Half-explicit Runge–Kutta methods with explicit stages for differential-algebraic systems of index 2 *

M. ARNOLD

DLR Research Center Oberpfaffenhofen, Institute of Robotics and System Dynamics, P.O. Box 1116, D – 82230 Wessling, Germany
email: martin.arnold@dlr.de

Abstract.

Usually the straightforward generalization of explicit Runge–Kutta methods for ordinary differential equations to half-explicit methods for differential-algebraic systems of index 2 results in methods of order $q \leq 2$ ([10]). The construction of higher order methods is simplified substantially by a slight modification of the method combined with an improved strategy for the computation of the algebraic solution components. We give order conditions up to order $q = 5$ and study the convergence of these methods. Based on the fifth order method of Dormand and Prince the fifth order half-explicit Runge–Kutta method HEDOP5 is constructed that requires the solution of 6 systems of nonlinear equations per step of integration.

AMS subject classification: 65L05.

Key words: differential-algebraic systems, half-explicit methods, explicit Runge–Kutta methods.

1 Introduction.

The numerical solution of differential-algebraic systems requires the solution of nonlinear equations. That is why implicit discretization methods (implicit Runge–Kutta methods, BDF) are very popular for the integration of differential-algebraic systems. However, for non-stiff systems the numerical effort can be substantially decreased if only the algebraic part is discretized implicitly (e. g. [9], [13], [3], [17], [2], [15], see also the survey in Section VII.6 of the monograph [12]).

In the present paper we study the solution of the initial value problem

$$(1.1) \qquad \left. \begin{array}{rcl} y'(t) & = & f(y(t), z(t)) \\ 0 & = & g(y(t)) \end{array} \right\} \ , \quad t \in [t_0, t_e] \ , \quad y(t_0) = y_0 \ , \quad z(t_0) = z_0$$

by half-explicit Runge–Kutta methods. We suppose that (1.1) has a solution $y : [t_0, t_e] \to \mathbb{R}^{n_y}$, $z : [t_0, t_e] \to \mathbb{R}^{n_z}$ and that f and g are sufficiently differentiable and satisfy the *index-2 condition*

$$(1.2) \qquad [g_y f_z](\eta, \zeta) \quad \text{non-singular}$$

*Received October 1995. Revised June 1997.

in a neighbourhood of the trajectory $\{ (y(t), z(t)) : t \in [t_0, t_e] \}$. The initial values y_0, z_0 are assumed to be consistent:

$$g(y_0) = 0 \ , \ \ [g_y f](y_0, z_0) = 0 \ .$$

Because of (1.2) the differential-algebraic system (1.1) is of (differential and perturbation) index 2 ([10, p. 3f]). Systems of the form (1.1) arise in various applications, e. g. as index-2 formulation of model equations for constrained mechanical systems and in the integration of systems of ordinary differential equations (ODEs) with invariants (see Section 4 and [6]).

Half-explicit Runge–Kutta methods for (1.1) were introduced by Hairer et al. in [10]. They compute the differential solution components y similar to explicit Runge–Kutta methods for ODEs. The algebraic solution components z are defined such that all stage values Y_{ni} remain in the manifold $\{ \eta : g(\eta) = 0 \}$ that is given by the algebraic constraints in (1.1). In contrast to implicit Runge–Kutta methods and BDF that require the solution of systems of $\geq n_y + n_z$ nonlinear equations the systems of nonlinear equations that have to be solved in half-explicit Runge–Kutta methods are of dimension n_z, only. This approach is studied in detail in [3], [4], [5]. The code HEM5 ([3]) that is based on a fifth order half-explicit method is successfully used in the dynamical simulation of constrained mechanical systems.

A drawback of these half-explicit Runge–Kutta methods (we call them *Type A methods*) is a severe order reduction: for traditional higher order explicit Runge–Kutta methods the order usually drops down to $q = 2$. I. e., to handle the algebraic part of (1.1) Type A methods lose efficiency in the integration of the differential part. In the present paper we modify the approach of [10] in the first Runge–Kutta stage (*Type B methods*). This modification is closely related to the work of Murua ([15]) who introduced independently but approximately at the same time the class of partitioned half-explicit Runge–Kutta methods.

We prove for a wide class of methods up to classical order $p = 5$ that Type B methods do not suffer from order reduction in the differential components provided that the approximation of the algebraic components is sufficiently good. For methods with $p = s \leq 4$ and for the fifth order method of Dormand and Prince ([8], [11, p. 178f]) this approximation is obtained by at most one additional stage of the half-explicit method. We compared HEDOP5 — a fifth order Type B method based on the explicit Runge–Kutta method of Dormand and Prince — with the fifth order Type A method HEM5 [3] and with the fifth order partitioned half-explicit Runge–Kutta method PHEM56 [15]. Both HEDOP5 and PHEM56 are very efficient integrators for non-stiff index-2 systems, they are clearly superior to the Type A method HEM5. If the integrators are applied to the model equations of constrained mechanical systems then HEDOP5 is a little bit faster than PHEM56 since HEDOP5 needs less function evaluations per step of integration.

In Section 2 we define Type B methods and prove convergence. In Section 3 we analyse the local error, give order conditions up to order $q = 5$ and construct methods of order $q \leq 4$ and the fifth order method HEDOP5. Details of the implementation and results of numerical tests are discussed in Section 4.

2 Type B methods: Definition and convergence.

One step of an s-stage half-explicit Runge–Kutta method for (1.1) is given by ([10, p. 20f])

$$\left.\begin{array}{rcl} Y_{ni} & = & y_n + h\sum_{j=1}^{i-1} a_{ij} f(Y_{nj}, Z_{nj}) \\ 0 & = & g(Y_{ni}) \end{array}\right\} , \quad (i = i_0(1)\hat{s}+1) \ , \tag{2.1}$$

$$\left.\begin{array}{rcl} y_{n+1} & = & y_n + h\sum_{j=1}^{s} b_j f(Y_{nj}, Z_{nj}) \\ 0 & = & g(y_{n+1}) \end{array}\right\} \tag{2.2}$$

with

$$Y_{n1} := y_n \ , \quad i_0 := 2 \ , \quad \hat{s} := s \ , \qquad \text{(Type A method)}.$$

The a_{ij}, b_j denote the coefficients of the method with $a_{ij} := 0$ if $j \geq i$. Throughout the paper we set

$$a_{s+1,j} := b_j \ , \ (j = 1(1)s) \ , \ \ c_1 := 0 \ , \ \ c_i := \sum\nolimits_j a_{ij} \ , \ \ (i = 2(1)\hat{s}+1).$$

For simplicity we restrict ourselves to autonomous systems (1.1) but the results can be carried over to the non-autonomous case adding the auxiliary equation $t' = 1$ to (1.1). If $a_{i,i-1} \neq 0$, $(i = 2(1)s)$, $b_s \neq 0$, then the method is well-defined ([10, Theorem 4.10]): in the i-th stage $Z_{n,i-1}$ is computed as solution ζ of the system of nonlinear equations

$$0 = \Phi_i(\zeta) := g\Big(y_n + h\sum_{j=1}^{i-2} a_{ij} f(Y_{nj}, Z_{nj}) + h a_{i,i-1} f(Y_{n,i-1}, \zeta)\Big) \tag{2.3}$$

that is locally uniquely solvable if h is sufficiently small and y_n is close to $y(t_n)$ (because of (1.2) the Implicit function Theorem is applicable here). The method defines an approximation y_{n+1} to $y(t_n + h)$, Hairer et al. [10] suggest to use methods (2.1)/(2.2) with $c_s = 1$ such that $z_{n+1} := Z_{ns}$ gives an approximation to $z(t_n + h)$.

Remark 2.1.

a) *As in the ODE case* the final stage y_{n+1} should give a high order approximation to $y(t_n + h)$ but the first stage vector Y_{n2} gives only a poor approximation to $y(t_n + c_2 h)$. If an explicit Runge–Kutta method is applied with $y_n := y(t_n)$ to the ODE $y' = \varphi(y)$ then

$$Y_{n2} = y_n + c_2 h\varphi(y_n) = y(t_n) + c_2 h y'(t_n) = y(t_n + c_2 h) + \mathcal{O}(h^2)$$

and $\varphi(Y_{n2})$ is therefore a poor approximation to $y'(t_n + c_2 h)$. In higher order methods the influence of Y_{n2} on the final stage vector y_{n+1} is usually kept small setting $b_2 := 0$. In the fifth order method of Dormand and Prince ([11, p. 178f]) we even have for $i = 2$

$$b_i = 0 \ , \ \ \sum\nolimits_j b_j a_{ji} = 0 \ , \ \ \sum\nolimits_j b_j c_j a_{ji} = 0 \ , \ \ \sum\nolimits_{j,k} b_j a_{jk} a_{ki} = 0 \ . \tag{2.4}$$

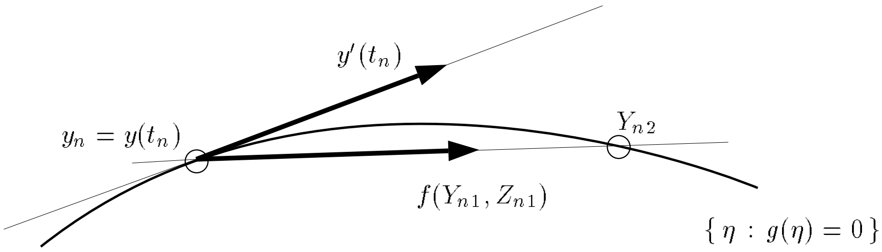


Figure 2.1: Difference between $f(Y_{n1}, Z_{n1})$ and $y'(t_n)$.

b) *In contrast to the ODE case* $f(Y_{ni}, Z_{ni})$ is not only for $i = 2$ a poor approximation to $y'(t_n + c_i h)$ but also for $i = 1$ (even if $y_n := y(t_n)$): The Taylor expansion of [10, p. 68f] proves

$$\begin{aligned} f(Y_{n1}, Z_{n1}) &= f(y(t_n), z(t_n)) + f_z(y(t_n), z(t_n))(Z_{n1} - z(t_n)) + \mathcal{O}(h^2) \\ &= y'(t_n) + \frac{c_2 h}{2}[f_z(-g_y f_z)^{-1} g_{yy}(f, f)](y(t_n), z(t_n)) + \mathcal{O}(h^2) \ , \end{aligned}$$

i. e., if the algebraic constraints g are nonlinear ($g_y \neq \text{const}$) then

$$f(Y_{n1}, Z_{n1}) - y'(t_n) = \mathcal{O}(h) \ .$$

Figure 2.1 illustrates that the large difference between $f(Y_{n1}, Z_{n1})$ and $y'(t_n)$ is caused by the condition $g(Y_{n2}) = 0$. If $Y_{n1} = y(t_n)$ then $y'(t_n)$ lies in the tangential plane of the manifold $\{\, \eta \,:\, g(\eta) = 0 \,\}$ at $y(t_n)$ since

$$g_y(y(t_n))y'(t_n) = \frac{d}{dt} g(y(t))\Big|_{t=t_n} = 0 \ .$$

On the other hand

$$f(Y_{n1}, Z_{n1}) = \frac{1}{c_2 h}(Y_{n2} - y(t_n)) \ ,$$

i. e. $f(Y_{n1}, Z_{n1})$ is parallel to the vector that joins the points $y(t_n)$, Y_{n2} of this manifold.

Remark 2.2.

a) Brasey and Hairer ([4], [3]) reduce the influence of $f(Y_{n1}, Z_{n1})$ on the final stage setting

$$b_1 = b_2 = 0 \ , \ \sum\nolimits_j b_j a_{j1} = \sum\nolimits_j b_j a_{j2} = 0$$

for a method of order $q = 4$ and (2.4) for $i = 1, 2, 3$ for a method of order $q = 5$. The resulting methods have 5 and 8 stages, respectively, i. e. more stages than in the ODE case (there are explicit Runge–Kutta methods for ODEs of order $p = 4$ with $s = 4$ and of order $p = 5$ with $s = 6$ stages). Another possibility to eliminate the large error term in $f(Y_{n1}, Z_{n1})$ are extrapolation methods [10, p. 49].

b) Figure 2.1 suggests an alternative definition of half-explicit Runge–Kutta methods: For higher order methods it is in general important that $f(Y_{n1}, Z_{n1})$ approximates $y'(t_n)$, it might be less important that Y_{n2} lies in the manifold $\{\, \eta \,:\, g(\eta) = 0 \,\}$. We therefore define

$$Z_{n1} := z_n \ , \ \ Y_{n2} := y_n + c_2 h f(y_n, z_n)$$

and need as z_n a sufficiently good approximation of $z(t_n)$. In the first step of integration z_n is available from the (consistent) initial values, approximations for the subsequent steps are obtained adding new stages at the end of each step of integration.

DEFINITION 2.1. *An $\hat{s}$-stage half-explicit Runge–Kutta method of Type B is given by*

$$Y_{n1} := y_n \ \ , \ \ Z_{n1} := z_n \ \ , \ \ Y_{n2} := y_n + c_2 h f(y_n, z_n)$$

and (2.1)/(2.2) with $i_0 = 3$, $\hat{s} \geq s$,

$$z_{n+1} := \sum_{i=1}^{\hat{s}} d_i Z_{ni} \ \ .$$

Here $d_1, \ldots, d_{\hat{s}}$ and a_{ij}, ($i = 2(1)\hat{s}+1$, $j = 1(1)i-1$), are parameters of the method with

$$a_{s+1,j} := b_j \ , \ (j = 1(1)s) \ , \ c_{s+1} := 1 \ , \ \ a_{i,i-1} \neq 0 \ , \ (i = 2(1)\hat{s}+1) \ .$$

We form a lower triangular matrix

$$\tilde{A} = \begin{pmatrix} a_{21} & & & \\ a_{31} & a_{32} & & \\ \vdots & \vdots & \ddots & \\ a_{\hat{s}+1,1} & a_{\hat{s}+1,2} & \cdots & a_{\hat{s}+1,\hat{s}} \end{pmatrix} \in \mathbb{R}^{\hat{s}\times\hat{s}}$$

that contains on and below the main diagonal the non-vanishing parameters a_{ij}. With the assumptions of Definition 2.1 matrix $\tilde{A}$ is non-singular, the elements of the inverse $W = \tilde{A}^{-1}$ are denoted by w_{ij}.

In contrast to Type A methods the stage vectors Y_{ni} and y_{n+1} depend now on z_n, i. e. during integration errors are propagated not only in the differential components y but also in the algebraic components z. More precisely: consider vectors $\hat{y}_n$, $\hat{z}_n$, $\hat{Y}_{ni}$, $\hat{Z}_{ni}$, $\hat{y}_{n+1}$, $\hat{z}_{n+1}$, that satisfy

$$\hat{Y}_{n1} := \hat{y}_n \ , \ \ \hat{Z}_{n1} := \hat{z}_n \ , \ \ \hat{Y}_{n2} := \hat{y}_n + c_2 h f(\hat{y}_n, \hat{z}_n) + h\delta_2 \ ,$$

$$\left.\begin{array}{rcl} \hat{Y}_{ni} & = & \hat{y}_n + h \displaystyle\sum_{j=1}^{i-1} a_{ij} f(\hat{Y}_{nj}, \hat{Z}_{nj}) + h\delta_i \\ \theta_i & = & g(\hat{Y}_{ni}) \end{array}\right\} \ , \quad (i = 3(1)\hat{s}+1) \ , \tag{2.5}$$

$$\hat{y}_{n+1} := \hat{Y}_{n,s+1} \ , \ \ \hat{z}_{n+1} := \sum\nolimits_i d_i \hat{Z}_{ni} \ .$$

It holds (see also [10, Theorem 4.2])

THEOREM 2.1. *Let (y_n, z_n), $(\hat{y}_n, \hat{z}_n)$ be given with*

$$\|y_n - y(t_n)\| + \|z_n - z(t_n)\| = \mathcal{O}(h)\ ,\ \|\hat{y}_n - y(t_n)\| + \|\hat{z}_n - z(t_n)\| = \mathcal{O}(h)$$

and

$$\|g(y_n)\| + \|g(\hat{y}_n)\| = \mathcal{O}(h^2),\ \|\delta\| := \max_i \|\delta_i\| = \mathcal{O}(h),\ \|\theta\| := \max_i \|\theta_i\| = \mathcal{O}(h^2)\ .$$

Then there are vectors (Y_{ni}, Z_{ni}), $(\hat{Y}_{ni}, \hat{Z}_{ni})$ with (2.1), (2.5) and

$$\|Y_{ni} - y(t_n)\| + \|Z_{ni} - z(t_n)\| = \mathcal{O}(h)\ ,\ \|\hat{Y}_{ni} - y(t_n)\| + \|\hat{Z}_{ni} - z(t_n)\| = \mathcal{O}(h)$$

if the stepsize $h \geq 0$ is sufficiently small. These vectors satisfy the estimates

$$\|\hat{Y}_{ni} - Y_{ni}\| \leq (1 + Ch)\|P(t_n)(\hat{Y}_{ni} - Y_{ni})\| + C\|\theta_i\|,\ (i = 3(1)\hat{s} + 1), \tag{2.6}$$

$$\|P(t_n)(\hat{Y}_{ni} - Y_{ni})\| \leq C(\|\hat{y}_n - y_n\| + h^2\|\hat{z}_n - z_n\| + h\|\delta\| + \|\theta\|),\quad (i = 2(1)\hat{s} + 1), \tag{2.7}$$

$$\begin{aligned} \|\hat{Z}_{ni} - Z_{ni} - c_2 w_{i1}(\hat{z}_n - z_n)\| &\leq C(\tfrac{1}{h}\|g_y(y(t_n))(\hat{y}_n - y_n)\| + \\ &+ \|\hat{y}_n - y_n\| + h\|\hat{z}_n - z_n\| + \|\delta\| + \tfrac{1}{h}\|\theta\|)\ ,\ (i = 1(1)\hat{s}) \end{aligned} \tag{2.8}$$

with a constant C that is independent of h, δ and θ. Here P denotes the projector $P(t) := I - [f_z(g_y f_z)^{-1} g_y](y(t), z(t))$ that represents a projection into the tangential plane of the manifold $\{\,\eta : g(\eta) = 0\,\}$, ([10, p. 35 and p. 68]).

PROOF. Theorem 2.1 is the counterpart to Theorems 4.1 and 4.2 in [10]. We prove that there are such vectors (Y_{ni}, Z_{ni}), $(\hat{Y}_{ni}, \hat{Z}_{ni})$ in a $\mathcal{O}(h)$–neighbourhood of the analytical solution (part a) of the proof) and that these vectors satisfy estimates (2.6)–(2.8) (part b)).

a) In (2.1) the vectors Z_{ni}, $(i \geq i_0 - 1)$ are defined implicitly as solutions ζ of (2.3): $\Phi_i(Z_{n,i-1}) = 0$. With the assumptions of the theorem we prove by induction that there is a constant C (independent of h) such that

$$\|\Big(\frac{\partial}{\partial\zeta}\Phi_i(z(t_n))\Big)^{-1}\Phi_i(z(t_n))\| \leq Ch\ ,\ \ \|\Big(\frac{\partial}{\partial\zeta}\Phi_i(z(t_n))\Big)^{-1}\| \leq C \cdot \frac{1}{h}\ ,$$

$$\|\frac{\partial^2}{\partial\zeta^2}\Phi_i(z(t_n))\| \leq Ch$$

is satisfied for all sufficiently small $h > 0$. Therefore the (locally) unique solvability of (2.3) is guaranteed by the Theorem of Newton–Kantorovich and we have $\|Z_{n,i-1} - z(t_n)\| = \mathcal{O}(h)$ ([16]). Using $Z_{n,i-1}$ the vector Y_{ni} can be computed explicitly from (2.1). Going on stage by stage we obtain all stage vectors (Y_{ni}, Z_{ni}) (this proof was originally given by Brasey and Hairer [4]). In a similar way the existence of vectors $(\hat{Y}_{ni}, \hat{Z}_{ni})$ can be shown.

b) Estimate (2.6) is a consequence of

$$
\begin{aligned}
& g_y(y(t_n))(\hat{Y}_{ni} - Y_{ni}) = \\
& \quad = \int_0^1 \Big(g_y(y(t_n))(\hat{Y}_{ni} - Y_{ni}) - \frac{d}{d\vartheta} g(Y_{ni}^{\vartheta}) \Big) d\vartheta + g(\hat{Y}_{ni}) - g(Y_{ni}) \\
& \quad = \int_0^1 \Big(g_y(y(t_n)) - g_y(Y_{ni}^{\vartheta}) \Big) d\vartheta \cdot (\hat{Y}_{ni} - Y_{ni}) + \theta_i
\end{aligned}
\tag{2.9}
$$

with $Y_{ni}^{\vartheta} := Y_{ni} + \vartheta(\hat{Y}_{ni} - Y_{ni})$ since

$$
\begin{aligned}
\|\hat{Y}_{ni} - Y_{ni}\| &\le \|P(t_n)(\hat{Y}_{ni} - Y_{ni})\| + \|(I - P(t_n))(\hat{Y}_{ni} - Y_{ni})\| \\
&\le \|P(t_n)(\hat{Y}_{ni} - Y_{ni})\| + C(h\|\hat{Y}_{ni} - Y_{ni}\| + \|\theta_i\|) \ .
\end{aligned}
$$

To prove (2.7) and (2.8) we transform (2.1) and (2.5) to

$$
\begin{aligned}
\hat{Y}_{ni} - Y_{ni} = \hat{y}_n - y_n + h \sum\nolimits_j a_{ij} \Big(& (f_y^n + \mathcal{O}(h))(\hat{Y}_{nj} - Y_{nj}) + \\
& + (f_z^n + \mathcal{O}(h))(\hat{Z}_{nj} - Z_{nj}) \Big) + h\delta_i
\end{aligned}
\tag{2.10}
$$

where the upper index "n" indicates that the Jacobians f_y, f_z, g_y are evaluated at $(y(t_n), z(t_n))$. With the notations

$$
Y := (Y_{n2}^T, \ldots, Y_{n,\hat{s}+1}^T)^T \ , \quad Z := (Z_{n1}^T, \ldots, Z_{n,\hat{s}}^T)^T \ , \quad \ldots
$$

we thus obtain

$$
\begin{aligned}
h((\tilde{A} \otimes [g_y^n f_z^n]) + \mathcal{O}(h))(\hat{Z} - Z) = & -(\mathbf{1} \otimes g_y^n)(\hat{y}_n - y_n) + \\
& + \mathcal{O}(h)\|\hat{Y} - Y\| + (I \otimes g_y^n)(\hat{Y} - Y) + \mathcal{O}(h\|\delta\|)
\end{aligned}
\tag{2.11}
$$

with $\mathbf{1} := (1, \ldots, 1)^T$ and the Kronecker product $\otimes$. Because of

$$
g(Y_{ni}) = 0 \ , \quad g(\hat{Y}_{ni}) = \theta_i \ , \ (i \ge 3)
$$

the term $(I \otimes g_y^n)(\hat{Y} - Y)$ can be expressed as (see (2.9))

$$
(I \otimes g_y^n)(\hat{Y} - Y) = \mathcal{O}(\|\theta\|) + \mathcal{O}(h)\|\hat{Y} - Y\| + \boldsymbol{e_1} \otimes g_y^n \cdot (\hat{Y}_{n2} - Y_{n2})
$$

with $\boldsymbol{e_1} := (1, 0, \ldots, 0)^T$ and

$$
g_y^n \cdot (\hat{Y}_{n2} - Y_{n2}) = g_y^n \cdot (\hat{y}_n - y_n) + c_2 h [g_y^n f_z^n](\hat{z}_n - z_n) + \mathcal{O}(h)(\|\hat{y}_n - y_n\| + h\|\hat{z}_n - z_n\|) \ .
$$

So we get in (2.11)

$$
\begin{aligned}
((\tilde{A} \otimes [g_y^n f_z^n]) + \mathcal{O}(h)) \Big(\hat{Z} - Z - c_2 W \boldsymbol{e_1} \cdot (\hat{z}_n - z_n) \Big) = & \\
= \mathcal{O}(\tfrac{1}{h})\|g_y^n \cdot (\hat{y}_n - y_n)\| + \mathcal{O}(1)(\|\hat{y}_n - y_n\| + \|\hat{Y} - Y\|) + & \\
+ \mathcal{O}(h)\|\hat{z}_n - z_n\|) + \mathcal{O}(1)\|\delta\| + \mathcal{O}(\tfrac{1}{h})\|\theta\| \ . &
\end{aligned}
\tag{2.12}
$$

Because of (1.2) the matrix $\tilde{A} \otimes [g_y^n f_z^n]$ is non-singular. Hence, the estimates (2.7), (2.8) follow from (2.12) and (2.10), (note, that

$$P(t) f_z(y(t), z(t)) \equiv 0 \quad \text{and} \quad P(t_n)(f_z^n + \mathcal{O}(h)) = \mathcal{O}(h) \;).$$

□

Theorem 2.1 is the essential part of the convergence analysis for Type B methods that bounds the global discretization error in terms of the local error:

DEFINITION 2.2. *Let $\hat{y}_{n+1}$, $\hat{z}_{n+1}$ be defined by (2.5) with $\delta_i = 0$, $\theta_i = 0$, $(i = 2(1)\hat{s}+1)$ and the analytical solution of (1.1) at time t_n as $\hat{y}_n$, $\hat{z}_n$: $\hat{y}_n := y(t_n)$, $\hat{z}_n := z(t_n)$. Then*

$$\delta y_h(t_n) := \hat{y}_{n+1} - y(t_n + h) \;\;,\;\; \delta z_h(t_n) := \hat{z}_{n+1} - z(t_n + h)$$

are called the local discretization errors *in the differential and algebraic part, respectively.*

THEOREM 2.2. *Suppose that in (2.1) $a_{i,i-1} \neq 0$, $(i = 2(1)\hat{s}+1)$, that the local discretization errors are of size*

$$\delta y_h(t_n) = \mathcal{O}(h^q) \;\;,\;\; P(t_n)\delta y_h(t_n) = \mathcal{O}(h^{q+1}) \;\;,\;\; \delta z_h(t_n) = \mathcal{O}(h^{q-1})$$

with some $q \geq 2$ and that the contractivity condition

$$|c_2 d^T W \boldsymbol{e_1}| < 1 \tag{2.13}$$

is satisfied. Then the method is convergent with orders q and $q-1$, respectively:

$$\|y_m - y(t_m)\| + h\|z_m - z(t_m)\| = \mathcal{O}(h^q) \quad \textit{for } t_m = t_0 + mh \in [t_0, t_e] \;.$$

Here d denotes the vector of coefficients $d := (d_1, \dots, d_{\hat{s}})^T$, i. e.

$$c_2 d^T W \boldsymbol{e_1} = c_2 \sum\nolimits_j d_j w_{j1} \;.$$

PROOF. a) As in (2.6) we get

$$\|y_m - y(t_m)\| \leq (1 + \mathcal{O}(h)) \|P(t_m)(y_m - y(t_m))\|$$

since $g(y_m) = g(y(t_m)) = 0$. I. e. if $\|P(t_m)(y_m - y(t_m))\| = \mathcal{O}(h^q)$ then convergence with order q in the differential components is guaranteed.

b) Let $\hat{y}_{n+1}$, $\hat{z}_{n+1}$ be defined as in Definition 2.2 then

$$\begin{aligned} P(t_{n+1})(y_{n+1} - y(t_{n+1})) &= P(t_{n+1})(y_{n+1} - \hat{y}_{n+1}) + P(t_{n+1})\delta y_h(t_n) \\ h(z_{n+1} - z(t_{n+1})) &= h(z_{n+1} - \hat{z}_{n+1}) + \mathcal{O}(h^q) \end{aligned}$$

and

$$P(t_{n+1})\delta y_h(t_n) = P(t_n)\delta y_h(t_n) + \mathcal{O}(h)\delta y_h(t_n) = \mathcal{O}(h^{q+1}) \;.$$

In contrast to Type A methods errors are not only propagated in the function values $y_n \approx y(t_n)$ of the differential components but also in the derivatives $f(y_n, z_n) \approx y'(t_n)$. This is similar to the error propagation in Rosenbrock methods that are applied to differential-algebraic equations $B(y)y' = \varphi(y)$ of index 1 ([14]). The statement of the theorem can be proved following the proof of [14, Theorem 4.1] using the inequality

$$\begin{pmatrix} \|P(t_{n+1})\Delta y_{n+1}\| \\ h\|\Delta z_{n+1}\| \end{pmatrix} \leq \begin{pmatrix} 1+\mathcal{O}(h) & \mathcal{O}(h) \\ \mathcal{O}(1) & \rho+\mathcal{O}(h) \end{pmatrix} \begin{pmatrix} \|P(t_n)\Delta y_n\| \\ h\|\Delta z_n\| \end{pmatrix} \tag{2.14}$$

with

$$\Delta y_n := \hat{y}_n - y_n \ , \quad \Delta z_n := \hat{z}_n - z_n \quad \text{and} \quad \rho := |c_2 d^T W \boldsymbol{e_1}|$$

that has to be read componentwise. Estimate (2.14) is obtained straightforwardly from Theorem 2.1 since

$$\|y_n - y(t_n)\| \leq (1+\mathcal{O}(h))\|P(t_n)(y_n - y(t_n))\| \ ,$$

(see (2.6)) and $P(t_{n+1}) = P(t_n) + \mathcal{O}(h)$. □

Remark 2.3.

a) As in [14] the restriction to fixed stepsizes is not essential.

b) In contrast to Type A methods we need consistent initial values for differential *and* algebraic solution components, the statement of Theorem 2.2 remains valid if $z_0 = z(t_0)$ is substituted by $z_0 = z(t_0) + \mathcal{O}(h^{q-1})$.

c) The contractivity condition $|c_2 d^T W \boldsymbol{e_1}| < 1$ is important. If $|c_2 d^T W \boldsymbol{e_1}| > 1$ then the numerical solution in general diverges, if $|c_2 d^T W \boldsymbol{e_1}| = 1$ then order reduction may occur.

d) The solution of (1.1) satisfies

$$0 = \frac{d}{dt} g(y(t)) = g_y(y(t))y'(t) = [g_y f](y(t), z(t)) \ .$$

Hence, a numerical solution ζ for the algebraic components $z(t_n)$ can be obtained as solution of $[g_y f](y_n, \zeta) = 0$ if a high order approximation y_n to the differential components $y(t_n)$ has been computed before ([4]). Thus [3], [4] do not focus on a high order of convergence for the algebraic components z, the methods HEM4 and HEM5 that converge in the differential components y with order $q = 4$ and $q = 5$, respectively, have order $q - 2$ for z. As a by-product of our approach Type B methods give a better approximation for the algebraic components z: a method of order q (in y) has a global error of order $\mathcal{O}(h^{q-1})$ in z.

3 Type B methods of order $q \leq 5$.

Estimates for the local error are obtained by Taylor expansion using a generalization of Butcher's tree model to differential-algebraic systems of index 2 that was developed by Hairer et al. in [10, Section 5]. For Type A methods this analysis is carried out in detail in [10, p. 68ff] (see also [4], [3]). The way in that order conditions are obtained is similar to ODE theory ([7]) but it needs a lot of additional notations. Deriving the order conditions for Type B methods we therefore follow the analysis in [10] as far as possible. The essential difference to Type A methods is in the first stage where we do not have $g(Y_{n2}) = 0$ but

$$Y_{n2} = y(t_n) + c_2 h y'(t_n) \ , \quad Z_{n1} = z(t_n)$$

provided that $(y_n, z_n) = (y(t_n), z(t_n))$. The order conditions can be obtained as in [10] if Formula (5.34b) that reads in our notation

$$\dot{Z}_{ni}(0) = \frac{1}{2} \sum\nolimits_{j,k,l} w_{ij} a_{j+1,k} a_{j+1,l} \cdot (-g_y f_z)^{-1} g_{yy}(f,f) + \sum\nolimits_j a_{ij} \cdot (-g_y f_z)^{-1} g_y f_y f$$

is substituted by

$$\dot{Z}_{ni}(0) = \frac{1}{2} \sum\nolimits_{j,k,l} w_{ij} \hat{a}_{j+1,k} \hat{a}_{j+1,l} \cdot (-g_y f_z)^{-1} g_{yy}(f,f) + \sum\nolimits_j a_{ij} \cdot (-g_y f_z)^{-1} g_y f_y f$$

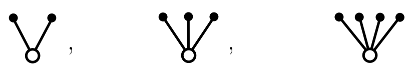
with

$$\hat{a}_{j+1,k} := \begin{cases} 0 \quad , & \text{if } j = 1 \ , \\ a_{j+1,k} , & \text{if } j > 1 \ . \end{cases}$$

In the higher order terms $\frac{d^p}{dh^p} Z_{ni}\Big|_{h=0}$ similar modifications are necessary for the coefficients of all elementary differentials of the form

$$(-g_y f_z)^{-1} g_{yy}(f,f) \ , \ (-g_y f_z)^{-1} g_{yyy}(f,f,f) \ , \ (-g_y f_z)^{-1} g_{yyyy}(f,f,f,f) \ , \ \ldots$$

that correspond to the so-called *bushy* trees with fat root:

(3.1)  , $\ldots$.

(These trees contain exactly one fat vertex — the root vertex; all other vertices are meagre and lie directly above the fat root vertex.)

Table 3.1 summarizes the order conditions for Type B methods up to order $q_y = 3$ for the differential components and up to order $q_z = 1$ for the algebraic components. Here and in the following we use the notation

$$\hat{c}_i := \sum_{j=1}^{i-1} \hat{a}_{ij} \ , \quad (i = 1(1)\hat{s}+1) \ ,$$

i. e. $\hat{c}_1 = \hat{c}_2 = 0$ and $\hat{c}_i = c_i$, $(i \geq 3)$.

The order conditions in Table 3.1 correspond to the given trees and can be obtained by the following algorithm (this algorithm is very similar to the one for Type A methods [4]):

ALGORITHM 3.1. Attach to each vertex of a given rooted tree T one summation index i, j, ... If the root of T is a fat vertex then attach an additional summation index k to this tree. The sub-graph of T that contains a vertex "i" and all vertices lying above "i" is denoted by **subtree**("i"), i. e. "i" is the root vertex of **subtree**("i").

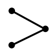
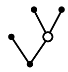
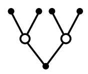
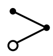
no.	tree	order	order condition
1_y		1	$\sum b_i = 1$
2_y		2	$\sum b_i c_i = \frac{1}{2}$
3_y		3	$\sum b_i c_i^2 = \frac{1}{3}$
4_y		3	$\sum b_i a_{ij} c_j = \frac{1}{6}$
5_y		3	$\sum b_i c_i w_{ij} \hat{c}_{j+1}^2 = \frac{2}{3}$
6_y		3	$\sum b_i w_{ij} \hat{c}_{j+1}^2 w_{ik} \hat{c}_{k+1}^2 = \frac{4}{3}$
1_z		1	$\sum d_i w_{ij} \hat{c}_{j+1}^2 = 2$
2_z		1	$\sum d_i w_{ij} a_{j+1,k} c_k = 1$

Table 3.1: Order conditions for Type B methods ($q_y \leq 3$, $q_z \leq 1$).

With these notations 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 vertex and this root is meagre;
- $d_k w_{ki}$, if "i" is the index of the root vertex and this root is fat (here "k" denotes the additional index that was attached to the tree);
- a_{ij}, if the meagre vertex "j" lies directly above the meagre vertex "i";
- w_{ij}, if the fat vertex "j" lies directly above the meagre vertex "i";
- $\hat{a}_{i+1,j}$ if the meagre vertex "j" lies directly above the fat vertex "i" and `subtree`("i") is a bushy tree;
- $a_{i+1,j}$ if the meagre vertex "j" lies directly above the fat vertex "i" and `subtree`("i") is not a bushy tree.

As for Type A methods ([4, p. 541f]) 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.

Here r denotes the order of `subtree`("i"), i. e. the difference of the number of meagre and the number of fat vertices in `subtree`("i").

Trees that have only meagre vertices correspond to classical order conditions for the underlying explicit Runge–Kutta method (see Table 3.1 and [7]).

Taking into account $W = \tilde{A}^{-1}$ some order conditions may be substantially simplified, e. g. condition 2_z of Table 3.1 is transformed to

$$1 = \sum\nolimits_{i,j,k} d_i w_{ij} a_{j+1,k} c_k = \sum\nolimits_{i,k} d_i \delta_{ik} c_k = \sum\nolimits_i d_i c_i$$

with the Kronecker delta δ_{ij}.

Formally the order conditions for Type B methods are a little bit more complicated than the ones for Type A methods. But this slight difference in the order conditions simplifies the construction of higher order half-explicit Runge–Kutta methods essentially:

An underlying explicit Runge–Kutta method that satisfies the simplifying condition

$$\mathrm{C}(2)\ :\quad \sum_{j=1}^{i-1} a_{ij} c_j = \frac{1}{2} c_i^2 \quad ,\quad (i = 3(1)\hat{s}+1)$$

implies for Type B methods (note, that $W = \tilde{A}^{-1}$)

$$\mathrm{C}(2)\mathrm{R}\ :\quad \sum_{j=1}^{i} w_{ij} \hat{c}_{j+1}^2 = 2c_i \quad ,\quad (i = 1(1)\hat{s})\ .$$

These *reciproque conditions* are substantially simpler than that for Type A methods ([3]) and guarantee that the order condition corresponding to the tree T_1 of Figure 3.1 is satisfied whenever the order condition corresponding to tree T_2 is satisfied. Here T denotes an arbitrary trunk. In Table 3.1 identical order conditions are obtained from trees 3_y, 5_y, and 6_y if C(2) is satisfied.

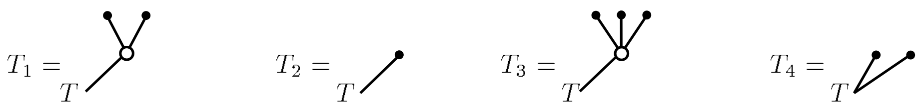


Figure 3.1: Trees illustrating the simplification of order conditions by conditions C(2) and C(3).

Furthermore, the order condition corresponding to the tree T_3 in Fig. 3.1 is satisfied if the order condition corresponding to the tree T_4 is satisfied and the coefficients of the underlying explicit Runge–Kutta method fulfill the simplifying condition

$$\mathrm{C}(3)\ :\quad \sum_{j=1}^{i-1} a_{ij} c_j^l = \frac{1}{l+1} c_i^{l+1} \quad ,\quad (i = 3(1)\hat{s}+1,\ l = 1,2)\ ,$$

since C(3) implies the reciproque conditions

$$\mathrm{C}(3)\mathrm{R}\ :\quad \sum_{j=1}^{i} w_{ij} \hat{c}_{j+1}^{l+1} = (l+1) c_i^l \quad ,\quad (i = 1(1)\hat{s},\ l = 1,2)\ .$$

(Note, that C(3) implies C(2) and C(3)R implies C(2)R). Constructing higher order half-explicit methods we restrict ourselves in the present paper to methods up to order $q_y = 5$ and make use of the simplifying conditions C(2), C(3) and

$$\mathrm{D}(1)\ :\quad \sum_{j=i+1}^{s} b_j a_{ji} = b_i (1 - c_i) \quad ,\quad (i = 1(1)s)\ .$$

THEOREM 3.1. *An explicit Runge–Kutta method of order p with $2 \le p \le 5$ yields to a Type B method with local discretization error $\delta y_h(t) = \mathcal{O}(h^{p+1})$ if the simplifying conditions*

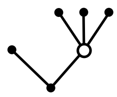
- C(2) *if* $p = 3$,
- C(2) *and* D(1) *if* $p = 4$,
- C(3) *and* D(1) *if* $p = 5$

are satisfied.

PROOF. $p \le 3$: If $p = 2$ or $p = 3$ and C(2) is satisfied then all order conditions of order $q_y \le p$ in Table 3.1 are satisfied (see also condition C(2)R).
$p = 4$: If $p = 4$ and C(2) is satisfied then $b_2 = 0$ since

$$b_2 c_2^2 = \sum_i b_i (c_i^2 - 2 \sum_j a_{ij} c_j) = \sum_i b_i c_i^2 - 2 \sum_{i,j} b_i a_{ij} c_j = \frac{1}{3} - \frac{1}{3} = 0$$

and $c_2 = a_{21} \neq 0$. We now have to check all order conditions that result from Algorithm 3.1 applied to trees T of order ≤ 4 that have a meagre root vertex. Compared with the ODE case the only additional order condition for a Type B method satisfying C(2) is

(3.2) 
$$\sum_{i,j} b_i c_i w_{ij} \hat{c}_{j+1}^3 = \frac{3}{4} \ .$$

Because of $W = \tilde{A}^{-1}$ we have

$$\sum_{i,j,k} b_i a_{ij} w_{jk} \hat{c}_{k+1}^3 = \sum_{i,k} b_i \delta_{i,k+1} \hat{c}_{k+1}^3 = \sum_{i>2} b_i c_i^3 = \frac{1}{4}$$

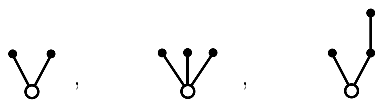
and

$$\sum_{i,j} b_i w_{ij} \hat{c}_{j+1}^3 = \sum_j \delta_{sj} \hat{c}_{j+1}^3 = \hat{c}_{s+1}^3 = c_{s+1}^3 = 1$$

such that condition D(1) implies (3.2).
$p = 5$: If condition C(3) is satisfied then the number of order conditions is reduced substantially because of the reciproque condition C(3)R and because of the identity

$$\sum_{j,k} w_{ij} c_{j+1} a_{j+1,k} c_k = \sum_{j,k} w_{ij} \hat{c}_{j+1} a_{j+1,k} c_k = \frac{1}{2} \sum_j w_{ij} \hat{c}_{j+1}^3 = \frac{1}{2} \cdot 3 c_i^2 \ .$$

Trees T that contain one of the trees

(3.3) 

do not give additional order conditions. Therefore the statement for $p = 5$ can be proved as in the case $p = 4$. $\square$

The order of the local error in the differential components y is completely defined by the parameters of the underlying s stage explicit Runge–Kutta method since y_{n+1} is identical with the stage vector $Y_{n,s+1}$. The Type B method has additional parameters d_i, $(i = 1(1)\hat{s})$ and — if $\hat{s} > s$, i. e. if new stage are added — additional parameters a_{ij}, $(i = s+2(1)\hat{s}+1,\ j = 1(1)i-1)$. They are defined such that the contractivity condition (2.13) and the order conditions for the algebraic components z are satisfied. The order conditions are obtained if Algorithm 3.1 is applied to the trees U with a fat root vertex.

As for the differential components the number of independent order conditions is reduced drastically using simplifying conditions:

THEOREM 3.2. *The local error in the algebraic components is of size*

$$\delta z_h(t) = \mathcal{O}(h^{r+1}) \quad \textit{with} \quad 0 \le r \le 3$$

if the coefficients of the method satisfy the simplifying condition C(r) *(if $r > 1$) and*

$$\sum_{i=1}^{\hat{s}} d_i c_i^l = 1\ ,\quad (l = 0(1)r)\ , \tag{3.4}$$

$$\frac{1}{r+1} \sum_{i,j=1}^{\hat{s}} d_i w_{ij} \hat{c}_{j+1}^{r+1} = 1\ , \tag{3.5}$$

$$d_2 = 0\ ,\quad \textit{(if } r \ge 2\textit{) and}$$

$$\sum_{i=1}^{\hat{s}} d_i a_{i2} = \sum_{i,j=1}^{\hat{s}} d_i w_{ij} c_{j+1} a_{j+1,2} = 0\ ,\quad \textit{(if } r = 3\textit{).} \tag{3.6}$$

PROOF. The order conditions (3.4) correspond to the "one-leg" trees with fat root and a "bushy" tree of order $\le r+1$ as branch. See e. g. the tree U_1 of Figure 3.2 that corresponds to

$$1 = \sum\nolimits_{i,j,k} d_i w_{ij} a_{j+1,k} c_k^2 = \sum\nolimits_{i,k} d_i \delta_{ik} c_k^2 = \sum\nolimits_i d_i c_i^2\ .$$

The "bushy" trees with a fat root (see (3.1)) give order conditions

$$\sum_{i,j=1}^{\hat{s}} d_i w_{ij} \hat{c}_{j+1}^{l+1} = l+1\ ,\quad (l = 1(1)r)$$

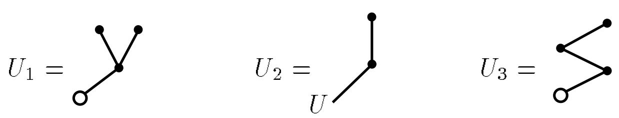


Figure 3.2: Trees illustrating the proof of Theorem 3.2.

(see e. g. condition 1_z in Table 3.1). For $l = 1(1)r-1$ these conditions are equivalent to (3.4) if C(r) (and hence also C(r)R) is satisfied. For $l = r$ we obtain condition (3.5).

The conditions (3.6) guarantee that up to order r the order conditions for trees of the form U_2 are satisfied whenever (3.4) and the simplifying assumption C(r) is satisfied: e. g. the tree U_3 of Figure 3.2 gives ($r \geq 2$)

$$\sum\nolimits_{i,j} d_i a_{ij} c_j = \frac{1}{2}\sum\nolimits_i d_i c_i^2 + \sum\nolimits_{i\geq 3} d_i (\sum\nolimits_j a_{ij} c_j - \frac{1}{2} c_i^2) - \frac{1}{2} d_2 c_2^2 = \frac{1}{2}\sum\nolimits_i d_i c_i^2 = \frac{1}{2}$$

All the remaining trees contain at a branch leaving a meagre vertex one of the trees of (3.3). As for the differential components these trees do not give new order conditions if the simplifying condition C(r) is satisfied. □

In Example 3.1 we construct Type B methods up to order $q = 5$ (in the differential components). Table 3.2 summarizes the essential numerical effort of these methods per one step of integration. Note, that (in contrast to Type A methods) the first stage of a Type B method is explicit and does not require the solution of a system (2.3) of nonlinear equations. For a Type B method of order q the order of convergence for the algebraic components equals $q-1$. Order and contractivity conditions result in systems of equations that have to be solved w. r. t. the parameters. In Example 3.1 we transform at first these conditions into systems of *linear* equations that are finally solved using MATHEMATICA [20].

	$q=2$	$q=3$	$q=4$	$q=5$
# of stages (ODE)	2	3	4	6
# of stages (Type A)	2	3	5	8
# of stages (Type B)	2	3	5	7
# of systems (2.3) (Type B)	1	2	4	6

Table 3.2: Numerical effort of explicit Runge–Kutta methods and half-explicit methods of Type A and B

EXAMPLE 3.1.

a) $q = 2$. An explicit Runge–Kutta method with $p = s = 2$ yields a Type B method of order $q = 2$ if $\hat{s} := s = 2$, $d_i := b_i$, $(i = 1, 2)$ since

$$\sum\nolimits_i d_i = \sum\nolimits_i b_i = 1 \quad \text{and} \quad c_2 d^T W \boldsymbol{e_1} = c_2 \sum\nolimits_i d_i w_{i1} = c_2 \sum\nolimits_i b_i w_{i1} = c_2 \delta_{21} = 0$$

b) $q = 3$. The (classical) order conditions for an explicit Runge–Kutta method with $p = s = 3$ are

$$\sum\nolimits_i b_i c_i^l = \frac{1}{l+1}\,, \ (l = 0, 1, 2)\,, \quad b_3 a_{32} c_2 = \frac{1}{6}\,.$$

Straightforward transformations show that a method satisfying the simplifying condition C(2) has parameters

$$b_1 = \frac{1}{4}\ ,\ \ b_2 = 0\ ,\ \ b_3 = \frac{3}{4}\ ,\ \ c_3 = \frac{2}{3}\ ,\ \ a_{32} = \frac{2}{9c_2}\ ,$$

the parameter $c_2 \neq 0$ remains free ([11, Section II.1]). If we apply Theorem 3.2 with $r = q - 2 = 1$ then conditions (3.5) and (3.4) with $l = 1$ coincide since condition C(2) and therefore also condition C(2)R is satisfied. The unknown parameters d_1, d_2, d_3 are determined by the order conditions and by the equation $c_2 d^T W \boldsymbol{e_1} = 0$ that guarantees that the contractivity condition (2.13) is satisfied. We end up with a system of 3 linear equations in the unknowns d_1, d_2, d_3, that has the unique solution

$$d_1 = -2 + \frac{1}{c_2}\ ,\ \ d_2 = -\frac{1}{c_2}\ ,\ \ d_3 = 3\ .$$

With these parameters the half-explicit method has order $q = 3$, ($\hat{s} = s = 3$).

c) $q = 4$. An explicit Runge–Kutta method with $p = s = 4$ satisfies condition D(1) and thus $c_4 = 1$ (since $b_4(1 - c_4) = \sum b_j a_{j4} = 0$). The methods that fulfill the simplifying condition C(2) form a family with the free parameter c_2 and have coefficients with

$$b_2 = \sum\nolimits_j b_j a_{j2} = 0\ ,\ \ c_3 = \frac{1}{2}$$

([11, p. 138]). If $\hat{s} = s = 4$ then the order conditions (3.4), (3.5) for $r = q - 2 = 2$ and the reciproque condition C(2)R imply

$$\sum_{i=1}^{4} \gamma_i = 1 \quad , \quad \frac{1}{2}\sum_{i=1}^{4} \gamma_i \hat{c}_{i+1} = 1 \quad , \quad \frac{1}{3}\sum_{i=1}^{4} \gamma_i \hat{c}_{i+1}^2 = 1 \tag{3.7}$$

with

$$\gamma_i := \sum\nolimits_j d_j w_{ji} c_{i+1}\ ,\ \ (i = 1(1)4)\ .$$

(The first equation in (3.7) follows from

$$\sum\nolimits_i w_{ji} c_{i+1} = \sum\nolimits_{i,k} w_{ji} a_{i+1,k} = \sum\nolimits_k \delta_{jk} = 1\ .)$$

The solutions of (3.7) are

$$\gamma_1 = 1\ ,\ \ \gamma_2 = -4\ ,\ \ \gamma_3 + \gamma_4 = 4$$

(note, that $c_4 = c_5 = 1$). Because of $\gamma_1 = c_2 d^T W \boldsymbol{e_1}$ a method with

$$\hat{s} = s = 4\ ,\ \ P(t)\delta y_h(t) = \mathcal{O}(h^5)\ ,\ \ \delta z_h(t) = \mathcal{O}(h^3)$$

can therefore not satisfy the contractivity condition (2.13). With the same arguments it is proved that the assumptions of Theorem 2.2 with $q = 4$ can not be satisfied adding a 5th stage with $c_6 = \sum_i a_{6i} = 1$. Thus we add a stage with $c_6 \neq 1$ and define

$$z_{n+1} := Z_{n5} \ , \quad \text{i. e.} \ \ d_i := 0 \ , \ (i = 1(1)4) \ , \quad d_5 := 1 \ .$$

With this definition of z_{n+1} conditions (3.4), (3.6) are trivial and the last call of f in time step $t_n \longrightarrow t_{n+1}$ is identical with the first call of f in time step $t_{n+1} \longrightarrow t_{n+2}$ ("first same as last technique" (FSAL) $f(Y_{n,s+1}, Z_{n,s+1}) = f(y_{n+1}, z_{n+1})$). We choose the parameters a_{6i} such that (see C(2) and (3.5))

$$\text{(3.8)} \quad \sum\nolimits_j a_{6j} c_j^l = \frac{1}{l+1} c_6^{l+1} \, , \ (l = 0,1) \ , \quad \frac{1}{3} \sum\nolimits_j w_{5j} \hat{c}_{j+1}^3 = 1 \ , \quad w_{51} = 0 \ ,$$

then the method is convergent with order $q = 4$ (because of $c_2 d^T W \boldsymbol{e_1} = c_2 w_{51}$ and $w_{51} = 0$ Theorem 2.2 is applicable). Eqs. (3.8) have a solution with $a_{65} \neq 0$ whenever $c_6 \notin \{0, \frac{1}{2}, 1\}$, free parameters are a_{62} and c_6.
We choose c_6 such that parameters a_{6j} with large $|a_{65}|$ are obtained. This is motivated by the system (2.3) of nonlinear equations that has a Jacobian of the form $h a_{i,i-1} [g_y f_z](y(t_n), z(t_n)) + \mathcal{O}(h)$. For simplicity we assume $a_{62} = b_2 = 0$ and get

$$a_{61} = \frac{1}{6} - \frac{1}{108}\sqrt{3} \ , \quad a_{63} = \frac{1}{3} - \frac{4}{27}\sqrt{3} \ , \quad a_{64} = -\frac{7}{108}\sqrt{3} \ , \quad a_{65} = \frac{1}{18}\sqrt{3} \ .$$

d) $q = 5$. An explicit Runge–Kutta method of order $p = 5$ has $s \geq 6$ stages, 17 order conditions have to be satisfied. We therefore restrict ourselves to the 5th order method of Dormand and Prince that is known to be a very efficient method for ODEs ([8], [11, p. 178f]). This method has $s = 6$ stages and satisfies the simplifying conditions C(3), D(1). The parameters a_{ij}, $(i = 2(1)6,\ j = 1(1)i-1)$ and b_j, $(j = 1(1)6)$ are given in Table II.5.2 of [11].
To get a half-explicit method of order $q = 5$ the local error in the algebraic components has to be of order $\delta z_h(t) = \mathcal{O}(h^4)$, i. e. the coefficients d_i, $(i = 1(1)\hat{s})$ have to satisfy the 8 order conditions (3.4), (3.5), (3.6) for $r = 3$, that are linear in d_i. Straightforward computation shows that these 8 conditions can not be fulfilled simultaneously if $\hat{s} = s = 6$, (at least) one new stage has to be added. As for the 4th order method we use the FSAL-technique, set

$$z_{n+1} := Z_{n,s+1} = Z_{n7}$$

and define parameters a_{8i}, $(i = 1(1)7)$ by the simplifying condition C(3), by the order conditions

$$\text{(3.9)} \qquad \sum\nolimits_j w_{7j} c_{j+1} a_{j+1,2} = 0 \ , \quad \frac{1}{4} \sum\nolimits_j w_{7j} \hat{c}_{j+1}^4 = 1$$

and by $0 = d^T W \boldsymbol{e_1} = w_{71}$. The first 6 rows of $\tilde{A}$ contain only parameters of the underlying explicit Runge–Kutta method. Because of the lower triangular structure of $\tilde{A}$ the elements w_{ij}, $(i \leq 6)$ are independent of the additional coefficients

a_{8i}, $(i = 1(1)7)$ and we have (note, that $\tilde{A}W = I$)

$$a_{87}w_{7j} = \sum_{i=1}^{7} a_{8i}w_{ij} - \sum_{i=1}^{6} a_{8i}w_{ij} = -\sum_{i=1}^{6} a_{8i}w_{ij} \ , \quad (j = 1(1)6)$$

and $a_{87}w_{77} = 1$. Multiplying $w_{71} = 0$ and the equations in (3.9) by a_{87} we thus get an equivalent set of conditions that are *linear* in a_{8i}, $(i = 1(1)7)$. For any given parameters a_{87}, c_8 with $c_8 \neq 1$ these conditions determine uniquely parameters $a_{81}, a_{82}, \dots , a_{86}$. In numerical tests we found it useful to choose a_{87} and c_8 such that $|a_{81}|, \dots , |a_{87}|$ and $1/|a_{87}|$ are not too large and the coefficients in the leading error term of $\delta z_h(t)$ are small. With

$$c_8 = \tfrac{19}{20} \ , \quad a_{87} = -\tfrac{3280}{75413}$$

we get the parameters

$$a_{81} = -\tfrac{18611506045861}{19738176307200} \ , \quad a_{82} = \tfrac{59332529}{14479296} \ , \quad a_{83} = -\tfrac{2509441598627}{893904224850} \ ,$$

$$a_{84} = \tfrac{2763523204159}{3289696051200} \ , \quad a_{85} = -\tfrac{41262869588913}{116235927142400} \ , \quad a_{86} = \tfrac{46310205821}{287848404480}$$

that are used in the half-explicit integrator HEDOP5 (see Section 4).

Type B methods that are based on the 5th order method of Dormand and Prince are especially advantageous because of the bound

$$\text{(3.10)} \quad \|y_m - y(t_m)\| = \mathcal{O}(1)(\, m \cdot \max_{0 \le n < m} \|\delta y_h(t_n)\| + h^r \max_{0 \le n < m} \|\delta z_h(t_n)\| \,)$$

that is satisfied for all $t_m = t_0 + mh \in [t_0, t_e]$ with $r = 2$, $\delta y_h(t_n) = \mathcal{O}(h^6)$ and $\delta z_h(t_n) = \mathcal{O}(h^4)$. Estimate (3.10) with $r = 1$ could be obtained with the techniques that are used in the proof of Theorem 2.2. However, to prove (3.10) with $r = 2$ the error propagation has to be studied much more in detail ([1, Corollary 3]). Estimate (3.10) with $r = 2$ shows that for HEDOP5 the influence of the local error $\delta z_h(t_n)$ on the global error $y_m - y(t_m)$ is in general negligible since $h^2\|\delta z_h(t_n)\| = \mathcal{O}(h^6)$ and $\|y_m - y(t_m)\| = \mathcal{O}(h^5)$.

4 Stepsize control, implementation and numerical tests.

In the present paper we concentrate on the convergence analysis of Type B methods, the details of an efficient implementation will be discussed somewhere else. The 5th order Type B method with 7 stages that was constructed in Example 3.1d has been implemented as integrator HEDOP5 for non-stiff constrained mechanical systems (**H**alf–**E**xplicit integrator based on the **5**th order method of **DO**rmand and **P**rince)[1]. Implementing HEDOP5 we could use many parts of the code MHERK5 by Simeon. MHERK5 is readily available in the software library MBSPACK ([18]), it is an implementation of the 5th order Type A method HEM5 ([3]).

[1]available in the Internet at
`ftp://ftp.mathematik.th-darmstadt.de/pub/department/software/mbspack/hedop5.f.`

In this section we consider the problem of stepsize control in HEDOP5 and the application to constrained mechanical systems. Results of a numerical test prove the efficiency of half-explicit methods with an explicit stage.

Stepsize control

In the 5th order explicit Runge–Kutta method of Dormand and Prince an embedded 4th order method is used to control the stepsize. In the notation of (2.1) a vector $\tilde{y}_{n+1}$ is defined by

$$\tilde{y}_{n+1} := y_n + \sum_{j=1}^{6} \hat{b}_j f(Y_{nj}, Z_{nj}) + h\hat{b}_7 f(y_{n+1}, z_{n+1}) \tag{4.1}$$

such that $\tilde{y}_{n+1} - y_{n+1}$ approximates the local error in the time step $t_n \to t_{n+1}$ (the coefficients $\hat{b}_j$ are given in [11, Table II.5.2]). Following an idea of Murua ([15]) the same coefficients $\hat{b}_j$ are used to define the vector

$$\hat{y}_{n+1} := \tilde{y}_{n+1} - [f_z(g_y f_z)^{-1}](\eta, \zeta) \cdot g(\tilde{y}_{n+1}) \tag{4.2}$$

in the stepsize control algorithm of HEDOP5. In (4.2) we have $\tilde{y}_{n+1}$ from (4.1) and vectors η, ζ with $\|\eta - y_n\| = \mathcal{O}(h)$, $\|\zeta - z_n\| = \mathcal{O}(h)$. In (4.1) the function value $f(Y_{n7}, Z_{n7})$ of the 7th stage of HEDOP5 is re-used as $f(y_{n+1}, z_{n+1})$ (see Example 3.1d).

Using the tree model the order of $\tilde{y}_{n+1}$ and $\hat{y}_{n+1}$ is analysed. In contrast to the ODE case $\tilde{y}_{n+1}$ gives only a 3rd order approximation to $y(t_{n+1})$ but with $\hat{y}_{n+1}$ from (4.2) we get again a 4th order approximation to $y(t_{n+1})$ such that $\hat{y}_{n+1} - y_{n+1}$ can be used as approximation to $\delta y_h(t_n)$. With this approximation standard stepsize control strategies from ODE-theory can be carried over to the Type B method HEDOP5 because the influence of $\delta z_h(t_n)$ on the global error in y is negligible (see Example 3.1d). The stepsize control in HEDOP5 is similar to the one that is proposed for the ODE–method DOPRI5 in the Appendix of [11]. Just as other integrators for index-2 systems HEDOP5 does not consider the global error in z in the stepsize control algorithm.

Application to constrained mechanical systems

Half-explicit methods are expected to be much more efficient than fully implicit methods if a non-stiff differential-algebraic system (1.1) is of the special form ($y = (q^T, u^T, w^T)^T$, $z = (a^T, \lambda^T)^T$)

$$\begin{aligned} q'(t) &= u(t) \\ u'(t) &= a(t) \\ w'(t) &= M(q(t))a(t) - \hat{f}(q(t), u(t)) + G^T(q(t))\lambda(t) \\ 0 &= w(t) \\ 0 &= G(q(t))u(t) \end{aligned} \tag{4.3}$$

since Equations (2.3) are *linear* in this case ([3]). Systems of the form (4.3) arise as model equations of constrained mechanical systems if the original holonomic constraints $\hat{g}(q) = 0$ are substituted by the equations

$$0 = \frac{d}{dt}\hat{g}(q(t)) = \frac{\partial}{\partial q}\hat{g}(q)q'(t) = G(q)u$$

(index-2 formulation, [12, p. 463ff]). Here q and u denote position and velocity coordinates, the Lagrangian multipliers λ couple the constraints to the dynamical equations, the Jacobian $G(q) := \frac{\partial}{\partial q}\hat{g}(q)$ is supposed to have full rank. $\hat{f}(q,u)$ are the applied forces and $M(q)$ the (symmetric) positive semi-definite mass matrix. In (4.3) the accelerations $a(t)$ and the artificial variables $w(t) \equiv 0$ could be eliminated straightforwardly. They have been introduced to keep the semi-explicit structure of (1.1). With the notations of (4.3) the index-2 condition (1.2) reads

$$\begin{pmatrix} M(q) & G^T(q) \\ G(q) & 0 \end{pmatrix} \quad \text{is non-singular,}$$

it is satisfied iff $G(q)\xi = 0$ and $\xi \neq 0$ implies $\xi^T M(q)\xi > 0$.

In the numerical tests we compare 4 half-explicit integrators that are tailored to the simulation of constrained mechanical systems:

HEM5 (Brasey [3]). The most efficient Type A method from the literature.

HEDOP5. The new integrator based on the Type B method of Example 3.1d.

MDOP5 (Simeon [18]). The application of the 5th order ODE–method of Dormand and Prince ([11, Appendix]) to the index-1 formulation of the model equations that is obtained from (4.3) if $0 = \frac{d}{dt}\hat{g}(q) = G(q)u$ is substituted by

$$0 = \frac{d^2}{dt^2}\hat{g}(q(t)) = \hat{g}_{qq}(q(t))(u(t), u(t)) + G(q(t))a(t)$$

([12, p. 465]).

PHEM56 (Murua [15]). A partitioned half-explicit Runge–Kutta (PHERK) method for (4.3) that is also based on the 5th order method of Dormand and Prince. PHERK methods like PHEM56 can be interpreted as generalizations of Type B methods, they are given by (2.1)/(2.2) with $i_0 = 3$, $\hat{s} \geq s$ if in (2.1) the equations $g(Y_{ni}) = 0$ are substituted by

$$g(\eta_{ni}) = 0 \quad \text{with} \quad \eta_{ni} = y_n + h\sum_{j=1}^{i-1} \gamma_{ij} f(Y_{nj}, Z_{nj})\ . \tag{4.4}$$

Type B methods satisfy (4.4) with $\eta_{ni} = Y_{ni}$ and $\gamma_{ij} = a_{ij}$. Compared with this special case there are additional free parameters γ_{ij} in general PHERK methods. This fact results in additional degrees of freedom in the order conditions that could be used to reduce the number of stages or the size of the leading error term in $\delta y_h(t)$ or $\delta z_h(t)$.

However, in the application to constrained mechanical systems most of these benefits are lost because of the very special structure of (4.3): In the notation of (1.1) and (4.3) the matrix $G(q)$ is part of the constraints $g(y) = 0$ and $G^T(q)$ is part of $f(y, z)$. In Type B methods both $f(y, z)$ and $g(y)$ are evaluated for the stage vectors Y_{ni}, Z_{ni} such that some evaluations of G^T can be saved in the computational process. These savings are not possible for PHERK methods of the general form (4.4) such that PHEM56 needs (compared with HEDOP5) 4 additional evaluations of G per step of integration.

	calls of					order	
integrator	f	G	$\hat{g}_{qq}$	DEC	SOL	in y	in z
HEM5	8	8	0	8	8	5	3
HEDOP5	6	8	0	6	7	5	4
MDOP5	6	6	6	6	6	5	5
PHEM56	6	12	0	6	7	5	4

Table 4.1: Numerical effort of half-explicit integrators (per step of integration); DEC ... matrix decompositions, SOL ... number of systems (2.3) of linear equations that are to be solved.

In the numerical tests we used codes from the package MBSPACK ([18]): MDOP5 for the index-1 formulation and various modifications of MHERK5 for the index-2 formulation (implementing HEM5, HEDOP5, and PHEM56). Common features of all 4 half-explicit integrators include a dense output option and projection steps to avoid a drift off the manifold $\{ q : \hat{g}(q) = 0 \}$ ([18]). In Table 4.1 we compare the essential numerical effort of the integrators (including the additional effort for stepsize control, cf. (4.2)).

Numerical tests

The half-explicit integrators were compared for various non-stiff benchmark problems from the literature. Typically all methods of Table 4.1 are substantially faster than a fully implicit code like DASSL ([6]). The 3 integrators that are based on the method of Dormand and Prince (HEDOP5, MDOP5, PHEM56) reach the robustness and the high efficiency that is characteristic of the ODE–method DOPRI5. They need up to 40% less cpu-time than HEM5 to compute a solution of the same accuracy.

The differences between HEDOP5, MDOP5, and PHEM56 are smaller and depend on the problem that is to be solved. If the evaluation of $\hat{g}_{qq}$ is time-consuming then HEDOP5 and PHEM56 are superior, otherwise MDOP5 is the fastest integrator. The savings in the evaluation of G^T make the Type B method HEDOP5 typically 5%–10% faster than the PHERK method PHEM56.

The results for a typical benchmark problem are summarized in Fig. 4.1. In this benchmark a wheel suspension is described by $n_q = 14$ position coordinates that have to satisfy $n_\lambda = 12$ holonomic constraints $\hat{g} = 0$ (for details see [19]).

We applied the integrators with various tolerances using RTOL $= 10^{-4-j/8}$, ($j = 0, 1, \ldots, 56$) as bound for the relative error and ATOL $= 0.1$ RTOL as bound for the absolute error. Fig. 4.1 shows in double logarithmic scale the cpu-time on a SUN Sparc5 workstation versus the obtained accuracy. The markers indicate the results for the "integer tolerances" 10^{-4}, 10^{-5}, $\ldots$. The results for the integrators that use the index-2 formulation (4.3) of the model equations are summarized in the upper plots. The lower plots show the results for HEDOP5, MDOP5, and for DASSL, that was applied to the Gear–Gupta–Leimkuhler formulation of the model equations ([12, p. 465]).

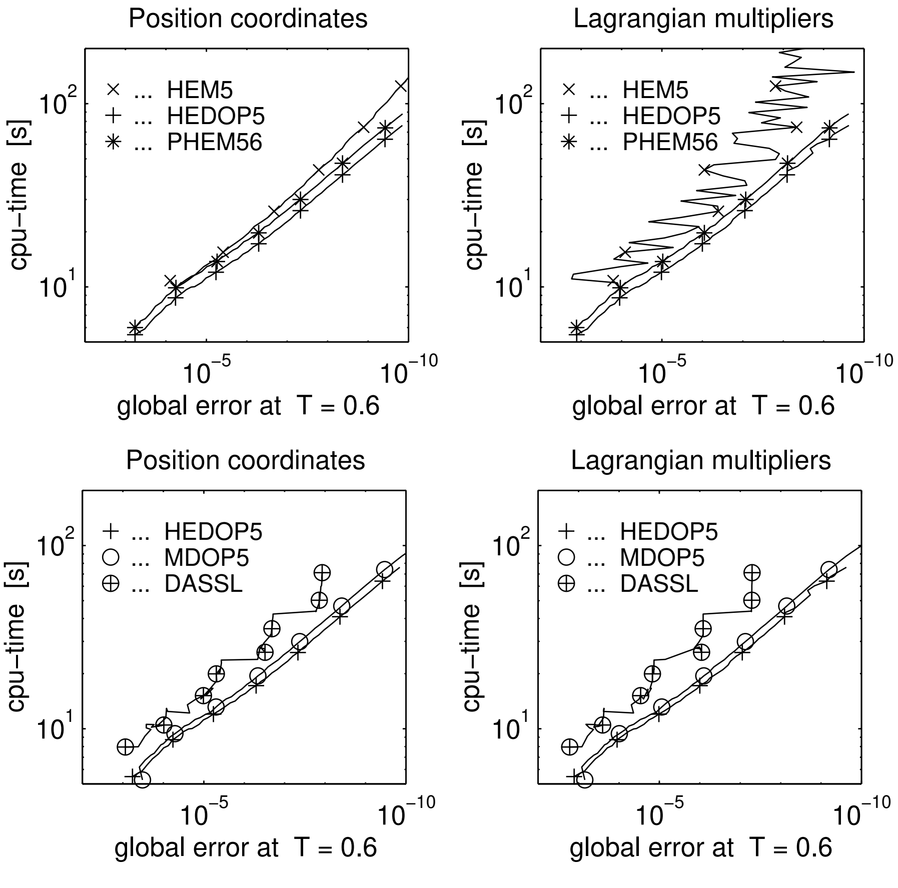


Figure 4.1: Work–precision diagrams for the benchmark "Wheel suspension" ([19]). Upper plots: HEM5, HEDOP5, PHEM56; lower plots: HEDOP5, MDOP5, DASSL.

It is well known that the evaluation of $\hat{g}_{qq}$ is very expensive in this benchmark problem. Therefore HEDOP5 is the most efficient code for this example. The differences between MDOP5 and PHEM56 are negligible, all 3 codes are faster than HEM5. Finally, the larger cpu-times for DASSL underline once again the benefits of half-explicit integrators for non-stiff problems.

5 Summary.

We constructed a new class of half-explicit methods ("Type B methods") for differential-algebraic systems of index 2 that is based on the half-explicit Runge–Kutta methods of Hairer et al. ([10], "Type A methods") and differs from these methods in the substitution of the first stage by an explicit Runge–Kutta stage. With this modification well known high order explicit Runge–Kutta methods for ODEs can be extended to half-explicit methods for differential-algebraic systems without any order reduction. Furthermore the construction of higher order methods is simplified since most of the order conditions coincide with classical order conditions for the underlying explicit Runge–Kutta method.

In view of the recent development of half-explicit methods the Type B methods can be seen as a class of partitioned half-explicit Runge–Kutta methods ([15]) that has special advantages in the application to constrained mechanical systems. Type B methods are convergent with the same order as the underlying explicit Runge–Kutta method if the local discretization error in the algebraic components is sufficiently small and a contractivity condition is satisfied. For the approximation of the algebraic components the methods can be extended by additional stages.

Based on explicit Runge–Kutta methods from the literature Type B methods up to order $q = 5$ are constructed. Currently, there are two half-explicit Runge–Kutta methods with an explicit first stage, that are based on the 5th order explicit Runge–Kutta method of Dormand and Prince: PHEM56 of Murua ([15]) and the Type B method HEDOP5. Both methods are seen to be very efficient: they have 7 stages and require the solution of 6 systems (2.3) of n_z nonlinear equations per step of integration. In numerical tests for the index-2 formulation of model equations for constrained mechanical systems both HEDOP5 and PHEM56 are superior to the most efficient Type A method that is known from the literature (HEM5). There are minor differences between HEDOP5 and PHEM56 making HEDOP5 a little bit more efficient in most of the applications.

Acknowledgement I am grateful to Prof. Dr. K. Strehmel and to J. Wensch (both Halle/S.) for helpful comments on a first version of this manuscript. Furthermore I gratefully acknowledge the support of A. Murua (San Sebastian) and B. Simeon (Darmstadt) in the implementation and in the tests of HEDOP5.

REFERENCES

1. M. Arnold, *Zur Theorie und zur numerischen Lösung von Anfangswertproblemen für differentiell-algebraische Systeme von höherem Index.* Universität Rostock, Mathematisch–Naturwissenschaftliche Fakultät – Habilitationsschrift, Dezember 1996.

2. U. ASCHER AND P. LIN, *Sequential regularization methods for higher index DAEs with constraint singularities. The linear index-2 case*, SIAM J. Numer. Anal., 33 (1996), pp. 1921–1940.
3. V. BRASEY, *A half-explicit method of order 5 for solving constrained mechanical systems*, Computing, 48 (1992), pp. 191–201.
4. V. BRASEY AND E. HAIRER, *Half-explicit Runge–Kutta methods for differential-algebraic systems of index 2*, SIAM J. Numer. Anal., 30 (1993), pp. 538–552.
5. ———, *Symmetrized half-explicit methods for constrained mechanical systems*, Appl. Numer. Math., 13 (1993), pp. 23–31.
6. K. BRENAN, S. CAMPBELL, AND L. PETZOLD, *Numerical solution of initial–value problems in differential–algebraic equations*, SIAM, Philadelphia, 2nd ed., 1996.
7. J. BUTCHER, *The numerical analysis of ordinary differential equations: Runge–Kutta and general linear methods*, Wiley&Sons, Chichester New York et al., 1987.
8. J. DORMAND AND P. PRINCE, *A family of embedded Runge–Kutta formulae*, J. Comp. Appl. Math., 6 (1980), pp. 19–26.
9. C. GEAR, B. LEIMKUHLER, AND G. GUPTA, *Automatic integration of Euler–Lagrange equations with constraints*, J. Comp. Appl. Math., 12&13 (1985), pp. 77–90.
10. E. HAIRER, C. LUBICH, AND M. ROCHE, *The numerical solution of differential-algebraic systems by Runge–Kutta methods*, Lecture Notes in Mathematics, 1409, Springer-Verlag, Berlin Heidelberg New York, 1989.
11. E. HAIRER, S. NØRSETT, AND G. WANNER, *Solving Ordinary Differential Equations. I. Nonstiff Problems*, Springer-Verlag, Berlin Heidelberg New York, 2nd ed., 1993.
12. E. HAIRER AND G. WANNER, *Solving Ordinary Differential Equations. II. Stiff and Differential-Algebraic Problems*, Springer-Verlag, Berlin Heidelberg New York, 2nd ed., 1996.
13. C. LUBICH, *h^2-extrapolation methods for differential-algebraic systems of index 2*, Impact Comput. Sc. Eng., 1 (1989), pp. 260–268.
14. C. LUBICH AND M. ROCHE, *Rosenbrock methods for differential-algebraic systems with solution dependent singular matrix multiplying the derivative*, Computing, 43 (1990), pp. 325–342.
15. A. MURUA, *Partitioned half-explicit Runge–Kutta methods for differential-algebraic systems of index 2*, Computing, 59 (1997), pp. 43–61.
16. J. ORTEGA AND W. RHEINBOLDT, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York London, 1970.
17. A. OSTERMANN, *A class of half-explicit Runge–Kutta methods for differential-algebraic systems of index 3*, Appl. Numer. Math., 13 (1993), pp. 165–179.

18. B. SIMEON, *MBSPACK – Numerical integration software for constrained mechanical motion*, Surveys on Mathematics for Industry, 5 (1995), pp. 169–202.

19. ———, *On the numerical solution of a wheel suspension benchmark problem*, Comp. Appl. Math., 66 (1996), pp. 443–456.

20. S. WOLFRAM, *Mathematica. A System for Doing Mathematics by Computer*, Addison Wesley, Redwood City, 2nd ed., 1991.