SIAM J. NUMER. ANAL.
Vol. 30, No. 2, pp. 538–552, April 1993

011

HALF-EXPLICIT RUNGE–KUTTA METHODS FOR DIFFERENTIAL-ALGEBRAIC SYSTEMS OF INDEX 2*

V. BRASEY† AND E. HAIRER†

Abstract. Half-explicit Runge–Kutta methods for differential-algebraic problems of index 2 are investigated. It is shown how the arising order conditions can be solved and a particular method of order 4 is constructed. In addition, this paper simplifies the known convergence theory for such methods and demonstrates by numerical experiments their excellent properties when applied to constrained multibody systems.

Key words. differential-algebraic systems, Runge–Kutta methods, order conditions, constrained multibody systems

AMS subject classification. 65L05

1. Introduction. We consider differential-algebraic systems of the form

$$y' = f(y,z), \qquad 0 = g(y). \tag{1.1}$$

We assume that f and g are sufficiently differentiable and that

$$g_y(y)f_z(y,z) \quad \text{is nonsingular} \tag{1.2}$$

in a neighbourhood of the solution, so that the problem has index 2 (the subscripts in g_y and f_z indicate partial derivatives). The initial values x_0, y_0, z_0 are required to be consistent, which means that

$$g(y_0) = 0, \qquad g_y(y_0)f(y_0,z_0) = 0. \tag{1.3}$$

Whenever an initial value y_0 satisfying $g(y_0) = 0$ has been specified, the second equation of (1.3) determines a locally unique value for z_0. This is a consequence of (1.2) and the implicit function theorem. Then (1.1) possesses a unique solution.

Problems of the form (1.1) are frequently encountered in practice. An important class of such problems are multibody systems with constraints on the velocity level (see, e.g., [7], [6], [5], [13]). Problem (1.1) has also the typical structure of a control problem where the z-variable acts as control parameter which forces the solution of the differential equation to stay on the manifold $g(y) = 0$ (for concrete examples see [1] or [8]). Further, differential equations with discontinuous right-hand side may lead to problems of the type (1.1) (see the example of §6).

In the last years, much research has been devoted to the development of numerical methods for the differential-algebraic problem (1.1). Convergence results for BDF methods have been given in [6], [11], [2], [14]; implicit Runge–Kutta methods have been considered in [3] and [8]; Lubich [12] has recently proposed an extrapolation method based on a half-explicit mid-point rule.

The application of explicit Runge–Kutta methods to problems of the form (1.1) is proposed in [8]. There, only the differential equation is treated in an explicit manner; the implicit equation $g(y) = 0$ has still to be solved (hence the name "half-explicit" methods). The advantages of these methods are similar to those of explicit

*Received by the editors January 30, 1991; accepted for publication March 26, 1992.

†Université de Genève, 2-4, rue du Lièvre, Case postale 240, 1211 Genève 24, Switzerland.

Runge–Kutta methods for (nonstiff) ordinary differential equations. They are easy to implement and do not need a special starting procedure.

The aim of this paper is to search for efficient methods of this type. In §2 we give the definition of half-explicit methods and discuss their convergence. In §3 we present the order conditions of [8] and we give some preliminary results on the existence of half-explicit methods of a certain order. In §4 we then construct particular methods of order 4. The application to mechanical multibody systems is discussed in §5 and the last section presents some numerical experiments as well as numerical comparisons with other codes.

The results of this article extend in a natural way to problems of the form

$$y' = f(y,z,u), \qquad 0 = k(y,z,u), \qquad 0 = g(y), \tag{1.4}$$

where the matrix $k_u(y,z,u)$ is assumed to be nonsingular. By the implicit function theorem the equation $0 = k(y,z,u)$ can be solved for u. This inserted into the first equation of (1.4) gives a problem of type (1.1). The numerical methods considered in this article are invariant under this transformation.

2. Half-explicit Runge–Kutta methods. For the numerical solution of the differential-algebraic system (1.1), we consider the following method (half-explicit Runge–Kutta method):

$$Y_i = y_0 + h\sum_{j=1}^{i-1} a_{ij} f(Y_j, Z_j), \qquad i = 1,\dots,s, \tag{2.1a}$$

$$0 = g(Y_i), \tag{2.1b}$$

$$y_1 = y_0 + h\sum_{i=1}^{s} b_i f(Y_i, Z_i), \tag{2.1c}$$

$$0 = g(y_1). \tag{2.1d}$$

The initial value is assumed to satisfy $g(y_0) = 0$, h is the step size, and y_1 the approximation to the solution at $x_0 + h$.

The algorithm is applied as follows: for $i = 1$, (2.1a) defines $Y_1 = y_0$ and (2.1b) is automatically satisfied. If we insert Y_2 from (2.1a) into (2.1b), we obtain

$$0 = g\big(y_0 + ha_{21} f(Y_1, Z_1)\big)$$

which is a nonlinear equation for Z_1. Once Z_1 is computed, (2.1a) constitutes an explicit formula for Y_2 and the procedure can be repeated for the further stages. In the ith stage, one has to solve the nonlinear system

$$F(Z_i) = 0 \quad \text{with } F(Z_i) = g\Big(y_0 + h\sum_{j=1}^{i} a_{i+1,j} f(Y_j, Z_j)\Big) \tag{2.2}$$

for the unknown Z_i. This can be done by simplified Newton iterations (see the proof of Lemma 1 below).

Methods of the type (2.1) have been proposed in [8]. The convergence analysis given there is based on results for general implicit Runge–Kutta methods. The following direct proof is more simple in its structure.

LEMMA 1 (existence of numerical solution). *Let* (1.2) *and* (1.3) *be satisfied and assume that for the coefficients of the Runge–Kutta scheme*

$$a_{i,i-1} \neq 0 \quad \textit{for } i = 2, \dots, s \quad \textit{and} \quad b_s \neq 0. \tag{2.3}$$

Then, for sufficiently small h, the numerical solution of (2.1) *exists and is* (*locally*) *unique.*

Proof. By induction on i we shall show that Newton's method applied to (2.2) converges, when $Z_i^{(0)} = z_0$ is taken as starting value. Moreover we shall show that

$$Z_i - z_0 = \mathcal{O}(h). \tag{2.4}$$

Assuming that $Z_1, \dots, Z_{i-1}$ are $\mathcal{O}(h)$-approximations to z_0, one easily verifies for $F(Z_i)$, given by (2.2), that

$$\|F'(Z_i)^{-1}\| \leq \frac{C_1}{h}, \qquad \|F''(Z_i)\| \leq C_2 h, \qquad \|F'(z_0)^{-1}F(z_0)\| \leq C_0 h$$

for $h \leq h_0$ and for Z_i in a sufficiently small neighbourhood of z_0. Hence, the theorem of Newton–Kantorovich [15] implies convergence of Newton's method to a unique solution of (2.2) which satisfies (2.4). □

LEMMA 2 (error propagation). *Assume that* (1.2) *holds and let y_0 and $\widehat{y}_0$ be two sufficiently close initial values which both satisfy* (1.3). *Then the corresponding numerical solutions, defined by* (2.1), *satisfy*

$$\|y_1 - \widehat{y}_1\| \leq (1 + Ch)\|y_0 - \widehat{y}_0\| \tag{2.5}$$

if h is sufficiently small.

Proof. Suppose, by induction on i, that $Y_1, Z_1, \dots, Y_{i-1}, Z_{i-1}$ depend smoothly on y_0. From (2.1a) it then follows that the same is true for Y_i. In order to prove the statement for Z_i, we write (2.2) in the equivalent form

$$\begin{aligned} 0 &= \frac{1}{h}\left(g\left(y_0 + h \sum_{j=1}^{i} a_{i+1,j} f(Y_j, Z_j) \right) - g(y_0) \right) \\ &= \int_0^1 g_y\left(y_0 + \tau h \sum_{j=1}^{i} a_{i+1,j} f(Y_j, Z_j) \right) d\tau \cdot \sum_{j=1}^{i} a_{i+1,j} f(Y_j, Z_j). \end{aligned} \tag{2.6}$$

The derivative of (2.6) with respect to Z_i is given by

$$a_{i+1,i} \cdot g_y(y_0) f_z(Y_i, Z_i) + \mathcal{O}(h)$$

and by (1.2) this matrix is invertible for sufficiently small h. Hence the implicit function theorem implies that also Z_i depends smoothly on y_0. Equation (2.1c) together with the Lipschitz continuity of f then yields the desired estimate. □

The local error of method (2.1) is defined by

$$\delta y_h(x) = y_1 - y(x + h), \tag{2.7}$$

where y_1 is the numerical solution of (2.1) with initial value $y_0 = y(x)$.

THEOREM 3 (convergence). *Suppose that* (1.2) *holds in a neighbourhood of the solution* $(y(x), z(x))$ *of* (1.1) *and that the initial values are consistent. If the coefficients of the half-explicit Runge–Kutta method satisfy* (2.3) *and if the local error satisfies*

$$\delta y_h(x) = \mathcal{O}(h^{p+1}), \tag{2.8}$$

then the method is convergent of order p, *i.e.*,

$$y_n - y(x_n) = \mathcal{O}(h^p) \quad \textit{for } x_n - x_0 = nh \leq \text{Const.}$$

Proof. Lemma 2 states that the error propagation behaves in exactly the same way as for one-step methods applied to nonstiff ordinary differential equations. Therefore, standard techniques (see, e.g., [9, Theorem II.3.4]) yield the convergence result. □

3. Order conditions. This section is devoted to the verification of (2.8). In principle, this can be done as follows: expand the exact solution $y(x_0 + h)$ as well as the numerical solution y_1 (considered as a function of h) into Taylor series and compare the coefficients of h^q for $q \leq p$. Its realization, which is intricate without introducing suitable notations (trees, elementary differentials), is given in [8] (see also [10]). The result is presented in Table 1. There, the coefficients ω_{ij} are the entries of the matrix

$$\begin{pmatrix} \omega_{ij} \end{pmatrix} = \begin{pmatrix} a_{21} & & & & \\ a_{31} & a_{32} & & & \\ \vdots & \vdots & \ddots & & \\ a_{s1} & a_{s2} & \dots & a_{s,s-1} & \\ b_1 & b_2 & \dots & b_{s-1} & b_s \end{pmatrix}^{-1}, \tag{3.1}$$

which exists by (2.3). We further use the standard notation

$$\sum_{j=1}^{i-1} a_{ij} = c_i, \qquad c_1 = 0, \tag{3.2}$$

and we also write

$$a_{s+1,i} = b_i, \qquad i = 1, \dots, s, \qquad c_{s+1} = 1. \tag{3.3}$$

We observe that one condition is required for a method of order 1, two conditions for order 2, six conditions for order 3, and 20 conditions for a method of order 4.

The order conditions are connected to the corresponding trees (see Table 1) by the following algorithm.

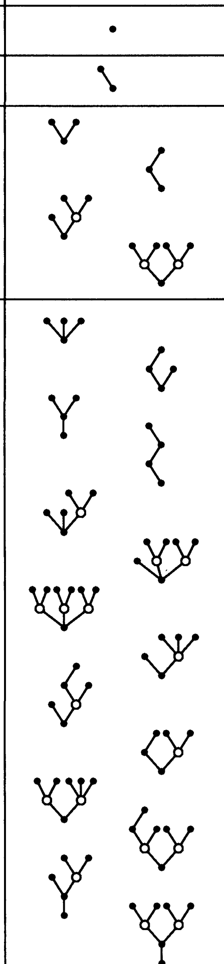
ALGORITHM. *Forming the order condition for a given tree.* Attach to each vertex one summation index $i, j, \dots$. Then the left-hand side of the order condition is a sum over all indices of a product with factors

b_i	if "i" is the index of the root (lowest vertex);
a_{ij}	if the meagre vertex "j" lies directly above the meagre vertex "i";
$a_{i+1,j}$	if the meagre vertex "j" lies directly above the fat vertex "i";
ω_{ij}	if the fat vertex "j" lies directly above the meagre vertex "i".

The right-hand side of the order condition is a rational number which is the product over all indices of the factor

$1/r$	if the vertex "i" is meagre;
$r+1$	if the vertex "i" is fat.

TABLE 1
Order conditions up to order 4.

Nr.	tree	order	order condition
1		1	$\sum b_i = 1$
2		2	$\sum b_i c_i = \frac{1}{2}$
3		3	$\sum b_i c_i^2 = \frac{1}{3}$
4		3	$\sum b_i a_{ij} c_j = \frac{1}{6}$
5		3	$\sum b_i c_i \omega_{ij} c_{j+1}^2 = \frac{2}{3}$
6		3	$\sum b_i \omega_{ij} c_{j+1}^2 \omega_{ik} c_{k+1}^2 = \frac{4}{3}$
7		4	$\sum b_i c_i^3 = \frac{1}{4}$
8		4	$\sum b_i c_i a_{ij} c_j = \frac{1}{8}$
9		4	$\sum b_i a_{ij} c_j^2 = \frac{1}{12}$
10		4	$\sum b_i a_{ij} a_{jk} c_k = \frac{1}{24}$
11		4	$\sum b_i c_i^2 \omega_{ij} c_{j+1}^2 = \frac{1}{2}$
12		4	$\sum b_i c_i \omega_{ij} c_{j+1}^2 \omega_{ik} c_{k+1}^2 = 1$
13		4	$\sum b_i \omega_{ij} c_{j+1}^2 \omega_{ik} c_{k+1}^2 \omega_{il} c_{l+1}^2 = 2$
14		4	$\sum b_i c_i \omega_{ij} c_{j+1}^3 = \frac{3}{4}$
15		4	$\sum b_i c_i \omega_{ij} c_{j+1} a_{j+1,k} c_k = \frac{3}{8}$
16		4	$\sum b_i a_{ij} c_j \omega_{ij} c_{j+1}^2 = \frac{1}{4}$
17		4	$\sum b_i \omega_{ij} c_{j+1}^2 \omega_{ik} c_{k+1}^3 = \frac{3}{2}$
18		4	$\sum b_i \omega_{ij} c_{j+1}^2 \omega_{ik} c_{k+1} a_{k+1,l} c_l = \frac{3}{4}$
19		4	$\sum b_i a_{ij} c_j \omega_{jk} c_{k+1}^2 = \frac{1}{6}$
20		4	$\sum b_i a_{ij} \omega_{jk} c_{k+1}^2 \omega_{jl} c_{l+1}^2 = \frac{1}{3}$

Here r denotes the difference of the meagre and fat vertices lying above "i" ("i" included).

Remark. For the important special case where the function f in (5.1) is linear in z, the order conditions with number (6), (12), (13), (17), (18), and (20) in Table 1 need not be considered, because their elementary differentials vanish (see [8, p. 57]).

PROPOSITION 4. *There exists a unique half-explicit method* (2.1) *of order* 3 *with* $s = 3$ *stages. Its coefficients are given in Table* 2.

Proof. There exists a two-parameter family of explicit Runge–Kutta methods of "classical" order 3 (i.e., satisfying (1), (2), (3), (4) of Table 1) with $s = 3$ (see e.g., [9, p. 141]). The coefficients of this method inserted into the order conditions (5) and (6) then lead (after tedious algebraic manipulations; we have used *Mathematica* [17]) to the unique conditions $c_2 = \frac{1}{3}$ and $c_3 = 1$. $\square$

TABLE 2
Method (2.1) *of order* 3.

0			
$\frac{1}{3}$	$\frac{1}{3}$		
1	-1	2	
	0	$\frac{3}{4}$	$\frac{1}{4}$

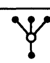
PROPOSITION 5. *There is no half-explicit method* (2.1) *of order* 4 *with* $s = 4$ *stages.*

Proof. The solution of (1), (2), (3), (4), (7)–(10) in Table 1 (classical order conditions for ordinary differential equations) represents a two-parameter family (c_2 and c_3 are free parameters) of explicit Runge–Kutta methods which satisfies $b_3 b_4 c_2(1-c_3) \neq 0$ (see, e.g., [9, p. 136]). A direct computation (again using *Mathematica* [17]) shows that

$$\sum_{i,j} b_i c_i^2 \omega_{ij} c_{j+1}^2 - \frac{1}{2} = \frac{1}{6} c_2 (1 - c_3).$$

Hence the order condition (11) cannot be satisfied. □

TABLE 3
Additional conditions for an embedded method.

Nr.	tree	order	order condition
1		2	$\sum \widehat{b}_i \omega_{ij} c_{j+1}^2 = 1$
2		3	$\sum \widehat{b}_i \omega_{ij} c_{j+1}^3 = 1$
3		3	$\sum \widehat{b}_i \omega_{ij} c_{j+1} a_{j+1,k} c_k = \frac{1}{2}$

Additional conditions for an embedded method and for dense output. In view of an algorithmic implementation of method (2.1) (step size selection), one is interested in a second approximation of $y(x_0 + h)$ which is for instance of the form

$$\widehat{y}_1 = y_0 + h \sum_{i=1}^{s} \widehat{b}_i f(Y_i, Z_i) \tag{3.4}$$

with Y_i, Z_i given by (2.1). For this approximation, we have in general $g(\widehat{y}_1) \neq 0$. Therefore also the error terms transversal to the manifold $g(y) = 0$ have to be considered. They lead to the conditions of Table 3 (for details we again refer to [8, p. 63]) and have to be appended to those of Table 1 (with b_i replaced by $\widehat{b}_i$).

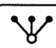
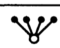
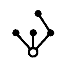
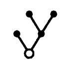
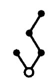
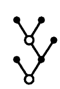
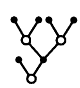
The same arguments are also valid for a dense output defined by

$$y_\theta = y_0 + h \sum_{i=1}^{s} b_i(\theta) f(Y_i, Z_i). \tag{3.5}$$

For obtaining $y_\theta - y(x_0 + \theta h) = \mathcal{O}(h^{q+1})$ the conditions of Tables 1 and 3 have to be satisfied up to order q. Obviously, one has to replace b_i (respectively, $\widehat{b}_i$) by $b_i(\theta)$ and in the right-hand side of the order condition one has to add the factor θ^r where r is the order of the corresponding condition (third column of Table 1).

Approximation of the z-component. For the application of method (2.1), one needs only the knowledge of the initial value y_0 and one gets an approximation y_1 to

TABLE 4
Additional order conditions for the z-component.

Nr.	tree	order	order condition
1		1	$\sum \omega_{sj} c_{j+1}^2 = 2$
2		2	$\sum \omega_{sj} c_{j+1}^3 = 3$
3		2	$\sum \omega_{sj} c_{j+1} a_{j+1,k} c_k = \frac{3}{2}$
4		3	$\sum \omega_{sj} c_{j+1}^4 = 4$
5		3	$\sum \omega_{sj} c_{j+1}^2 a_{j+1,k} c_k = 2$
6		3	$\sum \omega_{sj} a_{j+1,k} c_k a_{j+1,l} c_l = 1$
7		3	$\sum \omega_{sj} c_{j+1} a_{j+1,k} c_k^2 = \frac{4}{3}$
8		3	$\sum \omega_{sj} c_{j+1} a_{j+1,k} a_{kl} c_l = \frac{2}{3}$
9		3	$\sum \omega_{sj} c_{j+1} a_{j+1,k} c_k \omega_{kl} c_{l+1}^2 = \frac{8}{3}$
10		3	$\sum \omega_{sj} c_{j+1} a_{j+1,k} \omega_{kl} c_{l+1}^2 \omega_{km} c_{m+1}^2 = \frac{16}{3}$

$y(x_0 + h)$. If one is also interested in an approximation for the z-component, one can pursue several possibilities.

The most accurate one is certainly the computation of z_1 from the nonlinear equation (compare (1.3))

$$g_y(y_1) f(y_1, z_1) = 0. \tag{3.6}$$

Due to (1.2) and the implicit function theorem, we get in this way the same order of accuracy for z_1 as for y_1.

In [8] it is proposed to require $c_s = 1$ and to take $z_1 = Z_s$ as approximation to $z(x_0 + h)$. The order conditions for this choice of z_1 can be deduced from the results of [8]. It holds that $Z_s - z(x_0 + h) = \mathcal{O}(h^{r+1})$ if and only if $Y_s - y(x_0 + h) = \mathcal{O}(h^{r+1})$ and the conditions of Table 4 are satisfied up to order r. Since the formula for z_1 is only used locally, the value of r does not influence the convergence behaviour of the method.

4. Construction of methods of order 4. Our next task is to solve the system of equations imposed by the order conditions. For this aim, we use the simplifying assumptions

$$\text{C(2)} \qquad \sum_{j=1}^{i-1} a_{ij} c_j = \frac{c_i^2}{2}, \qquad i = 3, \ldots, s, \qquad b_2 = 0,$$

and

$$\text{D(1)} \qquad \sum_{i=j+1}^{s} b_i a_{ij} = b_j (1 - c_j), \qquad j = 1, \ldots, s.$$

Condition C(2) cannot be satisfied for $i = 2$. Indeed, we would have $c_2 = 0$ and the method would be reducible.

LEMMA 6. *Assumption* C(2) *implies that the pairs of conditions* (3)–(4), (7)–(8), *and* (11)–(16) (*see Table* 1) *are equivalent. Hence, the conditions corresponding to trees of type* I (*Fig.* 1) *are automatically satisfied.*

Proof. By C(2) we have

$$\sum_{i,j} b_i a_{ij} c_j = \sum_i b_i \frac{c_i^2}{2} = \frac{1}{2}\sum_i b_i c_i^2$$

which implies that the conditions (3) and (4) of Table 1 are equivalent. The other equivalences are seen similarly. □

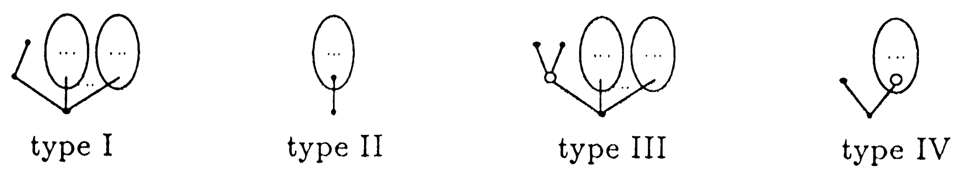


FIG. 1. *Simplification of order conditions.*

LEMMA 7. *A consequence of* D(1) *is that the conditions* (2) *and* (3) *of Table* 1 *imply* (4). *Further* (3), (7) ⇒ (9); (4), (8) ⇒ (10); (5), (11) ⇒ (19); *and* (6), (12) ⇒ (20). *Hence the conditions corresponding to trees of type* II (*Fig.* 1) *need not be considered.*

Proof. We show the first implication. From D(1) we have

$$\sum_{i,j} b_i a_{ij} c_j = \sum_j b_j(1-c_j)c_j = \sum_j b_j c_j - \sum_j b_j c_j^2$$

and the statement follows from Table 1. □

In matrix notation, the assumption C(2) together with the condition (2) of Table 1 can be written as

$$\begin{pmatrix} a_{21} & & & & \\ a_{31} & a_{32} & & & \\ \vdots & \vdots & \ddots & & \\ a_{s1} & a_{s2} & \dots & a_{s,s-1} & \\ b_1 & b_2 & \dots & b_{s-1} & b_s \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_{s-1} \\ c_s \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 0 \\ c_3^2 \\ \vdots \\ c_s^2 \\ c_{s+1}^2 \end{pmatrix}.$$

Multiplying with the inverse of the occurring matrix, which is (3.1), we obtain

$$\text{C(2)I} \qquad \sum_{j=1}^{i} \omega_{ij} c_{j+1}^2 = 2c_i + \omega_{i1} c_2^2, \qquad i = 1, \dots, s.$$

In a similar way, we see that assumption D(1) is equivalent to

$$\text{D(1)I} \qquad \sum_{i=j}^{s} b_i c_i \omega_{ij} = \sum_{i=j}^{s} b_i \omega_{ij} - b_{j+1} = \delta_{sj} - b_{j+1}, \qquad j = 1, \dots, s$$

with the conventions (3.3), $b_{s+1} = 0$ and

$$\delta_{sj} = 0 \quad \text{for} \quad j < s \quad \text{and} \quad \delta_{ss} = 1.$$

These relations simplify the order conditions as shown in the following lemma.

LEMMA 8. *If in addition to* C(2)I *we assume*

$$b_i\omega_{i1} = 0 \qquad \text{for } i = 1, \ldots, s, \tag{4.1}$$

then the following pairs of conditions from Table 1 *are equivalent:* (3)–(5), (3)–(6), (7)–(11), (7)–(12), (7)–(13), (8)–(16), (14)–(17), (15)–(18). *Hence, the order conditions corresponding to trees of type* III (*Fig.* 1) *need not be considered.*

Proof. Under the assumption (4.1) condition C(2)I becomes

$$b_i \sum_{j=1}^{i} \omega_{ij} c_{j+1}^2 = 2b_i c_i, \qquad i = 1, \ldots, s,$$

and the statements follow from Table 1. □

LEMMA 9. *The relation* D(1)I *implies that the pairs of conditions* (3)–(5) *and* (7)–(14) *from Table* 1 *are equivalent. Further the conditions* (2) *and* (8) *imply* (15). *Hence, the conditions corresponding to the trees of type* IV (*Fig.* 1) *need not be considered.*

Proof. With the use of D(1)I we have

$$\sum_{i,j} b_i c_i \omega_{ij} c_{j+1}^2 = c_{s+1}^2 - \sum_i b_i c_i^2$$

and the equivalence of the conditions (3) and (5) follows from (3.3) and Table 1. The other two statements follow in a similar way. □

We are now able to prove the main result of this section.

THEOREM 10 (reduced system). *The conditions* (3.2), C(2), D(1), (4.1) *and*

$$\sum_{i=1}^{s} b_i c_i^{q-1} = \frac{1}{q}, \qquad q = 1, \ldots, 4 \tag{4.2}$$

imply that the method (2.1) *has order* 4.

Proof. Relations (4.2) are equivalent to the order conditions (1), (2), (3), and (7) of Table 1 and, by the foregoing lemmas, all other order conditions are also satisfied. □

ALGORITHM. *Construction of a half-explicit method of order* 4 *with* $s = 5$ *stages.* In order to solve the reduced system of Theorem 10, we proceed as follows.

Step 1. Set $c_1 = 0$, $c_5 = 1$, $c_4 = (1 - 2c_3)/(2 - 6c_3)$; take c_2 and c_3 as free parameters.

Step 2. Set $b_1 = 0$, $b_2 = 0$, compute b_3, b_4, b_5 from (4.2). This is possible due to the above relation between c_3 and c_4.

Step 3. Obtain a_{32} from C(2) with $i = 3$; a_{54} from D(1) with $j = 4$; $a_{21} = c_2$, $a_{31} = c_3 - a_{32}$ from (3.2), and ω_{11}, ω_{21}, from the definition of (ω_{i1}), which reads

$$a_{21}\omega_{11} = 1, \qquad \sum_{j=1}^{i} a_{i+1,j}\omega_{j1} = 0, \qquad j = 2, \ldots, s. \tag{4.3}$$

In order to satisfy (4.1), set $\omega_{31} = 0$, $\omega_{41} = 0$, and $\omega_{51} = 0$.

Step 4. Use (3.2), C(2), and (4.3) to obtain the linear systems

$$
\begin{aligned}
&\begin{pmatrix} 1 & 1 & 1 \\ 0 & c_2 & c_3 \\ \omega_{11} & \omega_{21} & 0 \end{pmatrix}\begin{pmatrix} a_{41} \\ a_{42} \\ a_{43} \end{pmatrix} = \begin{pmatrix} c_4 \\ c_4^2/2 \\ 0 \end{pmatrix}, \\
&\begin{pmatrix} 1 & 1 & 1 \\ 0 & c_2 & c_3 \\ \omega_{11} & \omega_{21} & 0 \end{pmatrix}\begin{pmatrix} a_{51} \\ a_{52} \\ a_{53} \end{pmatrix} = \begin{pmatrix} 1 \\ \frac{1}{2} \\ 0 \end{pmatrix} - a_{54}\begin{pmatrix} 1 \\ c_4 \\ 0 \end{pmatrix}.
\end{aligned}
\tag{4.4}
$$

We assume that c_2 and c_3 are chosen such that the occuring matrix is invertible, and so obtain the remaining coefficients by solving the linear systems (4.4).

Remark. The above construction assures D(1) for the moment only for $j = 4$ (in step 3) and $j = 5$ (due to $c_s = 1$). The remaining relations (for $j = 1, 2, 3$) are automatically satisfied. To show this, we define

$$d_j = \sum_{i=j+1}^{s} b_i a_{ij} - b_j(1 - c_j) \quad \text{for } j = 1, 2, 3.$$

Then conditions (4.2) and (3.2) imply that

$$\sum_j d_j = \sum_{i,j} b_i a_{ij} - \sum_j b_j + \sum_j b_j c_j = 0.$$

Similarly the relations C(2), (4.2), (4.3), and $b_1 = 0$ imply

$$\sum_j d_j c_j = 0 \quad \text{and} \quad \sum_j d_j \omega_{j1} = 0.$$

If the matrix in (4.4) is invertible, this implies $d_j = 0$ also for $j = 1, 2, 3$.

Final choice of the method. We choose for c_3 and c_4 the Radau quadrature values with the aim of satisfying the additional fifth order condition $\sum_i b_i c_i^4 = \frac{1}{5}$. The value of c_2 (fixed as $c_2 = \frac{3}{10}$) has nearly no influence on the performance of the method. The coefficients resulting with the above procedure are listed in Table 5. They are used in the code HEM4 (half-explicit method of order 4) to be described in §6.

TABLE 5
Coefficients of HEM4 (*half-explicit method of order* 4).

0					
$\frac{3}{10}$	$\frac{3}{10}$				
$\frac{4-\sqrt{6}}{10}$	$\frac{1+\sqrt{6}}{30}$	$\frac{11-4\sqrt{6}}{30}$			
$\frac{4+\sqrt{6}}{10}$	$\frac{-79-31\sqrt{6}}{150}$	$\frac{-1-4\sqrt{6}}{30}$	$\frac{24+11\sqrt{6}}{25}$		
1	$\frac{14+5\sqrt{6}}{6}$	$\frac{-8+7\sqrt{6}}{6}$	$\frac{-9-7\sqrt{6}}{4}$	$\frac{9-\sqrt{6}}{4}$	
	0	0	$\frac{16-\sqrt{6}}{36}$	$\frac{16+\sqrt{6}}{36}$	$\frac{1}{9}$

5. Application to multibody systems. An important class of differential-algebraic systems are constrained multibody systems. We shall see in this section that for such problems the half-explicit method (2.1) can be implemented in such way that only *linear* systems have to be solved.

Consider the constrained mechanical system (for a detailed discussion, see [7], [13], [5], [6])

$$
\begin{aligned}
&(5.1a) \qquad & q' &= v, \\
&(5.1b) \qquad & M(q,t)v' &= f(q,v,t) - G^T(q,t)\lambda, \\
&(5.1c) \qquad & 0 &= g(q,t),
\end{aligned}
$$

where $q \in \mathbb{R}^n$ denotes the generalized coordinates of the system, v its velocity, $\lambda \in \mathbb{R}^m$ the Lagrange multipliers, M the generalized mass matrix, and

$$
G(q,t) = \frac{\partial g}{\partial q}(q,t). \tag{5.2}
$$

We assume for the moment that M is positive definite and that G has full rank. This implies that

$$
\begin{pmatrix} M & G^T \\ G & 0 \end{pmatrix} \quad \text{is invertible.} \tag{5.3}
$$

System (5.1) is of index 3. In order to get a system of type (1.1) we differentiate (5.1c) and obtain

$$
0 = G(q,t)v + \frac{\partial g}{\partial t}(q,t). \tag{5.1d}
$$

Application of method (2.1) to the system (5.1a,b,d) yields

$$
\begin{aligned}
&(5.4a) \qquad & Q_i' &= V_i, \\
&(5.4b) \qquad & M_i V_i' &= f_i - G_i^T \Lambda_i, \\
&(5.4c) \qquad & 0 &= G_i V_i + g_{ti},
\end{aligned}
$$

with

$$
Q_i = q_0 + h\sum_{j=1}^{i-1} a_{ij} Q_j' = q_0 + h\sum_{j=1}^{i-1} a_{ij} V_j, \qquad V_i = v_0 + h\sum_{j=1}^{i-1} a_{ij} V_j', \tag{5.4d}
$$

and the simplified notation

$$
\begin{aligned}
M_i &= M(Q_i, t_0 + c_i h), \qquad & f_i &= f(Q_i, V_i, t_0 + c_i h), \\
G_i &= G(Q_i, t_0 + c_i h), \qquad & g_{ti} &= \frac{\partial g}{\partial t}(Q_i, t_0 + c_i h).
\end{aligned}
$$

For $i = 1$, equations (5.4d) give $Q_1 = q_0$ and $V_1 = v_0$; then, assuming the initial values to be consistent, the constraint (5.4c) is automatically satisfied for $i = 1$. Relation (5.4d) now defines $Q_2 = q_0 + ha_{21}V_1$. We next insert $V_2 = v_0 + ha_{21}V_1'$ into (5.4c) and together with (5.4b) we obtain a linear system for V_1' and Λ_1 which can be solved. Proceeding in this way we get in the ith stage the linear system

$$
\begin{pmatrix} M_i & G_i^T \\ G_{i+1} & 0 \end{pmatrix} \begin{pmatrix} V_i' \\ \Lambda_i \end{pmatrix} = \begin{pmatrix} f_i \\ r_i \end{pmatrix}, \tag{5.5}
$$

where r_i denotes some known expression. All other computations are explicit.

The above algorithm does not require the positive definiteness of M. It can thus be applied to formulations of constrained mechanical systems (5.1) where M is singular but (5.3) still holds (see [13] for a discussion of such approaches). The remark at the end of §1 justifies this application.

6. Numerical experiments.

6.1. Step size selection. We found it important that the embedded solution $\widehat{y}_1$ also satisfies the algebraic relation of (1.1). Without an additional function evaluation, this is only possible if we take one of the internal stages as embedded solution. For the method of Table 5 we have $c_5 = 1$ and it is natural to put $\widehat{y}_1 = Y_5$, which is a second-order approximation to the solution so that

$$\text{err} := \|y_1 - \widehat{y}_1\| = \mathcal{O}(h^3). \tag{6.1}$$

This suggests the step size strategy

$$h_{\text{new}} = h_{\text{old}} \cdot \sqrt[3]{\frac{\text{tol}}{\text{err}}}, \tag{6.2}$$

where, as usual, a safety factor has to be added and a restriction on the change of the step size has to be imposed.

6.2. Numerical comparisons. The resulting code, called HEM4, has been implemented and tested on several problems of the form (1.1). Let us present its results for two mechanical problems. The first one is the simple pendulum in Cartesian coordinates (see, e.g., [8, p. 118]) on the interval $[0, 10]$, and the second one is the seven-body mechanism, used as a test problem in [16] and described in detail in [10, §VI.9]. As suggested by [13] we considered this problem on the interval $[0, 0.025]$. Figure 2 shows the work precision diagrams of the numerical results. We have applied the code with many different tolerances between 10^{-1} and 10^{-8} and then we have plotted the computer times in seconds (on an DN4000 Apollo work station) against the global error at the end of the integration interval. Since we have used logarithmic scales in both directions, the resulting curve is ideally a straight line whose slope indicates the effective order of the method. For a comparison we have included in Fig. 2 the analoguous results of Lubich's extrapolation code MEXX22 (see [13]).

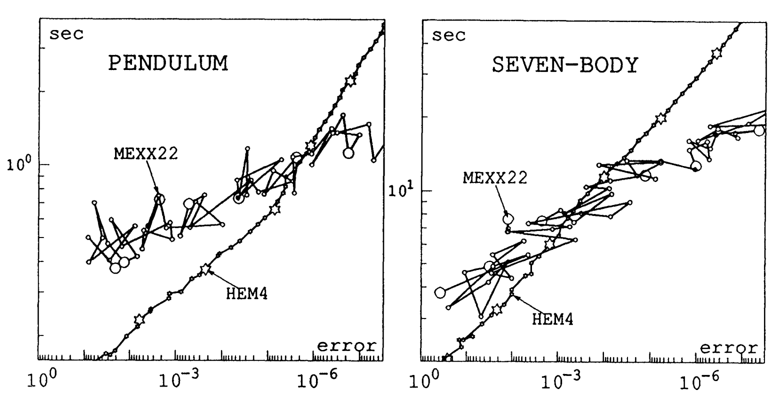


FIG. 2 *Work precision diagrams.*

6.3. Movement of the scraped string of a violin. Among the many applications of problems (1.1), let us demonstrate how differential equations with a discontinuous right-hand side can lead to differential-algebraic problems of index 2. Inspired by the article [4] we consider the problem

$$\frac{\partial^2 u}{\partial t^2} - \frac{\partial^2 u}{\partial x^2} = -\rho_s \chi(x) \cdot \operatorname{sign}\left(\frac{\partial u}{\partial t} - v\right) \quad \text{if} \quad \frac{\partial u}{\partial t} \neq v, \tag{6.3a}$$

$$\left|\frac{\partial^2 u}{\partial t^2} - \frac{\partial^2 u}{\partial x^2}\right| \leq \rho_a \chi(x) \quad \text{if} \quad \frac{\partial u}{\partial t} = v, \tag{6.3b}$$

$$u(0,t) = u(1,t) = 0, \qquad u(x,0) = \frac{\partial u}{\partial t}(x,0) = 0, \tag{6.3c}$$

where $x \in [0,1]$, $t \geq 0$ and

$$\chi(x) = \begin{cases} 1 & \text{if } |x - 1/2| \leq 1/40, \\ 0 & \text{else.} \end{cases}$$

This models the movement of the string of a violin, which is scraped by a violin bow with constant velocity v. The parameters ρ_s and ρ_a are coefficients proportional to the shearing force and to the adhesive force, respectively. We have fixed them as

$$v = 0.3, \qquad \rho_s = 12, \qquad \rho_a = 16. \tag{6.4}$$

We consider an equidistant grid $x_i = i/N$ with $N = 20$ and discretize the problem by the method of lines. This yields

$$u_i'' - N^2\big(u_{i-1} - 2u_i + u_{i+1}\big) = \begin{cases} -\rho_s z & \text{if } i = \frac{N}{2}, \\ 0 & \text{else.} \end{cases} \tag{6.5}$$

Whenever $u'_{N/2} \neq v$ we put

$$z = \operatorname{sign}(u'_{N/2} - v), \tag{6.6}$$

and together with

$$u_0(t) = u_N(t) = 0, \qquad u_i(0) = u_i'(0) = 0 \tag{6.7}$$

(6.5), with z inserted from (6.6), constitute an initial value problem of ordinary differential equations which can be solved without any difficulties. If at a time instant t_1 the value of $u'_{N/2}(t_1)$ becomes equal to v, then we switch from problem (6.3a) to (6.3b). This means that we consider z of (6.5) as a control variable which forces the solution of (6.5) to satisfy

$$u'_{N/2} - v = 0 \tag{6.8}$$

until the modulus of z becomes equal to ρ_a/ρ_s. Then we work with problem (6.3a) again. Equations (6.5) and (6.8) constitute an index 2 problem of type (1.1). If we consider in (6.5) the expression $\rho_s z$ as algebraic variable, then the problem is of the form (5.1) and our code HEM4 can be applied directly.

In order to solve this problem numerically, we applied the explicit Runge–Kutta code DOPRI5 (see [9]) on regions where the problem is an ordinary differential equation (i.e., where $u'_{N/2}(t) \neq v$) and we applied the half-explicit method HEM4 on regions where we have to deal with an index 2 problem. The switching points are computed by the use of dense output formulae. We have plotted in Fig. 3 the solution component $u_{N/2}$, its derivative $u'_{N/2}$, and the control variable z. The instants where we have to switch between the problems (6.3a) and (6.3b) are marked by (+) and (□).

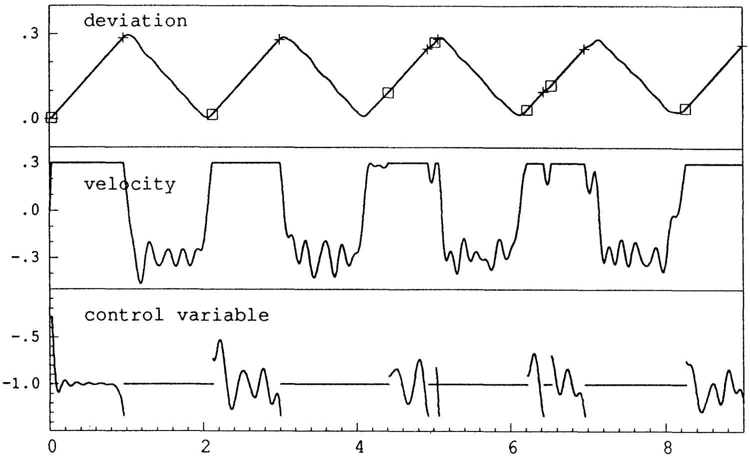


FIG. 3. *Solution of* (6.5).

The physical interpretation of the solution is as follows. In the beginning, the shearing force causes the string to move until the part of the string, which is in connection with the violin bow, gets a velocity equal to that of the violin bow (first "□"). Then the string will adhere for a while to the violin bow (index 2 situation) until the force on the string becomes too strong (first "+"). This is the moment where the string moves back and where we again have the situation of the departure (ordinary differential equation), and so on.

Acknowledgments. We are grateful to G. Wanner for proposing the violin problem, to A. Kastner-Maresch for drawing our attention to the connection between differential equations with discontinuities and differential-algebraic problems, and to W. Hundsdorfer for a careful reading of the manuscript.

REFERENCES

[1] K. E. BRENAN, S. L. CAMPBELL, AND L. R. PETZOLD, *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*, North-Holland, New York, 1989.

[2] K. E. BRENAN AND L. R. ENGQUIST, *Backward differentiation approximations of nonlinear differential/algebraic systems*, Math. Comp., 51 (1988), pp. 659–676.

[3] K. E. BRENAN AND L. R. PETZOLD, *The numerical solution of higher index differential/algebraic equations by implicit Runge–Kutta methods*, SIAM J. Numer. Anal., 26 (1989), pp. 976–996.

[4] H. CABANNES, *Mécanique des milieux continus—Propagation des discontinuités dans les cordes vibrantes soumises à un frottement solide*, Comptes Rendus Acad. Sci. Paris 289, série B (1979), pp. 127–130.

[5] C. FÜHRER AND B. LEIMKUHLER, *Numerical solution of differential-algebraic equations for constrained mechanical motion*, Numer. Math., 59 (1991), pp. 55–69.

[6] C. W. GEAR, G. K. GUPTA, AND B. LEIMKUHLER, *Automatic integration of Euler–Lagrange equations with constraints*, J. Comput. Appl. Math., 12 and 13 (1985), pp. 77–90.

[7] E. J. HAUG, *Computer Aided Kinematics and Dynamics of Mechanical Systems, Volume* I: *Basic Methods*, Allyn and Bacon, Boston, 1989.

[8] E. HAIRER, CH. LUBICH, AND M. ROCHE, *The numerical solution of differential-algebraic systems by Runge–Kutta methods*, Springer Lecture Notes in Math. 1409, Springer-Verlag, Berlin, New York, 1989.

[9] E. HAIRER, S. P. NØRSETT, AND G. WANNER, *Solving ordinary differential equations* I. *Nonstiff problems*, Computational Mathematics, Vol. 8, Springer-Verlag, Berlin, 1987.

[10] E. HAIRER AND G. WANNER, *Solving ordinary differential equations* II. *Stiff and differential-algebraic problems*, Computational Mathematics, Vol. 14, Springer-Verlag, Berlin, 1991.

[11] P. LÖTSTEDT AND L. PETZOLD, *Numerical solution of nonlinear differential equations with algebraic constraints* I: *Convergence results for backward differentiation formulas*, Math. Comp., 46 (1986), pp. 491–516.

[12] CH. LUBICH, *h^2-extrapolation methods for differential-algebraic systems of index* 2. Impact Comput. Sci. Engrg., 1 (1989), pp. 260–268.

[13] ———, *Extrapolation integrators for constrained multibody sytems*, Impact Comput. Sci. Engrg., 3 (1991), pp. 213–234.

[14] R. MÄRZ, *Higher-index differential-algebraic equations: Analysis and numerical treatment*, Banach Center Publ. 24.

[15] J. M. ORTEGA AND W. C. RHEINBOLDT, *Iterative Methods of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970.

[16] W. SCHIEHLEN, ED., *Multibody Systems Handbook*, Springer-Verlag, Berlin, 1990.

[17] S. WOLFRAM, *Mathematica. A System for Doing Mathematics by Computer*, Addison-Wesley, Reading, MA, 1988.