 + c_ih))\) as functions of \(x = x_0 + c_ih\) which we can expand at \(x_0\), we easily obtain
\[
\Delta Y = (e \otimes \Delta y_0) + (A \otimes I_M)(h\Delta F),
\]
\[
(h\Delta F) = \sum_{k=0}^{\mu} \frac{h^{k+1}}{k!} (C^k \otimes f_y^{(k)})(\Delta Y) + \sum_{k=0}^{\mu} \frac{h^{k+1}}{k!} (C^k \otimes f_z^{(k)})(\Delta Z) + O(h^{\mu+1}\|\Delta y_0\|),
\]
\[
0 = \sum_{k=0}^{\mu} \frac{h^k}{k!} (C^k \otimes g_y^{(k)})(\Delta Y) + O(h^{\mu+1}\|\Delta y_0\|).
\]
We look for an expansion of \(h\Delta F\) and \(\Delta Z\) of the form
\[
h\Delta F = \sum_{k=0}^{\mu} \frac{h^k}{(k+\eta)!} \phi^k(e \otimes \Delta y_0) + O(h^{\mu+1}\|\Delta y_0\|),
\]
(9)
\[
\Delta Z = \sum_{k=-1}^{\mu-1} \frac{h^k}{(k+\eta)!} \psi^k(e \otimes \Delta y_0) + O(h^\mu \|\Delta y_0\|). \tag{10}
\]

Inserting (6) into (7) gives
\[
(h\Delta F) = \sum_{k=0}^{\mu} \frac{h^{k+1}}{k!} (C^k \otimes f_y^{(k)}) (e \otimes \Delta y_0) + \sum_{k=0}^{\mu} \frac{h^{k+1}}{k!} (C^k A \otimes f_y^{(k)}) (h\Delta F)
\]
\[
+ \sum_{k=0}^{\mu} \frac{h^{k+1}}{k!} (C^k \otimes f_z^{(k)}) \Delta Z + O(h^{\mu+1} \|\Delta y_0\|),
\]

and hence
\[
\phi^0 = \eta(I_\delta \otimes f_z)\psi^{-1}, \tag{11}
\]
\[
\phi^j = \sum_{k=0}^{j} (j + \eta) \binom{j - 1 + \eta}{k} (C^k \otimes f_z^{(k)}) \psi^{j-k-1} + \frac{(j + \eta)!}{(j - 1)!} (C^{j-1} \otimes f_y^{(j-1)})
\]
\[
+ \sum_{k=0}^{j-1} (j + \eta) \binom{j - 1 + \eta}{k} (C^k A \otimes f_y^{(k)}) \phi^{j-k-1}, \quad j = 1, \ldots, \mu. \tag{12}
\]

Similarly, inserting (6) into (8) gives
\[
0 = \sum_{k=0}^{\mu} \frac{h^k}{k!} (C^k \otimes g_y^{(k)}) (e \otimes \Delta y_0) + \sum_{k=0}^{\mu} \frac{h^k}{k!} (C^k A \otimes g_y^{(k)}) (h\Delta F) + O(h^{\mu+1} \|\Delta y_0\|),
\]

and by comparing powers of h, we get
\[
0 = (C^j \otimes g_y^{(j)}) + \sum_{k=0}^{j} \frac{j!}{k!(j-k+\eta)!} (C^k A \otimes g_y^{(k)}) \phi^{j-k}, \quad j = 0, \ldots, \mu.
\]

Inserting (12) in the previous expression and solving for \(\psi^{j-1}\) finally leads to the following recurrence relations
\[
\psi^{j-1} = (\eta - 1)! (A^{-1} \otimes (-g_y f_z)^{-1} g_y), \tag{13}
\]
\[
\psi^{j-1} = \frac{(j + \eta - 1)!}{j!} (A^{-1} C^j \otimes (-g_y f_z)^{-1} g_y^{(j)}) + \frac{(j + \eta - 1)!}{(j - 1)!} (C^{j-1} \otimes (-g_y f_z)^{-1} g_y f_y^{(j-1)})
\]
\[
+ \sum_{k=1}^{j} \frac{1}{(j + \eta)} \binom{j + \eta}{k} (A^{-1} C^k A \otimes (-g_y f_z)^{-1} g_y^{(k)}) \phi^{j-k}
\]
\[
+ \sum_{k=1}^{j} \binom{j - 1 + \eta}{k} (C^k \otimes (-g_y f_z)^{-1} g_y^{(k)}) \psi^{j-k-1}
\]
\[
+ \sum_{k=0}^{j-1} \binom{j - 1 + \eta}{k} (C^k A \otimes (-g_y f_z)^{-1} g_y f_y^{(k)}) \phi^{j-k-1}, \quad j = 1, \ldots, \mu. \tag{14}
\]
It can now be verified easily by induction on \( j \) that we get expansions of the form stated in the lemma. □

3. Application to Radau IA methods

Runge–Kutta methods based on Radau IIA quadrature formulae are known to be appropriate for index-2 DAEs. One of the main reason for this is the so-called “stiff accuracy” property, which ensures that the local order of the \( y \)-component is equal to its classical order, as well as the value of the stability function at infinity, which ensures that there is no accumulation of the errors. Our aim is here to show that, while not being stiffly accurate, Radau IA methods have a similar behaviour on index-2 DAEs.

**Lemma 2.** The Radau IA method with \( s \) stages satisfies

\[
A^{-1}e = \frac{1}{b_1}e_1.
\]  

**Proof.** See [10, Lemma 5.6.4, p. 152]. □

**Theorem 1.** In addition to the hypotheses of Lemma 1, we suppose that \( A, b \) and \( c \) are the coefficients of a \( s \)-stage Radau IA method. Then, for \( \eta = s \), the perturbation \( \Delta y_1 \) of Lemma 1 has an expansion of the form

\[
\Delta y_1 = P_0 \Delta y_0 + O\left(h\|P_0 \Delta y_0\|\right) + O\left(h^{s-1}\|Q_0 \Delta y_0\|\right)
\]  

where \( Q_0 = (f_z(g_y f_z)^{-1} g_y)(y(x_0), z(x_0)) \) and \( P_0 = I - Q_0 \).

**Proof.** According to Lemma 1, \( \Delta y_1 \) has an expansion of the form (6) with \( \mu = s - 2 \) (the Radau IA methods satisfy \( C(s - 1) \)). If we split \( \Delta y_0 \) as \( \Delta y_0 = P_0 \Delta y_0 + Q_0 \Delta y_0 \), we see that we can treat the two terms separately. The difficult part is to show that all terms of the form \( \alpha(t)\gamma(t)(b^T \Phi(t)e)F(t)Q_0 \Delta y_0 \) disappear from (6) for all trees up to order \( 2s - 2 \). The two \( y \)-trees of order \( \eta = s \) are \( \emptyset_y \) and \( \overset{1}{\emptyset}_y \) and they correspond to the two expressions \( F(\emptyset_y) = I_M \) and \( F(\overset{1}{\emptyset}_y) = f_z(-g_y f_z)^{-1} g_y = -Q_0 \). Moreover, they have the same number of labellings, the same density and the same elementary weight. Since

\[
(F(\emptyset_y) + F(\overset{1}{\emptyset}_y))Q_0 \Delta y_0 = (Q_0 - Q_0^0)\Delta y_0 = 0,
\]

they can be omitted in the expansion (6). In a similar way, we see that all trees \( t \) of the form

\[
t = \begin{cases} 2^0 \quad \text{or} \quad t = \begin{cases} 2^0 \quad \text{or} \quad t = \begin{cases} 2^0 \quad \text{or} \quad t = \begin{cases} 2^1 \end{cases} 
\end{cases}
\end{cases}
\end{cases}
\]

can be associated with a similar tree \( \tilde{t} \), respectively
For both types of trees, the sum of the two terms,
\[ \alpha(t) \gamma(t) (b^T \Phi(t) e) F(t) \]
and
\[ (\alpha(t) \gamma(t) (b^T \Phi(t) e) F(t)) Q_0 \Delta y_0, \]
is null, since
\[ \alpha(t) = \alpha(t), \]
\[ \gamma(t) = \gamma(t), \]
\[ \Phi(t) = \Phi(t), \]
\[ F(t) Q_0 + F(t) Q_0 = -F(t) + F(t). \]
We thus see that the only remaining trees are those corresponding to elementary differentials of the form
\( f^{(m)}(-g_y f_z)^{-1} g_y \) with \( m > 0 \), i.e., of the form
\[ b^T \ldots C^m A^{-1} e, \quad m > 0. \]

Now, \( A^{-1} e = (1/b_1)e_1 \) and \( C = \text{diag}(0, c_2, \ldots, c_s) \) so that \( C^m A^{-1} e = 0 \), and these trees may also be removed from the summation (6).

Remark 1. The same result holds for Radau IIA methods with \( \eta = s + 1 \) and \( \mu = s - 1 \). The proof is indeed unchanged except for Lemma 2 that has to be replaced by [17, Theorem 4.1].

In order to get a numerical confirmation of the results previously shown, we consider the following system from [16]:

\[
\begin{align*}
y_1'(x) &= y_1(x)y_2^2(x)z^2(x), \\
y_2'(x) &= y_1^2(x)y_2^2(x) - 3y_2^2(x)z(x), \\
0 &= y_1^2(x)y_2(x) - 1,
\end{align*}
\]

(17)
Error in the perturbed solution versus stepsize

Fig. 1. Δy₁ versus h for Radau IA s = 3 and s = 4.

with exact solution \((y_1, y_2, z)(x) = (e^x, e^{-2x}, e^{2x})\). We consider the influence of perturbation at point \(x_0 = 0\) and we thus take

\[
y_0 = \begin{pmatrix} 1 \\ 1 \\ -1000 \end{pmatrix} + h^s \begin{pmatrix} 1 \\ -1000 \\ 1 \end{pmatrix} \quad \text{and} \quad \hat{y}_0 = y_0 + 10^3 h^s \begin{pmatrix} 2 \\ -3 \\ 1 \end{pmatrix} + h^{2s} \begin{pmatrix} 1000 \\ 1 \end{pmatrix},
\]

where it can be verified that \(Q(x_0)(2, -3)^T = (2, -3)^T\). In Fig. 1, we have plotted \(Δy₁\) for various values of \(h\) for the two methods of Radau IA type corresponding to \(s = 3\) and \(s = 4\). We observe clear-cut order 5 and 7, respectively, for the first and second method, in perfect agreement with what was predicted by Theorem 1.

The next lemma deals with the accumulation of errors and is a slight modification of [13, Lemma 4.5].

**Lemma 3.** Suppose that the sequence \(\{Δy_n\}\) satisfies

\[
Δy_{n+1} = P_n Δy_n + O(h ||P_n Δy_n||) + O(h^r ||Q_n Δy_n||),
\]

where \(P_n\) and \(Q_n\) are projectors satisfying \(P_n + Q_n = I\) and \(P_{n+1} = P_n + O(h)\). Then we have for sufficiently small values of \(h\) and \(nh ≤ \text{Const}\)

\[
||Δy_n|| ≤ C ||P_0 Δy_0|| + h^r ||Q_0 Δy_0||.
\]

Using standard techniques (see [13, Theorem 4.4]) and the previous lemma, the following result can be easily proven.

**Theorem 2.** For the \(s\)-stage Radau IA method applied to the index-2 problem (1), the global error satisfies

\[
y_n - y(x_n) = O(h^s),
\]

\[
P(x_n)(y_n - y(x_n)) = O(h^{2s-2}), \quad \text{for} \ s ≥ 2.
\]
Fig. 2. $y_n - y(x_n)$ and $P(x_n)(y_n - y(x_n))$ for Radau IA methods ($s = 3$: o; $s = 4$: x).

By virtue of Remark 1, we see that we get an order of convergence equal to $2s - 2$ if we replace the last step of Radau IA and the projection step by one step of Radau IIA.

4. Numerical experiments

We take up example (17) from previous section and we now integrate from $x_0 = 0$ up to $x = 0.5$. In Fig. 2, we have plotted the numerical solution for the 3-stage and the 4-stage Radau IA methods, with and without a projection after the final step. The step sizes have been chosen at random around a reference value $h$. Once again, we clearly observe the orders predicted by Theorem 2.

5. Interpretation in terms of effective order

Another possible interpretation of the results obtained in previous sections is given by the concept of effective order introduced by J.C. Butcher in [5]. If we consider the composition of one step of the reverse $s$-stage Radau IIA method followed by one step of the $s$-stage Radau IA method and one step of the $s$-stage Radau IIA method (see Fig. 3), we see that the resulting method $\mathcal{R} = (A, b, c)$ satisfies the following simplifying assumptions introduced by J.C. Butcher [6]:

$$B(2s - 1): \quad \sum_{i=1}^{3s} b_i c_i^{k-1} = \frac{1}{k}, \quad k = 1, \ldots, 2s - 1,$$

$$C(s - 1): \quad \sum_{j=1}^{3s} a_{ij} c_i^{k-1} = \frac{1}{k} c_i^k, \quad k = 1, \ldots, s - 1, \quad i = 1, \ldots, 3s,$$
Fig. 3. Composition of the reverse Radau IIA, Radau IA and Radau IIA.

Fig. 4. Effective order.

\[
D(s-1): \sum_{i=1}^{3s} b_i c_i^{k-1} a_{ij} = \frac{b_j}{k} (1 - c_j^k), \quad k = 1, \ldots, s-1, \quad j = 1, \ldots, 3s,
\]

\[(S): \quad b_i = a_{3s,i}, \quad i = 1, \ldots, 3s.
\]

It thus follows from [16], \(-R\) has an \(A\)-matrix with a first row of zeros, that \(R\) has global order \(2s - 2\). In its composition over several steps, we see on Fig. 4 that the Radau IIA steps cancel by pairs leaving in reality only one initial step of the reverse Radau IIA method, \(n\) steps of the Radau IA method and a final step of the Radau IIA method. It must be emphasized that this interpretation remains valid for variable step sizes, if we make \(R\) depend on the current step size \(h\), \(R\) becoming in this case the composition of one step \(h_{ref}\) of the reverse Radau IIA method followed by one step \(h\) of Radau IA and one step \(h_{ref}\) of Radau IIA.

A similar composition can be considered with the Radau IA method replaced by the \(s\)-stage Gauss method. In this case, \(R\) satisfies \(C(s)\) and \(D(s-1)\), as well as \((S)\) and \(B(2s)\) and we thus expect an order of convergence equal to \(2s\). Numerical experiments actually confirm this observation: an analysis in terms of perturbations as done here for Radau IA methods seems to be possible and is under investigation.

References


