

Rosenbrock Methods for Differential Algebraic Equations

Michel Roche

Université de Genève, Département de mathématiques, Rue du Lièvre 2–4, Case postale 240,
CH-1211 Genève 24, Switzerland

Summary. This paper deals with the numerical solution of Differential/Algebraic Equations (DAE) of index one. It begins with the development of a general theory on the Taylor expansion for the exact solutions of these problems, which extends the well-known theory of Butcher for first order ordinary differential equations to DAE's of index one. As an application, we obtain Butcher-type results for Rosenbrock methods applied to DAE's of index one, we characterize numerical methods as applications of certain sets of trees. We derive convergent embedded methods of order 4(3) which require 4 or 5 evaluations of the functions, 1 evaluation of the Jacobian and 1 LU factorization per step.

Subject Classifications: AMS(MOS): 65L05; CR: G 1.7.

1. Introduction

We consider the system of differential/algebraic equations (DAE) of index one

$$y' = f(y, z) \quad y(x_0) = y_0 \quad (1.1 \text{ a})$$

$$0 = g(y, z) \quad z(x_0) = z_0 \quad (1.1 \text{ b})$$

where y is in some space E and z in E' ; we suppose that the initial values are consistent, i.e. $g(y_0, z_0) = 0$. Moreover f and g are assumed to be sufficiently differentiable. Since x can be added to the system as $x' = 1$, it is of course no restriction of generality to assume (1.1) independent of x . We also assume the DAE system to be of index one; this means that $(\partial g / \partial z)^{-1}$ exists and is bounded in a neighbourhood of the exact solution; index 0 systems are ordinary differential equations and systems of index greater than one are algebraically incomplete which means that the existence and the uniqueness of the solutions are not guaranteed (see [1]). Recently much interest in the numerical treatment of (1.1) has appeared in the literature [2–9]. Among the many applications of DAE's in science see [10–12].

In this paper, we consider numerical methods which avoid non-linear equations. The idea is to consider (1.1) as a limit case of the stiff singular perturbation problem

$$y' = f(y, z) \quad y(x_0) = y_0 \quad (1.2a)$$

$$z' = \frac{1}{\varepsilon} g(y, z) \quad z(x_0) = z_0 \quad (1.2b)$$

where ε is a very small real number, and to study the application of known classes of methods to the problem. In particular we apply the general Rosenbrock method (see [13]), premultiply the second equation of the method by ε and set ε to 0. We then obtain the following formulae:

$$a_i = y_0 + \sum_{j=1}^{i-1} \alpha_{ij} l_j \quad (1.3a)$$

$$b_i = z_0 + \sum_{j=1}^{i-1} \alpha_{ij} k_j \quad (1.3b)$$

$$l_i = hf(a_i, b_i) + h \sum_{j=1}^i \gamma_{ij} ((D_y f)_0 l_j + (D_z f)_0 k_j) \quad (1.3c)$$

$$0 = g(a_i, b_i) + \sum_{j=1}^i \gamma_{ij} ((D_y g)_0 l_j + (D_z g)_0 k_j) \quad i = 1, \dots, s \quad (1.3d)$$

$$y_1 = y_0 + \sum_{i=1}^s \mu_i l_i \quad (1.3e)$$

$$z_1 = z_0 + \sum_{i=1}^s \mu_i k_i \quad (1.3f)$$

where α_{ij} , γ_{ij} and μ_i are real parameters, s is the number of stages and $(D_y f)_0$, $(D_z f)_0$, $(D_y g)_0$, $(D_z g)_0$ are the derivatives at the initial values (y_0, z_0) .

For each $i = 1, \dots, s$, (1.3c) and (1.3d) form a linear system in l_i and k_i with matrix

$$\begin{pmatrix} I - h\gamma_{ii}(D_y f)_0 & -h\gamma_{ii}(D_z f)_0 \\ -h\gamma_{ii}(D_y g)_0 & -h\gamma_{ii}(D_z g)_0 \end{pmatrix}.$$

If we choose $\gamma_{ii} = \gamma$ for all i , all these matrices are equal and we only need one LU-factorisation per step.

It happens that the limit process $\varepsilon \rightarrow 0$ destroys the order properties of the Rosenbrock methods. For example the well known method of Kaps-Rentrop (see [14]) which is of order 4 for ordinary differential equations is only of order 2 for the problem (1.1) as will be seen in Tables 1 and 2 below. Similar phenomena have been observed by Verwer [15].

The aim of this article is to study the order conditions for method (1.3). In Sect. 2 we develop a theory for the Taylor expansion of the exact solutions of a DAE of index one with the help of a "tree model". In Sect. 3, using this tree model, we find Butcher-type results for the numerical solution. In Sect. 4

Table 1. Order equation for the y -component














$\rho(t)$	t		
1		\dots	$\sum \mu_i = 1$ (4.11 a)
2		\dots	$\sum \mu_i \beta_{ij} = \frac{1}{2}$ (4.11 b)
3		\dots	$\sum \mu_i \alpha_{ij} \alpha_{ik} = \frac{1}{3}$ (4.11 c)
3		\dots	$\sum \mu_i \beta_{ij} \beta_{jk} = \frac{1}{6}$ (4.11 d)
4		\dots	$\sum \mu_i \alpha_{ij} \alpha_{ik} \alpha_{il} = \frac{1}{4}$ (4.11 e)
4		\dots	$\sum \mu_i \alpha_{ij} \alpha_{ik} \beta_{jl} = \frac{1}{8}$ (4.11 f)
4		\dots	$\sum \mu_i \beta_{ij} \alpha_{jk} \alpha_{jl} = \frac{1}{12}$ (4.11 g)
4		\dots	$\sum \mu_i \beta_{ij} \beta_{jk} \beta_{kl} = \frac{1}{24}$ (4.11 h)
4		\dots	$\sum \mu_i \alpha_{ij} \alpha_{ik} w_{kl} \alpha_{lm} \alpha_{ln} = \frac{1}{4}$ (4.11 i)
...

Table 2. Order equation for the z -component

$\rho(t)$	t		
2		\dots	$\sum \mu_i w_{ij} \alpha_{jk} \alpha_{jl} = 1$ (4.12 a)
3		\dots	$\sum \mu_i w_{ij} \alpha_{jk} \alpha_{jl} \alpha_{jm} = 1$ (4.12 b)
3		\dots	$\sum \mu_i w_{ij} \alpha_{jk} \alpha_{jl} \beta_{lm} = \frac{1}{2}$ (4.12 c)
3		\dots	$\sum \mu_i w_{ij} \alpha_{jk} \alpha_{jl} w_{lm} \alpha_{mn} \alpha_{mp} = 1$ (4.12 d)
...

the convergence and the order conditions of method (1.3) are studied. Solving these conditions in Sect. 5, we give the methods with s stages $s = 1, \dots, 5$ having the highest possible order. In Sect. 6 numerical experiments are presented.

2. Trees and Elementary Differentials

Notice first that differentiation of (1.1 b) gives

$$0 = D_y g \cdot y' + D_z g \cdot z'$$

so we have

$$z' = (-D_z g)^{-1} D_y g \cdot y' \tag{2.1}$$

Using the chain rule, $y' = f$, (2.1) and

$$((-D_z g)^{-1})' = (-D_z g)^{-1} \cdot (D_y D_z g \cdot y' + D_z^2 g \cdot z') \cdot (-D_z g)^{-1}$$

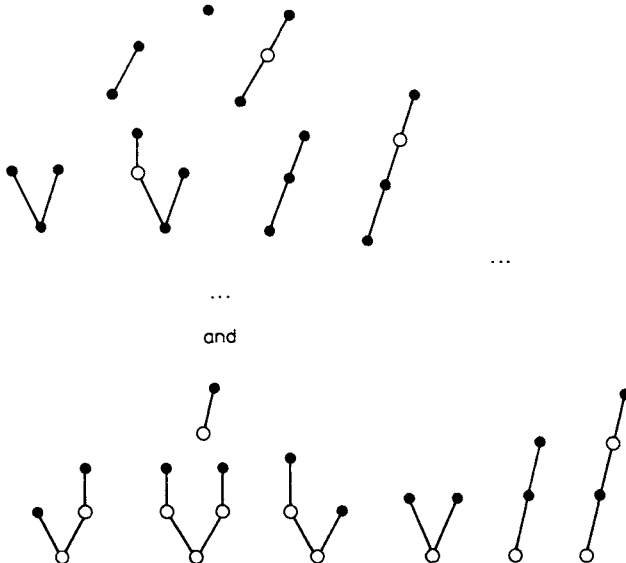
a continued differentiation of (1.1 a) and (2.1) gives:

$$\begin{aligned} y' &= f \\ y'' &= D_y f \cdot f + D_z f \cdot (-D_z g)^{-1} \cdot D_y g \cdot f \\ y''' &= D_y^2 f \cdot (f, f) + D_z D_y f \cdot ((-D_z g)^{-1} \cdot D_y g \cdot f, f) + D_y f D_y f \cdot f \\ &\quad + D_y f D_z f \cdot (-D_z g)^{-1} \cdot D_y g \cdot f + \dots \end{aligned}$$

and

$$\begin{aligned} z' &= (-D_z g)^{-1} \cdot D_y g \cdot f \\ z'' &= (-D_z g)^{-1} [D_y D_z g \cdot (f, (-D_z g)^{-1} \cdot D_y g \cdot f) \\ &\quad + D_z^2 g \cdot ((-D_z g)^{-1} \cdot D_y g \cdot f, (-D_z g)^{-1} \cdot D_y g \cdot f) + D_z D_y g \cdot ((-D_z g)^{-1} \cdot D_y g \cdot f, f) \\ &\quad + D_y^2 g \cdot (f, f) + D_y g \cdot D_y f \cdot f + D_y g \cdot D_z f \cdot (-D_z g)^{-1} \cdot D_y g \cdot f] \end{aligned}$$

We now identify f with a meagre vertex, any partial derivative of f with a branch leaving a meagre vertex, $(-D_z g)^{-1} g$ with a fat vertex, and any partial derivative of g with a branch leaving a fat vertex. The above expressions, which very soon become complicated, can be written in terms of trees as follows:



For a better understanding of the recursive construction of the trees, see Proposition (2.5) and the example that follows it.

Now let τ_y and τ_z be the following trees:

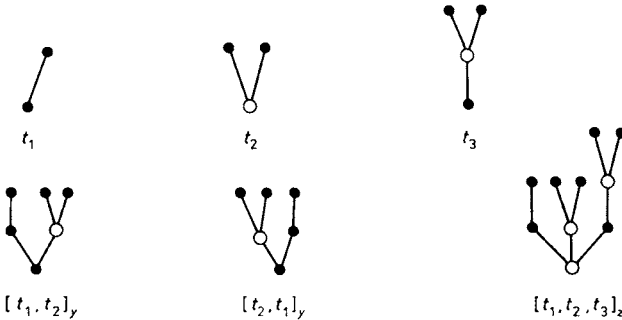


Definition (2.1). We denote by DAT , DAT_y and DAT_z the sets of the trees defined recursively by:

- a) $\emptyset \in DAT_y \cap DAT_z, \tau_y \in DAT_y, \tau_z \in DAT_z.$
- b) If $t_1, \dots, t_n \in DAT_y \cup DAT_z$, then $[t_1, \dots, t_n]_y \in DAT_y$
- c) If $t_1, \dots, t_m \in DAT_y \cup DAT_z, m > 1$, or $m = 1$ and $t_1 \in DAT_y$, then $[t_1, \dots, t_m]_z \in DAT_z$
- d) $DAT = DAT_y \cup DAT_z$

where $t = [t_1, \dots, t_n]_y$ is the tree obtained by connecting the roots of t_1, \dots, t_n by n arcs to a new meagre vertex which becomes the root of t , and $[t_1, \dots, t_n]_z$ is the tree obtained in the same manner, but with a new fat root.

Examples:

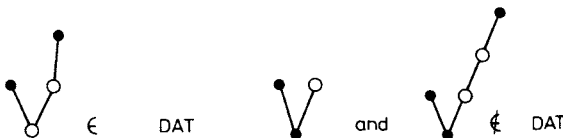


Notice that $[t_1, t_2]_y = [t_2, t_1]_y$

Remark. DAT_y and DAT_z can be seen as the sets of all the connected graphs with two different kinds of vertices, meagre vertices and fat vertices, which satisfy:

- a) The end vertices of the graph are meagre.
- b) A graph is in DAT_y if its root is meagre, and in DAT_z if its root is fat.
- c) If a fat vertex has no ramification, then the above vertex is meagre.

Example:



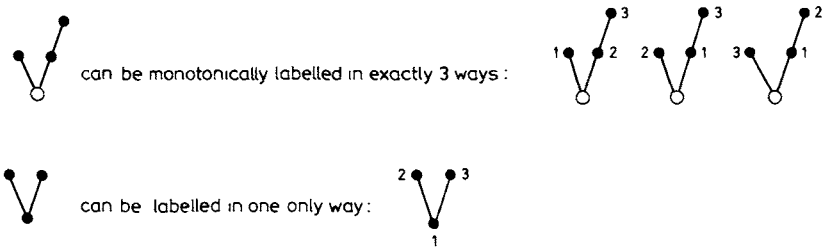
Definition (2.2). The number of meagre vertices of a tree t is called the *order* of t , denoted by $\rho(t)$.

Labelling

We now introduce the concept of labelled trees which is very helpful for the formulation of the theory:

Definition (2.3). Let $t \in DAT$. We say that t is *monotonically labelled* if every meagre vertex is associated with an integer i , $1 \leq i \leq \rho(t)$ and if, following any branch of t , the labels are monotonically increasing. The number of possible labellings of t is denoted by $\alpha(t)$. Finally, $LDAT_y$ denotes the set of monotonically labelled trees having a meagre root, $LDAT_z$ the set of monotonically labelled trees having a fat root and $LDAT = LDAT_y \cup LDAT_z$.

Examples:



Elementary Differentials

We give now a recursive definition of the terms which appear in the Taylor expansion of the exact solution of (1.1) and which are in one-to-one correspondance with the trees of DAT .

Definition (2.4). For every tree t of DAT_y we define a function $F(t): E \times E' \rightarrow E$ and for every tree u of DAT_z a function $G(u): E \times E' \rightarrow E'$ recursively by:

- a) $F(\emptyset)(y, z) = y, G(\emptyset)(y, z) = z$
- b) $F(\tau_y)(y, z) = f, G(\tau_z)(y, z) = (-D_z g)^{-1} \cdot D_y g \cdot f$
- c) $F(t)(y, z) = D_y^k D_z^l f \cdot (F(t_1), \dots, F(t_k), G(u_1), \dots, G(u_l))$ if $t = [t_1, \dots, t_k, u_1, \dots, u_l]_y$
- d) $G(u)(y, z) = ((-D_z g)^{-1}) \cdot D_y^k D_z^l g \cdot (F(t_1), \dots, F(t_k), G(u_1), \dots, G(u_l))$ if $u = [t_1, \dots, t_k, u_1, \dots, u_l]_z$ where $t_1, \dots, t_k \in DAT_y$ and $u_1, \dots, u_l \in DAT_z$

The expressions $F(t)(y, z)$, respectively $G(u)(y, z)$, are called the *elementary differentials* associated with the tree t , respectively u .

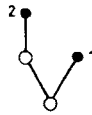
Because of the symmetry of partial derivatives, this definition does not depend on permutations amongst $t_1, \dots, t_k, u_1, \dots, u_l$ and therefore the functions F and G are well defined.

Proposition (2.5). *Let t be any tree of DAT, then the derivation with respect to x of its elementary differential consists of: (1) Splitting each fat vertex into two fat vertices and attaching at the lower of these vertices once τ_y and once τ_z (derivation of $(-D_z g)^{-1}$ with respect once to y and once to z). (2) attaching to each vertex of t once τ_y (derivative of the other terms with respect to y) and once τ_z (derivative of the other terms with respect to z).*

Proof. Comes from Definitions (2.1) and (2.4). See also the next example.

Remark (2.6). If $t \in \text{LDAT}$ (labelled tree), the labelling of the new trees built by derivation of the elementary differential of t simply consists of associating the new meagre vertex with the integer $\rho(t) + 1$.

Example:



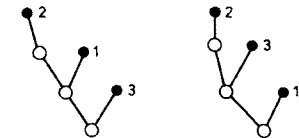
corresponds to the elementary differential:

$$(-D_z g)^{-1} \cdot D_y D_z g \cdot (f, (-D_z g)^{-1} \cdot D_y g \cdot f)$$

The derivation of this expression gives elementary differentials corresponding to the following trees:

(1) Derivation of $(-D_z g)^{-1}$:

a) with respect to y

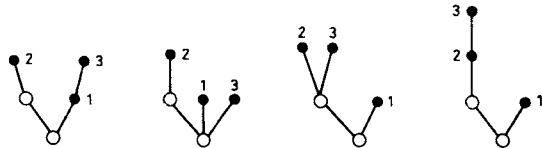


b) with respect to z

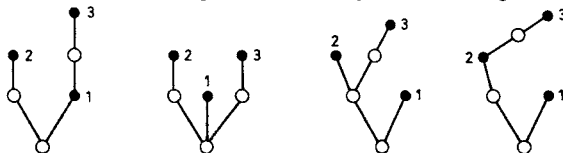


(2) Derivation of the other terms:

a) with respect to y



b) with respect to z



Following this procedure, the labels indicate the order of generation of the meagre vertices. Therefore the numbers of ways of labelling a tree t of DAT is equal to the number of times an elementary differential appears in the Taylor

expansion of the exact solutions of (1.1). As can be seen from Proposition (2.5) and Remark (2.6), every tree $t \in LDAT$ of order p appears once and exactly once in the p -th derivative of y (if $t \in LDAT_y$) or of z (if $t \in LDAT_z$). Thus:

Theorem (2.7). *For the exact solution of (1.1) we have:*

$$y^{(p)}(x_0) = \sum_{t \in LDAT_y, \rho(t)=p} F(t)(y_0, z_0) = \sum_{t \in DAT_y, \rho(t)=p} \alpha(t) \cdot F(t)(y_0, z_0)$$

$$z^{(p)}(x_0) = \sum_{u \in LDAT_z, \rho(u)=p} G(u)(y_0, z_0) = \sum_{u \in DAT_z, \rho(u)=p} \alpha(u) \cdot G(u)(y_0, z_0)$$

and

$$y(x_0 + h) = \sum_{t \in LDAT_y} F(t)(y_0, z_0) \cdot \frac{h^{\rho(t)}}{\rho(t)!}$$

$$z(x_0 + h) = \sum_{u \in LDAT_z} G(u)(y_0, z_0) \cdot \frac{h^{\rho(u)}}{\rho(u)!}$$

3. DA-Series

In Sect. 2, we described a very simple way to find the Taylor expansion of the exact solutions of (1.1) with the help of a “tree model”. To find the order of a numerical method applied to (1.1), one has to compare the Taylor expansion of the numerical solutions with those of the exact solutions. Applying Theorem (2.7), we now extend the concept of Butcher-series (see [16]).

Definition (3.1). Let $\mathbf{a}: LDAT_y \rightarrow \mathbf{R}$ and $\mathbf{b}: LDAT_z \rightarrow \mathbf{R}$ be any mappings. The series

$$DA_y(\mathbf{a}, y_0, z_0) = \sum_{t \in LDAT_y} \mathbf{a}(t) \cdot F(t)(y_0, z_0) \frac{h^{\rho(t)}}{\rho(t)!}$$

respectively

$$DA_z(\mathbf{b}, y_0, z_0) = \sum_{u \in LDAT_z} \mathbf{b}(u) \cdot G(u)(y_0, z_0) \cdot \frac{h^{\rho(u)}}{\rho(u)!}$$

are called DA_y -series, respectively DA_z -series.

Observe that the exact solutions of (1.1) are DA -series (see Theorem (2.7)):

$$y(x) = DA_y(\mathbf{p}_y, y_0, z_0)$$

$$z(x) = DA_z(\mathbf{p}_z, y_0, z_0)$$

where $\mathbf{p}_y(t) = 1$ for all $t \in LDAT_y$ and $\mathbf{p}_z(u) = 1$ for all $u \in LDAT_z$.

Results for DA-Series

Theorem (3.2). *Let a and b be DA-series, $a = DA_y(\mathbf{a}, y_0, z_0)$ and $b = DA_z(\mathbf{b}, y_0, z_0)$. We have: $c = h \cdot f(a, b)$ is a DA_y -series with coefficients $\mathbf{c}: LDAT_y \rightarrow \mathbf{R}$ defined by*

$$\mathbf{c}(\emptyset) = 0 \quad \mathbf{c}(\tau_y) = 1$$

$$\mathbf{c}(t) = \rho(t) \mathbf{a}(t_1) \dots \mathbf{a}(t_k) \mathbf{b}(u_1) \dots \mathbf{b}(u_l)$$

for $t = [t_1, \dots, t_k, u_1, \dots, u_l]_y$ where $t_1, \dots, t_k \in LDAT_y$ and $u_1, \dots, u_l \in LDAT_z$.

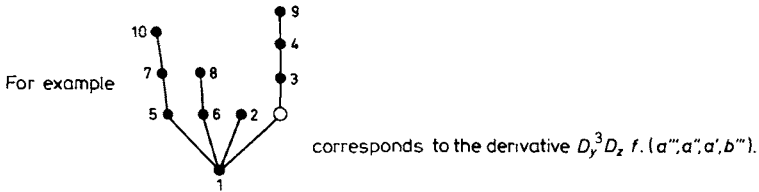
Proof. This theorem can be seen as a generalisation of Theorem (2.11) in [16]. Consider first the n -th derivative of c evaluated at $h=0$:

$$c^{(n)}(0) = n \cdot \left[\frac{\partial^{n-1}}{\partial h^{n-1}} f(a(h), b(h)) \right]_0$$

We have:

$$c^{(n)}(0) = n \cdot \sum_{\substack{\bar{t} \in SLDAT_y, \rho(\bar{t}) = n \\ \bar{t} = [t_1, \dots, t_k, u_1, \dots, u_l]_y}} (D_y^k D_z^l f)_0 \cdot (a^{(i_1)}, \dots, a^{(i_k)}, b^{(j_1)}, \dots, b^{(j_l)})_0 \quad (3.3)$$

where $\rho(t_s) = i_s, s = 1, \dots, k$ and $\rho(u_p) = j_p, p = 1, \dots, l$ and $SLDAT_y \subset LDAT_y$ is the subset of trees having no ramification (except possibly at the root) and such that only the vertices directly connected with the root can be fat. The summand in (3.3) corresponding to a tree $\bar{t} \in SLDAT_y$ begins with $D_y^k D_z^l f$ if \bar{t} has k branches with a meagre root and l branches with a fat root. The number of meagre vertices of each branch equals the order of differentiation of a (if the branch has a meagre root) or of b (if the branch has a fat root).



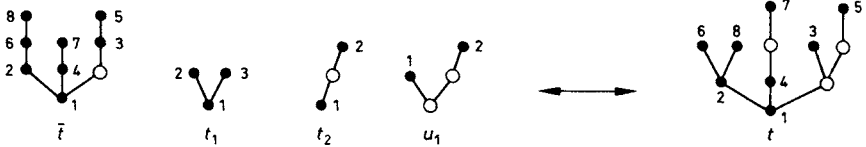
As a and b are DA -series, we have:

$$a^{(i_s)}(0) = \sum_{t_s \in LDAT_y, \rho(t_s) = i_s} \mathbf{a}(t_s) \cdot F(t_s)_0$$

$$b^{(j_p)}(0) = \sum_{u_p \in LDAT_z, \rho(u_p) = j_p} \mathbf{b}(u_p) \cdot G(u_p)_0$$

We now insert the above formulae into (3.3) and get a summation over the tuples $(\bar{t}, t_1, \dots, t_k, u_1, \dots, u_l)$. The main difficulty is now to understand that to each such tuple there corresponds a labelled tree $t \in LDAT_y$ such that the summand is $\rho(t) \mathbf{a}(t_1) \dots \mathbf{a}(t_k) \mathbf{b}(u_1) \dots \mathbf{b}(u_l) \cdot F(t)_0$. This labelled tree t is obtained by replacing the branches of t having a meagre root by t_1, \dots, t_k and those having a fat root by u_1, \dots, u_l . The labelling is carried over in a natural way, i.e. in the same order.

Example:



In this way, all the trees $t \in \text{LDAT}_y$ appear exactly once. Thus (3.3) becomes:

$$\begin{aligned} c^{(n)}(0) &= n \cdot \sum_{\substack{t \in \text{LDAT}_y, \rho(t) = n \\ t = [t_1, \dots, t_k, u_1, \dots, u_l]_y}} \mathbf{a}(t_1) \dots \mathbf{a}(t_k) \mathbf{b}(u_1) \dots \mathbf{b}(u_l) \\ &\quad \cdot [D_y^k D_z^l f \cdot (F(t_1), \dots, F(t_k), G(u_1), \dots, G(u_l))]_0 \\ &= \sum_{\substack{t \in \text{LDAT}_y, \rho(t) = n \\ t = [t_1, \dots, t_k, u_1, \dots, u_l]_y}} \rho(t) \cdot \mathbf{a}(t_1) \dots \mathbf{a}(t_k) \mathbf{b}(u_1) \dots \mathbf{b}(u_l) \cdot F(t)_0. \end{aligned}$$

As $c(h) = \sum_{n \geq 0} c^{(n)}(0) \cdot \frac{h^n}{n!}$, using Definition (3.1), the proof is complete.

Theorem (3.4). Under the assumptions of Theorem (3.2), we have:

$$d = (-D_z g)_0^{-1} \cdot g(a, b) \text{ is a } DA_z\text{-series}$$

with coefficients $\mathbf{d}: \text{LDAT}_z \rightarrow \mathbf{R}$ defined by:

$$\begin{aligned} \mathbf{d}(\emptyset) &= 0 & \mathbf{d}(\tau_z) &= 1 \\ \mathbf{d}(u) &= \mathbf{a}(t_1) \dots \mathbf{a}(t_k) \mathbf{b}(u_1) \dots \mathbf{b}(u_l) - \mathbf{b}(u) \end{aligned}$$

for $u = [t_1, \dots, t_k, u_1, \dots, u_l]_z$ where $t_1, \dots, t_k \in \text{LDAT}_y$ and $u_1, \dots, u_l \in \text{LDAT}_z$.

Proof. Proceeding as in the previous proof we obtain:

$$d^{(n)}(0) = (-D_z g)_0^{-1} \left[\frac{\partial^n}{\partial h^n} g(a(h), b(h)) \right]_0$$

and

$$\begin{aligned} d^{(n)}(0) &= \sum_{\substack{u \in \text{SLDAT}_z, \rho(u) = n \\ u = [t_1, \dots, t_k, u_1, \dots, u_l]_z}} ((-D_z g)^{-1} D_y^k D_z^l g \\ &\quad \cdot (a^{(i_1)}, \dots, a^{(i_k)}, b^{(j_1)}, \dots, b^{(j_l)}))_0 - b^{(n)}(0) \end{aligned}$$

where $\rho(t_s) = i_s$, $s = 1, \dots, k$ and $\rho(u_p) = j_p$, $p = 1, \dots, l$.

Then

$$\begin{aligned} d^{(n)}(0) &= \sum_{\substack{u \in \text{LDAT}_z, \rho(u) = n \\ u = [t_1, \dots, t_k, u_1, \dots, u_l]_z}} \mathbf{a}(t_1) \dots \mathbf{a}(t_k) \mathbf{b}(u_1) \dots \mathbf{b}(u_l) \cdot [(-D_z g)^{-1} D_y^k D_z^l g \\ &\quad \cdot (F(t_1), \dots, F(t_k), G(u_1) \dots G(u_l))]_0 - \sum_{u \in \text{LDAT}_z, \rho(u) = n} \mathbf{b}(u) \cdot G(u)_0 \\ &= \sum_{\substack{u \in \text{LDAT}_z, \rho(u) = n \\ u = [t_1, \dots, t_k, u_1, \dots, u_l]_z}} (\mathbf{a}(t_1) \dots \mathbf{a}(t_k) \mathbf{b}(u_1) \dots \mathbf{b}(u_l) - \mathbf{b}(u)) \cdot G(u)_0 \end{aligned}$$

Finally we obtain the following results:

- Theorem (3.5).** a) $(D_y f)_0 \cdot F(t)(y_0, z_0) = F([t]_y)(y_0, z_0)$ for $t \in LDAT_y$
 b) $(D_z f)_0 \cdot G(u)(y_0, z_0) = F([u]_y)(y_0, z_0)$ for $u \in LDAT_z$
 c) $((-D_z g)^{-1} D_y g)_0 \cdot F(t)(y_0, z_0) = G([t]_z)(y_0, z_0)$ for $t \in LDAT_y$.

Proof. Comes from Definition (2.4).

4. Order Conditions and Convergence

This section deals with the order and the convergence of one-step methods for DAE's of index one; the order and convergence conditions for method (1.3) are given explicitly.

We consider the following general class of one-step methods (only formally explicit) applied to (1.1):

$$y_1 = y_0 + h\Phi(y_0, z_0, h) \tag{4.1 a}$$

$$z_1 = \Psi(y_0, z_0, h) \tag{4.1 b}$$

Definition (4.2). The method (4.1) is of order p if

$$y(x_0 + h) - y_1 = O(h^{p+1}) \quad \text{and} \quad z(x_0 + h) - z_1 = O(h^p)$$

where y and z are the exact solutions of (1.1).

The proof of the next result can be found in [9].

Theorem (4.3). Consider method (4.1) and suppose:

- a) its order is p
 b) $\left\| \frac{\partial \Psi(y, z, 0)}{\partial z} \right\| \leq \alpha < 1$ in a neighbourhood of the solution.

Then convergence of order p occurs, i.e. for $x = n \cdot h$ fixed:

$$y_n - y(x) = O(h^p) \quad \text{and} \quad z_n - z(x) = O(h^p)$$

where y_n and z_n denote the numerical solutions of (1.1) when method (4.1) is applied n times.

a) Order Conditions for Method (1.3)

We now use Theorems (3.2), (3.4) and (3.5) to derive the order conditions for the coefficients α_{ij} , γ_{ij} and μ_i of method (1.3), by comparing the DA-series of $y(x_0 + h)$ and $y_1(x_0 + h)$ (the numerical solution) up to a certain order, and similarly for $z(x_0 + h)$ and $z_1(x_0 + h)$.

Let us first consider the functions $a_i, b_i, l_i, k_i, (i=1, \dots, s)$ and y_1, z_1 defined by (1.3).

Theorem (4.4). The functions $a_i, b_i, l_i, k_i, y_1, z_1$ are DA-series whose coefficients $\mathbf{a}_i, \mathbf{b}_i, \mathbf{l}_i, \mathbf{k}_i, \mathbf{y}_1, \mathbf{z}_1$ are recursively defined by:

$$\mathbf{a}_i(t) = \sum_{j=1}^{i-1} \alpha_{ij} \mathbf{l}_j(t) \quad \text{for } t \in \text{LDAT}_y \quad (4.5a)$$

$$\mathbf{b}_i(u) = \sum_{j=1}^{i-1} \alpha_{ij} \mathbf{k}_j(u) \quad \text{for } u \in \text{LDAT}_z \quad (4.5b)$$

$$\mathbf{l}_i(\emptyset) = 0 \quad \mathbf{l}_i(\tau_y) = 1$$

$$\mathbf{l}_i(t) = \rho(t) \mathbf{a}(t_1) \dots \mathbf{a}(t_k) \mathbf{b}(u_1) \dots \mathbf{b}(u_l) + \begin{cases} 0 & \text{if } k+l > 1 \\ \rho(t) \sum_{j=1}^i \gamma_{ij} \mathbf{l}_j(t_1) & \text{if } k=1, l=0 \\ \rho(t) \sum_{j=1}^i \gamma_{ij} \mathbf{k}_j(u_1) & \text{if } k=0, l=1 \end{cases} \quad (4.5c)$$

for $t = [t_1, \dots, t_k, u_1, \dots, u_l]_y$ where $t_1, \dots, t_k \in \text{LDAT}_y$ and $u_1, \dots, u_l \in \text{LDAT}_z$

$$\mathbf{k}_i(\emptyset) = 0 \quad \mathbf{k}_i(\tau_z) = 1$$

$$0 = \begin{cases} \mathbf{a}(t_1) \dots \mathbf{a}(t_k) \mathbf{b}(u_1) \dots \mathbf{b}(u_l) - \sum_{j=1}^i (\alpha_{ij} + \gamma_{ij}) \mathbf{k}_j(u) & \text{if } k+l > 1 \\ \sum_{j=1}^i (\alpha_{ij} + \gamma_{ij}) (\mathbf{l}_j(t_1) - \mathbf{k}_j(u)) & \text{if } k=1, l=0 \end{cases} \quad (4.5d)$$

for $u = [t_1, \dots, t_k, u_1, \dots, u_l]_z$ where $t_1, \dots, t_k \in \text{LDAT}_y$ and $u_1, \dots, u_l \in \text{LDAT}_z$

$$y_1(t) = \sum_{i=1}^s \mu_i \mathbf{l}_i(t) \quad \text{for } t \in \text{LDAT}_y \quad (4.5e)$$

$$z_1(u) = \sum_{i=1}^s \mu_i \mathbf{k}_i(u) \quad \text{for } u \in \text{LDAT}_z \quad (4.5f)$$

Proof. (4.5a), (4.5b), (4.5e) and (4.5f) follow directly from (1.3a), (1.3b), (1.3e) and (1.3f).

(4.5c) follows from (1.3c), Theorem (3.2) and Theorem (3.5) (a) and b)).

(4.5d) follows from (1.3d), Theorem (3.4) and Theorem (3.5) (c)). Q.E.D.

To simplify equations (4.5c) and (4.5d), we put:

$$\beta_{ij} = \alpha_{ij} + \gamma_{ij} \quad (\beta_{ii} = \gamma)$$

We obtain:

$$\mathbf{l}_i(t) = \rho(t) \begin{cases} \sum_{\substack{n_1, \dots, n_k \\ m_1, \dots, m_l}} \alpha_{in_1} \dots \alpha_{im_l} \mathbf{l}_{n_1}(t_1) \dots \mathbf{k}_{m_l}(u_l) & \text{if } k+l > 1 \\ \sum_{j=1}^i \beta_{ij} \mathbf{l}_j(t_1) & \text{if } k=1, l=0 \\ \sum_{j=1}^i \beta_{ij} \mathbf{k}_j(u_1) & \text{if } k=0, l=1 \end{cases} \quad (4.5c')$$

$$0 = \begin{cases} \sum_{\substack{n_1, \dots, n_k \\ m_1, \dots, m_l}} \alpha_{in_1} \dots \alpha_{im_l} \mathbf{l}_{n_1}(t_1) \dots \mathbf{k}_{m_l}(u) - \sum_{j=1}^i \beta_{ij} \mathbf{k}_j(u) & \text{if } k+l > 1 \\ \sum_{j=1}^i \beta_{ij} (\mathbf{l}_j(t_1) - \mathbf{k}_j(u)) & \text{if } k=1, l=0 \end{cases} \quad (4.5d)'$$

Proposition (4.6). For $u = [t_1]_z$, $t_1 \in \text{LDAT}_y$, we have:

$$\mathbf{k}_i(u) = \mathbf{l}_i(t_1)$$

Proof. For $i=1$, (4.5d)' gives: $\mathbf{k}_1(u) = \mathbf{l}_1(t_1)$.
Then use (4.5d)', and induct. Q.E.D.

Example:



$$\mathbf{k}_i(u) = \mathbf{l}_i(t)$$

We now set $\tilde{\beta} = (\beta_{ij})_{i=1, \dots, i}^{j=1, \dots, i}$ and 0 in the others places and

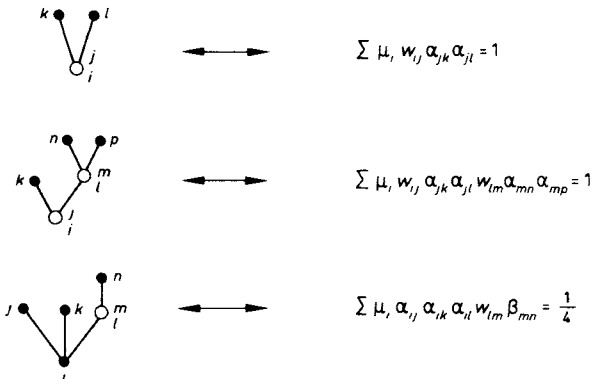
$$W = (w_{ij}) = \tilde{\beta}^{-1} \quad (4.7)$$

We then obtain:

$$\mathbf{k}_i(u) = \begin{cases} \sum_{j=1}^i w_{ij} \sum_{\substack{n_1, \dots, n_k \\ m_1, \dots, m_l}} \alpha_{jn_1} \dots \alpha_{jm_l} \mathbf{l}_{n_1}(t_1) \dots \mathbf{k}_{m_l}(u) & \text{if } k+l > 1 \\ \mathbf{l}_i(t_1) & \text{if } k=1, l=0 \end{cases} \quad (4.5d)''$$

Remark (4.8). Using (4.5c)' and (4.5d)'', there is a very simple way to find the order conditions for the trees of LDAT by comparing the DA-series (4.5e) and (4.5f) of Theorem (4.4) with the DA-series of the exact solutions (see Theorem (2.7)).

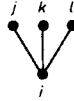
Examples:



and as $W = \tilde{\beta}^{-1}$, the last equation is:

$$\sum \mu_i \alpha_{ij} \alpha_{ik} \alpha_{il} = \frac{1}{4}$$

which is also obtained from the following tree:

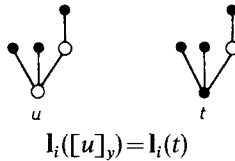


Proposition (4.9). For $u = [t_1, \dots, t_k, u_1, \dots, u_l]_z$ where $t_1, \dots, t_k \in LDAT_y$ and $u_1, \dots, u_l \in LDAT_z$, we have:

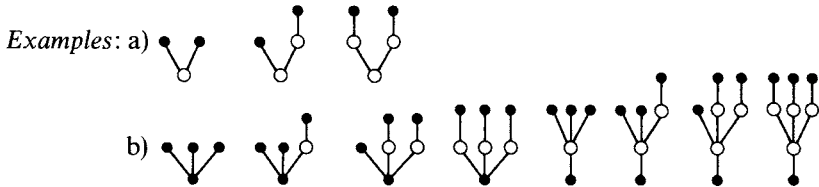
$$I_i([u]_y) = I_i([t_1, \dots, t_k, u_1, \dots, u_l]_y)$$

Proof. Comes directly from formulae (4.5c)' and (4.5d)'', or more nicely with the help of Remark (4.8) and the above examples. Q.E.D.

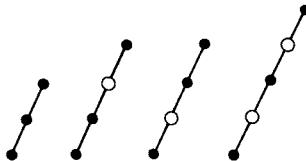
Example:



An important consequence of Propositions (4.6) and (4.9) is that a lot of different trees give the same order conditions.



c) All the trees of $LDAT_y$ of order n having no ramification. For example, if $n = 3$:



Proposition (4.10). The order conditions for t having only meagre vertices are identical to the order conditions of the classical Rosenbrock method for ODE's.

Proof. Set $l=0$ in (4.5c)' (so one has trees with meagre vertices only) and compare it with Theorem 1 or Theorem 2 of [13]. Q.E.D.

We give in Tables 1 and 2 the first order equations (for the z -component, because of Proposition (4.6), we only give the supplementary conditions).

The equations given in Tables 1 and 2 are the conditions for having a method of order 4. The convergence has now to be studied.

b) Calculation of α

To assure convergence of method (1.3), the condition b) of Theorem (4.3) must be satisfied; let $\mathbf{R}(z)$ be the stability function of the Rosenbrock method for the test ordinary differential equation $y' = \lambda y$, $y(0) = 1$, $\lambda \in \mathbb{C}$, $z = h\lambda$; we then have:

Theorem (4.13). *The contractivity number α is given by:*

$$\alpha = \mathbf{R}(\infty)$$

Proof. Let $\beta = (\beta_{ij})_{i=1, \dots, s}^{j=1, \dots, i-1}$ and 0 in the others places, $\vec{1} = (1, \dots, 1)^t$ and $\frac{\partial \vec{k}}{\partial z}(0) = \left(\frac{\partial k_1}{\partial z}(0), \dots, \frac{\partial k_s}{\partial z}(0) \right)^t$, vectors of dimension s . We have:

$$\alpha = 1 + \vec{\mu}^t \frac{\partial \vec{k}}{\partial z}(0) \tag{4.13.1}$$

Derivating equation (1.3d) with respect to z and evaluating the result at $h=0$ gives:

$$0 = (D_z g)_0 \cdot \left(1 + \sum_{j=1}^i \beta_{ij} \frac{\partial k_j}{\partial z}(0) \right)$$

and then

$$\frac{\partial \vec{k}}{\partial z}(0) = \sum_{j=1}^s (-1)^j \frac{1}{\gamma^j} \beta^{j-1} \cdot \vec{1} \tag{4.13.2}$$

Now, as

$$R(\infty) = 1 + \sum_{j=1}^s (-1)^j \frac{1}{\gamma^j} \vec{\mu}^t \beta^{j-1} \cdot \vec{1}$$

so, by inserting (4.13.1) into (4.13.2), one gets $\alpha = R(\infty)$. Q.E.D.

Remark (4.14). The evident underlying reason for this result is that $R(\infty)$ must be smaller than 1 because of the limit process $\varepsilon \rightarrow 0$ (or $\lambda \rightarrow \infty$ in this case), i.e., stability at infinity is necessary because (1.1) is considered as a limit case for (1.2).

5. Some Particular Methods

In this section we give the main results for the methods with s stages, $s = 1, \dots, 5$ having the highest possible order. Notice first that the matrix W defined by (4.7) satisfies

$$W = \frac{1}{\gamma} \sum_{j=0}^{s-1} \left(-\frac{\beta}{\gamma} \right)^j$$

which is helpful for the calculations of the order conditions.

Solving the order equations for different values of s , we obtain, for convergent methods, the results given in Table 3.

Table 3. Order of convergence for Rosenbrock methods

s	1	2	3	4	5
Order	1	2	3	3	4

Remark. For $s=1$, the method with $\mu_1=1$ and $\gamma=1/2$ is of order 2 but unfortunately not convergent of order 2, (only of order 1 because $\alpha=1$). The method with $\mu_1=1$ and $\gamma=1$ is convergent of order 1; in fact it is exactly the semi-implicit Euler discretization used in [9] for extrapolation.

We first give embedded methods of order 3(2) with $s=3$, $R(\infty)=0$ and only 2 evaluations of the function f and g :

Choose γ such that $R(\infty)=0$ and such that A-stability is assured for purely differential equations, i.e. $\gamma \in [1/3; 1.06858]$; choose $\alpha_{32}=0$ and $\alpha_{31}=\alpha_{21}$ to assure only 2 evaluations of the functions f and g ; choose also $\hat{\mu}_3=0$, so that $\hat{s}=2$, i.e. 2 stages for the method of order 2. Now α_{21} and β_{32} are non zero free parameters. Then:

$$\beta_{21} = \frac{1/6 - \gamma + \gamma^2}{-\gamma^2 + \gamma/3} \cdot \alpha_{21}^2 \quad \mu_3 = \frac{1/6 - \gamma + \gamma^2}{\beta_{32} \beta_{21}} \quad \mu_2 = \frac{1/3}{\alpha_{21}^2} - \mu_3$$

$$\beta_{31} = \frac{1/2 - \gamma - \mu_2 \beta_{21}}{\mu_3} - \beta_{32} \quad \text{and} \quad \mu_1 = 1 - \mu_2 - \mu_3.$$

In Table 4 we present one of these methods, called ROWDA3, having $R(\infty)=0$ (i.e. γ is root of the polynomial $\gamma^3 - 3\gamma^2 + \frac{3}{2}\gamma - \frac{1}{6}$) and a small error constant.

Proposition (5.3). *There exists no method of order 4 with $s=4$.*

Proof. For order 4 we have 13 equations to solve (see Sect. 4). After simplification, we get:

$$\mu_4 \beta_{43} \beta_{32} \beta_2 = p_6 \tag{4.12f}$$

$$\mu_4 \beta_{43} \beta_{32} \alpha_2^2 = p_{10} \tag{4.13 a}$$

$$\mu_4 \beta_{43} \alpha_3 \alpha_{32} \beta_2 = p_{12} \tag{4.13 c}$$

$$\mu_4 \beta_{43} \alpha_3 \alpha_{32} \alpha_2^2 = p_{13} \tag{4.13 d}$$

where $\alpha_i = \sum_{j=1}^{i-1} \alpha_{i,j}$ and

$$p_6 = \frac{1}{24} - \frac{\gamma}{2} + \frac{3}{2}\gamma^2 - \gamma^3$$

$$p_{10} = \gamma^3 - \frac{2}{3}\gamma^2 + \frac{\gamma}{12}$$

$$p_{12} = \gamma^3 - \frac{5}{6}\gamma^2 + \frac{\gamma}{8}$$

$$p_{13} = -\gamma^3 + \frac{\gamma^2}{4}$$

Table 4. Coefficients of ROWDA3

γ	=	0.435866521508459	
μ_1	=	0.3197278911564624	$\hat{\mu}_1 = 0.926163587124091$
μ_2	=	0.7714777906171382	$\hat{\mu}_2 = 0.073836412875909$
μ_3	=	-0.09120568177360061	$\hat{\mu}_3 = 0$
α_{21}	=	0.7	$\alpha_{31} = 0.7$
α_{32}	=	0	
γ_{21}	=	0.1685887625570998	$\gamma_{31} = 4.943922277836421$
γ_{32}	=	1	

We have:

$$\frac{\beta_2}{\alpha_2^2} = \frac{p_6}{p_{10}} = \frac{p_{12}}{p_{13}}$$

A calculation leads to the equation:

$$18\gamma^2 - 8\gamma + 1 = 0 \quad \text{unsolvable in } \mathbf{R}. \quad \text{Q.E.D.}$$

Theorem (5.4). *There exist embedded methods of order 4, convergent, with $s=5$ but only 4 evaluations of the functions f and g .*

Proof. Set $\alpha_{21}=0$ and $\beta_{43}=0$ in the equations; it is then very easy to solve them and to have a couple of free parameters to choose. Q.E.D.

Remark. We asked convergence ($\alpha=0$ is the best choice) only for the method of order 4. Unfortunately, the methods of Theorem (5.4) have a contractivity number α not equal to 0: the choice ($\alpha_{21}=0$ and $\beta_{43}=0$) forces γ to be $1/2$ or $1/6$ and for example, with $\gamma = 1/2$ we find $\alpha = 1/3$.

Table 5. A Rosenbrock method of order 4

γ	=	0.70751226521	
μ_1	=	0.2523628037277470	$\hat{\mu}_1 = 0.7747652563757017$
μ_2	=	-0.2209698738798533	$\hat{\mu}_2 = 0.003017168075271842$
μ_3	=	-0.2256411840923124	$\hat{\mu}_3 = -0.2924038105920804$
μ_4	=	0.3179133966013711	$\hat{\mu}_4 = -0.06984969876968235$
μ_5	=	0.8763348576430476	$\hat{\mu}_5 = 0.5844710849107893$
α_{21}	=	1.233311380872013	$\alpha_{31} = 0.6535453813273382$
α_{32}	=	0.2295950748229277	$\alpha_{41} = 2.681059792907162$
α_{42}	=	-1.554590259558157	$\alpha_{43} = -0.9682496302574051$
α_{51}	=	-0.6021422614217772	$\alpha_{52} = 0.2994399056322287$
α_{53}	=	0.4792338650945191	$\alpha_{54} = 0.8010415023569842$
γ_{21}	=	-1.818714325256271	$\gamma_{31} = -0.4589460040608732$
γ_{32}	=	0.3613323897595465	$\gamma_{41} = -3.424045164556574$
γ_{42}	=	1.553491448551290	$\gamma_{43} = 1.249712740807497$
γ_{51}	=	-0.2261466054228607	$\gamma_{52} = -0.3882326103473952$
γ_{53}	=	-0.3589041115714489	$\gamma_{54} = -0.01860845389367294$

Nevertheless it is possible, using Newton iterations, to find convergent embedded 5-stages methods of order 4(3) with $\alpha = 0$:

We have 19 equations to solve (13 for the method of order 4, 5 for the method of order 3 and 1 for the convergence) and 31 unknowns.

Solving these equations by Newton iterations, we obtain for example the values given in Table 5; every equation is satisfied with an error smaller than 10^{-14} .

6. Numerical Examples

The methods described in Sect. 5 have been applied to several "test-problems", and the theoretical orders actually observed.

Example:

$$\begin{cases} y' = z & y_0 = 0 \\ 0 = y^2 + z^2 - 1 & z_0 = 1 \end{cases} \quad (6.1)$$

Exact solutions: $y(t) = \sin(t)$ $z(t) = \cos(t)$

$$\frac{\partial g}{\partial z}(y_0, z_0) = 2 \cdot z_0 = 2$$

All the numerical experiments have been carried out in double precision on an Apollo DN330 computer (precision 10^{-15}).

To test the convergence of method (5.5), we integrated the problem (6.1) between $t=0$ and $t=1$ using constant stepsize $h=1/n$ (for various values of n). Let $e_y(h)$ be the error made on the y -component after n steps of length $h=1/n$. As $e_y(h) \approx C \cdot h^p$, the value

$$p_y = \log \left| \frac{e_y(h)}{e_y(h/2)} \right| / \log(2) \quad (6.2)$$

is taken as an approximation for p . Similarly $p_z = \log \left| \frac{e_z(h)}{e_z(h/2)} \right| / \log(2)$ is an approximation for p . The results are displayed in Table 6.

Table 6

h	$e_y(h)$	$e_z(h)$	p_y	p_z
$6.25 \cdot 10^{-2}$	$1.2 \cdot 10^{-5}$	$2.6 \cdot 10^{-4}$	3.85	3.65
$3.125 \cdot 10^{-2}$	$8.6 \cdot 10^{-7}$	$2.1 \cdot 10^{-5}$	3.92	3.81
$1.5625 \cdot 10^{-2}$	$5.6 \cdot 10^{-8}$	$1.5 \cdot 10^{-6}$	3.95	3.90
$7.8125 \cdot 10^{-3}$	$3.6 \cdot 10^{-9}$	10^{-7}	3.91	4.07

Acknowledgement. I wish to thank E. Hairer, G. Wanner and the members of the "Seminar on Numerical Analysis" in Geneva for many helpful remarks and stimulating discussions, the referee, P. Rentrop, for his many critical remarks, and the unknown referee. A special thanks to G. Cairns for a careful reading of the english manuscript.

References

1. Rheinboldt, W.C.: Differential-Algebraic Systems as Differential Equations on Manifolds. *Math. Comput.* **43**, 473–482 (1984)
2. Gear, C.W., Petzold, L.: ODE Methods for the Solution of Differential Algebraic Systems. *SIAM J. Numer. Anal.* **21**, 716–728 (1985)
3. Petzold, L.: Differential/Algebraic Equations are not ODE's. *SIAM J. Stat. Sci. Comput.* **3**, 367–384 (1982)
4. Petzold, L.: Order results for implicit Runge-Kutta methods applied to Differential/Algebraic systems. *SIAM J. Numer. Anal.* **23**, 837–852 (1986)
5. Petzold, L.: A description of DASSL. In: *A Differential/Algebraic System Solver*. R.S. Stepleman (ed.). Proc. IMACS Trans. on Scientific Computation **1** (1982)
6. März, R.: Multisteps methods for initial value problems in implicit differential-algebraic equations. *Numer. Math.* **12**, 107–123 (1984)
7. März, R.: On numerical integration methods for implicit ordinary differential equations and differential-algebraic equations. Proc. Kolloquium "Numerische Behandlung von Differentialgleichungen" pp. 1–15. *Wiss. Beitr. Univ. Jena*, 1983
8. Griepentrog, E., März, R.: *Differential-Algebraic Equations and Their Numerical Treatment*. Teubner Texte zur Mathematik, 88. Leipzig: Teubner 1986
9. Deuffhard, P., Hairer, E., Zugck, J.: One-step and Extrapolation Methods for Differential-Algebraic Systems. *Numer. Math.* **51**, 501–516 (1987)
10. Gallun, S., Holland, C.: Gear's procedure for the simultaneous solution of differential and algebraic equations with application to unsteady state distillation problems. *Comp. Chem. Eng.* **6**, 231–244 (1982)
11. Feng, A., Holland, C., Gallun, S.: Development and comparison of generalized semi-implicit Runge-Kutta method with Gear's method for systems of coupled differential and algebraic equations. *Comp. Chem. Eng.* **8**, 51–59 (1984)
12. Miranker, W.L.: *Numerical Methods for Stiff Equations and Singular Perturbation Problems*. Dordrecht: Reidel 1981
13. Kaps, P., Wanner, G.: A Study of Rosenbrock-Type Methods and High Order. *Numer. Math.* **38**, 279–298 (1981)
14. Kaps, P., Rentrop, P.: Generalized Runge-Kutta Methods of order four with step-size control for stiff ODE's. *Numer. Math.* **33**, 55–68 (1979)
15. Verwer, J.G.: Instructive experiments with some Runge-Kutta-Rosenbrock methods. *Comp. Comp. Math. Appl.* **8**, 217–229 (1982)
16. Hairer, E., Norsett, S., Wanner, G.: *Solving Ordinary Differential Equations*, Vol. 1. Berlin, Heidelberg, New York, Tokyo: Springer 1986

Received May 16, 1986 / June 30, 1987