Electronic Transactions on Numerical Analysis.
Volume 48, pp. 131–155, 2018.
Copyright © 2018, Kent State University.
ISSN 1068–9613.
DOI: 10.1553/etna_vol48s131

RUNGE-KUTTA METHODS REVISITED FOR A CLASS OF STRUCTURED STRANGENESS-FREE DIFFERENTIAL-ALGEBRAIC EQUATIONS[*]

VU HOANG LINH[†] AND NGUYEN DUY TRUONG[‡]

Abstract. Numerical methods for a class of nonlinear differential-algebraic equations (DAEs) of the strangeness-free form are investigated. Half-explicit and implicit Runge-Kutta methods are revisited as they are applied to a reformulated form of the original DAEs. It is shown that the methods preserve the same convergence order and the same stability properties as if they were applied to ordinary differential equations (ODEs). Thus, a wide range of explicit Runge-Kutta methods and implicit ones, which are not necessarily stiffly accurate, can efficiently solve the class of DAEs under consideration. Implementation issues and a perturbation analysis are also discussed. Numerical experiments are presented to illustrate the theoretical results.

Key words. differential-algebraic equation, strangeness-free form, Runge-Kutta method, half-explicit method, convergence, stability

AMS subject classifications. 65L80, 65L05, 65L06, 65L20

1. Introduction.

In this paper, we consider the initial value problem (IVP) for nonlinear differential-algebraic equations (DAEs) of the structured form

$$\begin{aligned} f\big(t, x(t), E(t)x'(t)\big) &= 0, \\ g\big(t, x(t)\big) &= 0, \end{aligned} \tag{1.1}$$

on a compact interval $\mathbb{I} = [t_0, T] \subset \mathbb{R}$, where $x \in C^1(\mathbb{I}; \mathbb{R}^m)$, $E \in C^1(\mathbb{I}; \mathbb{R}^{m_1,m})$, and an initial condition $x(t_0) = x_0$ is given, which is supposed to be consistent. We assume that $f = f(t,u,v) : \mathbb{I} \times \mathbb{R}^m \times \mathbb{R}^{m_1} \to \mathbb{R}^{m_1}$ and $g = g(t,u) : \mathbb{I} \times \mathbb{R}^m \to \mathbb{R}^{m_2}$, $m_1 + m_2 = m$, are sufficiently smooth functions with bounded partial derivatives. Furthermore, we assume that the unique solution of the IVP for (1.1) exists and that

$$\begin{bmatrix} f_v E \\ g_u \end{bmatrix} \quad \text{is nonsingular along the exact solution } x(t). \tag{1.2}$$

Here f_v and g_u denote the Jacobian of f with respect to v and that of g with respect to u, respectively. In the whole paper, unless confusion can arise, we will not display the variable(s) of the functions explicitly.

The system (1.1) is a special case of DAEs of the form

$$\begin{aligned} \bar{f}(t, x, x') &= 0, \\ \bar{g}(t, x) &= 0, \end{aligned} \tag{1.3}$$

where $\bar{f} : \mathbb{I} \times \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^{m_1}$ and $\bar{g} : \mathbb{I} \times \mathbb{R}^m \to \mathbb{R}^{m_2}$ are sufficiently smooth functions with bounded partial derivatives and

$$\begin{bmatrix} \bar{f}_{x'}(t,x,x') \\ \bar{g}_x(t,x) \end{bmatrix} \quad \text{is nonsingular along the exact solution } x(t). \tag{1.4}$$

DAEs of the form (1.3) satisfying (1.4) are said to be of strangeness-free form; see [13]. Numerical solutions by collocation methods and BDF methods are proposed in [13], which

[*] Received September 29, 2016. Accepted April 15, 2018. Published online on May 21, 2018. Recommended by Marlis Hochbruck. This work was supported by NAFOSTED project 101.02-2017.314.

[†] Faculty of Mathematics, Mechanics and Informatics, Vietnam National University, 334 Nguyen Trai, Thanh Xuan, Hanoi, Vietnam (linhvh@vnu.edu.vn).

[‡] Tran Quoc Tuan University, Son Tay, Hanoi, Vietnam (truong.nguyenduy80@gmail.com).

generalize those for semi-explicit index-1 DAEs; see [3, 8, 10]. For DAEs in general, the most popular one-step methods are implicit Runge-Kutta (IRK) methods, which are stiffly accurate; see [8, 10, 13, 15]. For semi-explicit DAEs, half-explicit Runge-Kutta (HERK) methods are also proven to be efficient in certain cases; see [1, 8].

Though efficient numerical methods and software packages have already been fairly well developed for general DAEs of lower index, the problem that motivates us to find efficient methods for solving structured DAEs of the form (1.1) arises when we propose QR- and SVD-based algorithms for approximating spectral intervals for linear DAEs; see [16, 18]. In the course of approximating certain stability characteristics, we are to integrate matrix-valued semi-linear DAEs on usually very long intervals, which form a special case of (1.1) and (1.3). The use of half-explicit methods are extended to DAEs of the form (1.3) in [17]. It turns out that for the class of matrix-valued DAEs investigated in [16, 18], the half-explicit methods are significantly cheaper than the well-known implicit methods. However, it is also shown there that DAEs (1.3) can be transformed into semi-explicit index-2 DAEs by a rearrangement and a partitioning of the variables. This explains why the standard half-explicit Runge-Kutta methods applied directly to (1.3) unfortunately suffer from order reduction with the exception of low-order methods; see [17].

In this paper, we exploit the special structure of the DAEs (1.1) and show that a wider class of Runge-Kutta methods are applicable. In particular, we demonstrate that after reformulating the DAEs (1.1) in a very simple and obvious way, discretizations by Runge-Kutta methods are essentially the same as those for the semi-explicit index-1 DAEs (2.1). Thus, all the convergence and stability results of Runge-Kutta methods well-known for ODEs (see [2, 9]) are preserved. The idea of applying (implicit) Runge-Kutta methods to a reformulated form instead of DAEs of standard form was first proposed in [11, 12], and it is shown that the modified discretization schemes possess better stability properties for *index-1 DAEs in the so-called numerically qualified form*. This approach is well discussed in the context of properly formulated DAEs in [15]. Extensions of this idea to fully implicit index-2 DAEs are also investigated in [4, 7]. In this paper, the same reformulating trick is used. However, avoiding the projector-based decoupling as in [11, 12], we use rather a very simple transformation to show that the discretization schemes applied to the reformulated DAEs are essentially equivalent to those proposed for semi-explicit index-1 DAEs. Roughly speaking, in this approach the reduction to semi-explicit form and the discretization commute. This explains why the modified discretization schemes preserve all the order and stability properties. As a major novelty of our results, all explicit Runge-Kutta methods can be adapted without order reduction and stability loss. Furthermore, the same statement holds for implicit Runge-Kutta methods, which are not necessarily stiffly accurate, a property that is usually required in the DAE literature. Applying the modified Runge-Kutta methods to the test DAE introduced in [14], the stability function turns out to be the same as that for the test ODE. Another alternative approach for treating the instability is proposed in [14] for linear time-varying DAEs, where the so-called spin-stabilized transformation is used. While the spin-stabilized matrix function (together with its derivative) has to be approximated at each meshpoint, which is relatively costly, in our approach we do not have to evaluate the transformation matrix explicitly. The only extra cost comes from the evaluation of the derivative of the matrix function E, which is assumed to be available by either an analytic formula or an appropriate finite difference scheme.

The paper is organized as follows. In Section 2 we briefly review the use of Runge-Kutta methods for semi-explicit index-1 DAEs which is helpful for later investigations. We also show that, after a reformulation, the DAE (1.1) essentially is equivalent to a DAE in semi-explicit form. We analyze the sensitivity of solutions for the DAE (1.1) and for the reformulated form in the linear case. In Section 3 we propose half-explicit and implicit Runge-Kutta methods for the reformulated DAEs and discuss their convergence and stability. We also investigate the influence of computational errors. Numerical results given in Section 4 illustrate the theoretical results in Section 3. The paper is closed by some conclusions.

2. Preliminaries.

2.1. Runge-Kutta methods for semi-explicit index-1 DAEs.

Semi-explicit index-1 DAEs are the simplest DAEs of the form

$$\begin{aligned} y'(t) &= \Phi\big(t, y(t), z(t)\big), \\ 0 &= \Gamma\big(t, y(t), z(t)\big), \end{aligned} \tag{2.1}$$

on an interval $\mathbb{I} = [t_0, T]$. The initial value (y_0, z_0) is assumed to be consistent, i.e., $\Gamma(t_0, y_0, z_0) = 0$. Here, we assume that the functions $\Phi : \mathbb{I} \times \mathbb{R}^{m_1} \times \mathbb{R}^{m_2} \to \mathbb{R}^{m_1}$ and $\Gamma : \mathbb{I} \times \mathbb{R}^{m_1} \times \mathbb{R}^{m_2} \to \mathbb{R}^{m_2}$ are sufficiently smooth. Furthermore, it is assumed that the Jacobian

$$\Gamma_z\big(t, y(t), z(t)\big) \quad \text{is nonsingular in a neighborhood of the solution.} \tag{2.2}$$

The convergence result established for Runge-Kutta methods for the semi-explicit DAE (2.1) plays an important role in the analysis of the methods that we construct in this paper.

In order to construct numerical solutions, first we take a mesh $t_0 < t_1 < \cdots < t_N$. For the sake of simplicity, here we consider only uniform meshes with stepsize h. All the results and the proofs presented in this paper are extendable to the case of variable stepsizes. Suppose that the coefficients of an s-stage RK method of order p are given in a Butcher tableau

$$\begin{array}{c|c} c & A \\ \hline & b^T \end{array} \qquad \text{with} \quad A = [a_{ij}]_{s\times s}\,, \qquad b = [b_1\, b_2\, \ldots\, b_s]^T, \qquad c = [c_1\, c_2\, \ldots\, c_s]^T.$$

This method may be either explicit or implicit. On a sub-interval $[t_n, t_{n+1}]$ we suppose that the approximations $y_n \simeq y(t_n)$, $z_n \simeq z(t_n)$ are given. Let $Y_{ni} \simeq y(t_n + c_i h)$, $Z_{ni} \simeq z(t_n + c_i h)$ be the internal stage approximations. The s-stage RK scheme for the DAE (2.1) (in the direct approach) is written in the form

$$\begin{aligned} Y_{ni} &= y_n + h \sum_{j=1}^{s} a_{ij} \Phi(T_j, Y_{nj}, Z_{nj}), \\ 0 &= \Gamma(T_i, Y_{ni}, Z_{ni}), \qquad\qquad i = 1, 2, \ldots, s, \\ y_{n+1} &= y_n + h \sum_{i=1}^{s} b_i \Phi(T_i, Y_{ni}, Z_{ni}), \\ 0 &= \Gamma(t_{n+1}, y_{n+1}, z_{n+1}), \end{aligned} \tag{2.3}$$

where $T_i = t_n + c_i h$, $h = t_{n+1} - t_n$. If the original Runge-Kutta method is explicit, then the corresponding discretization is called half-explicit. The condition (2.2) implies that in a neighbourhood of the solution, we can solve $z = \chi(t, y)$ from the second equation of (2.1) by the Implicit Function Theorem. Thus (2.1) becomes

$$y'(t) = \widetilde{\Phi}(t, y), \tag{2.4}$$

where $\widetilde{\Phi}(t, y) = \Phi\big(t, y, \chi(t, y)\big)$. Next, we show that the y-component of the numerical solution of (2.3) is exactly the same as the numerical solution of the RK method applied to the ordinary differential equation (ODE) (2.4). Then, we have the following convergence result for the scheme (2.3); see [8, 10, 13].

THEOREM 2.1. *Assume that* (2.2) *holds in a neighbourhood of the solution* $\big(y(t), z(t)\big)$ *of* (2.1) *and the initial values are consistent. Given a Runge-Kutta method of order* p*, the Runge-Kutta scheme* (2.3) *applied to the DAE* (2.1) *is convergent of order* p*, i.e.,*

$$\|y_n - y(t_n)\| = \mathcal{O}(h^p), \quad \|z_n - z(t_n)\| = \mathcal{O}(h^p) \qquad \textit{for } t_n - t_0 = nh \le \textit{const}.$$

REMARK 2.2. If the original Runge-Kutta scheme is explicit, then the implementation of the method (2.3) is rather simple. We evaluate the stage Y_{ni} explicitly and then solve the algebraic equation for the stage Z_{ni} consecutively for $i = 1, 2, \dots, s$. Then, we calculate y_{n+1} and again solve the algebraic equation for z_{n+1}. The algebraic equations can be solved efficiently by Newton's method. The implementation of the implicit method is more complicated. First, we have to solve a large nonlinear system for Y_{ni} and Z_{ni}, $i = 1, 2, \dots, s$, simultaneously by Newton's method. If the last row of A and b^T are different, then we first evaluate y_{n+1} and then solve the algebraic equation for z_{n+1}. Otherwise, we set $y_{n+1} = Y_{ns}$ and $z_{n+1} = Z_{ns}$.

2.2. A reformulation. The main investigation of this paper is the numerical solution of DAEs of the form (1.1). We will exploit the structure of the problem to construct numerical methods which preserve the order as well as the stability properties of the ODE case. The reformulation in this subsection is presented as a motivation of our approach and for the analysis of the numerical methods but is not used for the implementation.

Due to the special structure, problem (1.1) can be rewritten into the form

$$\begin{aligned} f\big(t, x(t), (Ex)'(t) - E'(t)x(t)\big) &= 0, \\ g\big(t, x(t)\big) &= 0, \end{aligned} \tag{2.5}$$

on an interval $\mathbb{I} = [t_0, T]$. The condition (1.2) implies that $E(t)$ is a full row-rank matrix-valued function of size $m_1 \times m$, $(m_1 \le m)$ and

$$\operatorname{rank}(E(t)) = m_1 \quad \text{for all } t \in \mathbb{I}.$$

Due to the existence of a smooth QR factorization (see [6]) there exists a pointwise orthogonal matrix function $\widetilde{Q}$ such that $\widetilde{Q}^T\widetilde{Q} = I$, $E\widetilde{Q} = [E_{11} \quad 0]$, where E_{11} is an invertible lower triangular $m_1 \times m_1$ matrix. Let us define the matrix function

$$Q = \widetilde{Q}\begin{bmatrix} E_{11}^{-1} & 0 \\ 0 & I \end{bmatrix}. \tag{2.6}$$

Hence, we obtain $EQ = [I \quad 0]$. We introduce the change of variables $x = Qy$. Then, we have

$$(Ex)'(t) = (EQy)'(t) = \big([I \quad 0]y\big)'(t).$$

Therefore, the state y can be partitioned as $y = [y_1^T,\ y_2^T]^T$, where $y_1 \in C^1(\mathbb{I}, \mathbb{R}^{m_1})$, $y_2 \in C^1(\mathbb{I}, \mathbb{R}^{m_2})$. We obtain $(Ex)' = y_1'$. Hence, the DAEs (2.5) can be rewritten as

$$\begin{aligned} f\big(t, Qy, y_1' - E'Qy\big) &= 0, \\ g\big(t, Qy\big) &= 0. \end{aligned} \tag{2.7}$$

By invoking the Implicit Function Theorem, there exists a function $\tilde{f}$ such that the identity $y_1' - E'Qy = \tilde{f}(t, Qy)$ holds. Let us define $F\big(t, y_1, y_2, y'{}_1\big) = \tilde{f}(t, Qy) + E'Qy$ and $G\big(t, y_1, y_2\big) = g\big(t, Qy\big)$. The condition (1.2) together with the definition of Q implies that

$$\begin{bmatrix} f_v E \\ g_u \end{bmatrix} Q = \begin{bmatrix} f_v EQ \\ g_u Q \end{bmatrix} = \begin{bmatrix} f_v[I \quad 0] \\ g_u[Q^{(1)} \; Q^{(2)}] \end{bmatrix} = \begin{bmatrix} f_v & 0 \\ g_u Q^{(1)} & g_u Q^{(2)} \end{bmatrix}$$

is nonsingular along the solution. Here $Q = [Q^{(1)} \; Q^{(2)}]$ and $Q^{(1)} \in C^1(\mathbb{I}, \mathbb{R}^{m,m_1})$, $Q^{(2)} \in C^1(\mathbb{I}, \mathbb{R}^{m,m_2})$. Hence it follows that f_v and $g_u Q^{(2)}$ are invertible as well. Hence, the system (2.7) becomes

$$\begin{aligned} y_1' &= F(t, y_1, y_2), \\ 0 &= G(t, y_1, y_2), \end{aligned} \tag{2.8}$$

where the Jacobian $G_{y_2} = g_u Q^{(2)}$ is nonsingular. This is an index-1 DAE of semi-explicit form. Hence, the class of problems (1.1) can be solved efficiently by Runge-Kutta methods after it is transformed into the form (2.8). However, an explicit realization of this transformation is almost impossible in computational practice. We will first show that we can apply a Runge-Kutta scheme directly to the reformulated DAE (2.5), and then we prove that the discretization and the transformation are commutative. The latter means that essentially we apply the same Runge-Kutta method to the transformed DAE (2.8). The Runge-Kutta methods applied to the reformulated DAE (2.5) have the same order and the same stability properties as if they are applied to semi-explicit DAEs of index 1. Here, the reformulation plays a key role since we will (numerically) differentiate Ex instead of x. If we apply the same method to the original DAE (1.1), then a loss of accuracy order and/or stability may happen; see the illustrative numerical experiments and comparisons in Section 4.

2.3. Sensitivity analysis of solutions for linear strangeness-free DAEs. We will see that the sensitivity analysis of solutions for linear DAEs of the form (1.1) is completely different if we consider the reformulated form (2.5) instead of (1.1).

a) Consider the linear DAE

$$\begin{aligned} E_{11}(t)x_1'(t) + E_{12}(t)x_2'(t) &= A_{11}(t)x_1(t) + A_{12}(t)x_2(t) + q_1(t), \\ 0 &= A_{21}(t)x_1(t) + A_{22}(t)x_2(t) + q_2(t), \end{aligned} \tag{2.9}$$

where $E_{ij}, A_{ij} \in C(\mathbb{I}, \mathbb{R}^{m_i,m_j})$, $q_i \in C(\mathbb{I}, \mathbb{R}^{m_i})$, $i, j = 1, 2$, $m_1+m_2 = m$. The strangeness-free condition (1.2) requires that the matrix

$$\begin{bmatrix} E_{11} & E_{12} \\ A_{21} & A_{22} \end{bmatrix} \quad \text{be nonsingular for all } t \in \mathbb{I}. \tag{2.10}$$

By an appropriate rearrangement of the variables, we may assume that A_{22} is nonsingular. From the second equation of (2.9), we have

$$x_2 = -A_{22}^{-1} A_{21} x_1 - A_{22}^{-1} q_2. \tag{2.11}$$

Differentiating both sides of (2.11), we then obtain

$$x_2' = -A_{22}^{-1} A_{21} x_1' - \big(A_{22}^{-1} A_{21}\big)' x_1 - \big(A_{22}^{-1} q_2\big)'. \tag{2.12}$$

Substituting (2.11), (2.12) into the first equation of (2.9) yields

$$\overline{E}_{11} x_1' = \overline{A}_{11} x_1 + \overline{q}_1,$$

where

$$\overline{E}_{11} = E_{11} - E_{12}A_{22}^{-1}A_{21}, \qquad \overline{A}_{11} = A_{11} + E_{12}\big(A_{22}^{-1}A_{21}\big)' - A_{12}A_{22}^{-1}A_{21},$$
$$\overline{q}_1 = q_1 - A_{12}A_{22}^{-1}q_2 + E_{12}\big(A_{22}^{-1}q_2\big)'.$$

It is easy to check that $\overline{E}_{11}$ is nonsingular by (2.10). Consequently, we obtain the ODE

$$x_1' = B_1x_1 + r_1.$$

Here, B_1 and r_1 are defined as follows:

$$B_1 = \overline{E}_{11}^{-1}\overline{A}_{11}, \qquad r_1 = \overline{E}_{11}^{-1}\overline{q}_1 = \overline{E}_{11}^{-1}\Big(q_1 - A_{12}A_{22}^{-1}q_2 + E_{12}\big(A_{22}^{-1}q_2\big)'\Big).$$

By this analysis, the equations that we have just obtained appear as if the solution x depends on the derivative of $A_{22}^{-1}q_2$. However, this is not true as the next analysis shows.

b) Now, we consider the linear DAE in the reformulated form

$$\begin{aligned}\big(E_1x\big)'(t) &= \big(A_1(t) + E_1'(t)\big)x(t) + q_1(t),\\ 0 &= A_2(t)x(t) + q_2(t),\end{aligned}$$

for all $t \in \mathbb{I}$, where $x = [x_1^T,\ x_2^T]^T$, $E_1 = [E_{11}\ E_{12}]$, $A_1 = [A_{11}\ A_{12}]$, $A_2 = [A_{21}\ A_{22}]$. By introducing again the transformation $x(t) = Q(t)y(t) = Q(t)[y_1^T(t),\ y_2^T(t)]^T$ with Q defined by (2.6), we arrive at the system

$$\begin{aligned}y_1'(t) &= \tilde{A}_{11}(t)y_1(t) + \tilde{A}_{12}(t)y_2(t) + q_1(t),\\ 0 &= \tilde{A}_{21}(t)y_1(t) + \tilde{A}_{22}(t)y_2(t) + q_2(t),\end{aligned} \tag{2.13}$$

where $[\tilde{A}_{11}\ \tilde{A}_{12}] = \big(A_1 + E_1'\big)Q$, $[\tilde{A}_{21}\ \tilde{A}_{22}] = A_2Q$. From (2.10), it follows that $\tilde{A}_{22}$ is nonsingular. Therefore, the second equation of (2.13) leads to

$$y_2 = -\tilde{A}_{22}^{-1}\tilde{A}_{21}y_1 - \tilde{A}_{22}^{-1}q_2. \tag{2.14}$$

Substituting (2.14) into the first equation of (2.13) yields the so-called *essential underlying ODE* [5, 15]

$$y_1' = \Big(\tilde{A}_{11} - \tilde{A}_{12}\tilde{A}_{22}^{-1}\tilde{A}_{21}\Big)y_1 + q_1 - \tilde{A}_{12}\tilde{A}_{22}^{-1}q_2.$$

It is clearly seen that neither y nor $x = Qy$ depends on the derivative of any expression containing q_2. The above comparison suggests that it is more reasonable to consider the reformulated form (2.5) instead of (1.1).

3. Runge-Kutta methods for the reformulated DAE.

In this section we will analyze the use of half-explicit and implicit Runge-Kutta methods for the reformulated DAE (2.5).

3.1. Discretization by half-explicit Runge-Kutta schemes.

First, we propose half-explicit Runge-Kutta methods (HERK) for the reformulated DAE. We take an arbitrary explicit Runge-Kutta method, i.e., the coefficient matrix $A = [a_{ij}]$ associated with it is a strictly lower triangular matrix. Consider a sub-interval $[t_n, t_{n+1}]$, $h = t_{n+1} - t_n$, and assume that an approximation x_n to $x(t_n)$ is given. Let us introduce $T_i = t_n + c_ih$ and the stage approximations $U_i \simeq x(T_i)$, $K_i \simeq (Ex)'(T_i)$, $i = 1, 2, \ldots, s$. We assume in addition that the function values $E_i = E(T_i)$, $E'_i = E'(T_i)$ are available. Then, the s-stage HERK scheme for the DAEs (2.5) reads as follows

$$U_1 = x_n, \tag{3.1a}$$

$$E_i U_i = E(t_n)U_1 + h\sum_{j=1}^{i-1} a_{ij}K_j, \tag{3.1b}$$

$$0 = f\big(T_{i-1}, U_{i-1}, K_{i-1} - E'_{i-1}U_{i-1}\big), \tag{3.1c}$$

$$0 = g(T_i, U_i), \qquad i = 2, 3, \ldots, s, \tag{3.1d}$$

$$E(t_{n+1})x_{n+1} = E(t_n)U_1 + h\sum_{i=1}^{s} b_i K_i, \tag{3.1e}$$

$$0 = f\big(T_s, U_s, K_s - E'_s U_s\big), \tag{3.1f}$$

$$0 = g(t_{n+1}, x_{n+1}). \tag{3.1g}$$

To verify the feasibility of this s-stage HERK scheme, we consider the (nonlinear) system (3.1b), (3.1c), (3.1d) denoted as $\mathcal{H}_i(U_i, K_{i-1}) = 0$ at the i-th stage, assuming that the preceding values U_j and K_{j-1}, $1 \le j \le i-1$, are already determined and they approximate the corresponding exact values with $O(h)$ accuracy. We have

$$\frac{\partial \mathcal{H}_i}{\partial U_i} = \begin{bmatrix} E_i \\ 0 \\ g_u(T_i, U_i) \end{bmatrix}, \qquad \frac{\partial \mathcal{H}_i}{\partial K_{i-1}} = \begin{bmatrix} a_{i,i-1}hI_m \\ f_v(T_{i-1}, U_{i-1}, K_{i-1} - E'_{i-1}U_{i-1}) \\ 0 \end{bmatrix}.$$

Consider a neighborhood of the exact solution x and the derivative of Ex defined by

$$\Omega(h) = \left\{ [U_i^T \ \ K_{i-1}^T]^T \in \mathbb{R}^{m+m_1}, \|U_i - x(T_i)\| \le Ch, \ \|K_{i-1} - (Ex)'(T_{i-1})\| \le Ch \right\}$$

for some positive constant C. It is easy to see that the assumption (1.2) holds if and only if both f_v and $\begin{bmatrix} E \\ g_u \end{bmatrix}$ are nonsingular along the exact solution. One can verify without difficulty that for sufficiently small h, the Jacobian of $\mathcal{H}_i$ is boundedly invertible, i.e., it is invertible and the inverse as a function of h is bounded. The exact solution satisfies

$$\mathcal{H}_i\big(x(T_i), (Ex)'(T_{i-1})\big) = O(h).$$

By the Implicit Function Theorem, the system given by (3.1b), (3.1c), (3.1d) has a locally unique solution (U_i^*, K_{i-1}^*) that satisfies

$$\|U_i^* - x(T_i)\| = O(h), \qquad \|K_{i-1} - (Ex)'(T_{i-1})\| = O(h).$$

Similarly, the system (3.1e), (3.1f), (3.1g) has a locally unique solution (x_{n+1}^*, K_s^*) that satisfies

$$\|x_{n+1}^* - x(t_{n+1})\| = O(h), \qquad \|K_s - (Ex)'(T_s)\| = O(h).$$

These nonlinear systems can be solved approximately, e.g., by Newton's method.

If we assume in addition that $a_{i,i-1} \ne 0$, for $i = 2, \ldots, s$, and $b_s \ne 0$, then the computational cost for solving (3.1b)–(3.1g) is reduced by explicitly solving for K_{i-1} and K_s, respectively. The equations (3.1b) and (3.1e) yield

$$K_1 = \frac{E_2 U_2 - E(t_n)U_1}{ha_{21}},$$

and

$$K_{i-1} = \Big(\frac{E_iU_i - E(t_n)U_1}{h} - \sum_{j=1}^{i-2} a_{i,j}K_j\Big)\frac{1}{a_{i,i-1}}, \quad i = 3,\dots,s,$$

$$K_s = \Big(\frac{E(t_{n+1})x_{n+1} - E(t_n)U_1}{h} - \sum_{i=1}^{s-1} b_iK_i\Big)\frac{1}{b_s}.$$

At the i-th stage ($i = 2,\dots,s$), the approximation U_i can be determined from a nonlinear system $\mathcal{F}_i(U_i) = 0$ given by

$$\begin{aligned} 0 &= hf\Big(T_{i-1}, U_{i-1}, \big(\frac{E_iU_i - E(t_n)U_1}{h} - \sum_{j=1}^{i-2} a_{i,j}K_j\big)\frac{1}{a_{i,i-1}} - E'_{i-1}U_{i-1}\Big), \\ 0 &= g(T_i, U_i). \end{aligned} \tag{3.2}$$

Here, we suppose that $U_1, U_2, \dots, U_{i-1}$, $K_1, K_2, \dots, K_{i-2}$ are given sufficiently close to the exact values. The Jacobian matrix of $\mathcal{F}_i$ with respect to U_i is

$$\frac{\partial \mathcal{F}_i}{\partial U_i} = \begin{bmatrix} \frac{1}{a_{i,i-1}} f_v(T_{i-1}, U_{i-1}, K_{i-1} - E'_{i-1}U_{i-1})E_i \\ g_u(T_i, U_i) \end{bmatrix}. \tag{3.3}$$

For sufficiently small h, the system (3.2) has a locally unique solution U_i^*, which can be approximated by Newton's method.

Next, the approximation K_i is obtained. Finally, a unique solution x_{n+1} at the time step $t = t_{n+1}$ is determined by the system $\mathcal{G}_n(x_{n+1}) = 0$, which is written as

$$\begin{aligned} 0 &= hf\Big(T_s, U_s, \big(\frac{E(t_{n+1})x_{n+1} - E(t_n)x_n}{h} - \sum_{i=1}^{s-1} b_iK_i\big)\frac{1}{b_s} - E'_sU_s\Big), \\ 0 &= g(t_{n+1}, x_{n+1}), \end{aligned} \tag{3.4}$$

where $U_1, U_2, \dots, U_s$, $K_1, K_2, \dots, K_{s-1}$ are already obtained. Here the Jacobian

$$\frac{\partial \mathcal{G}_n}{\partial x_{n+1}} = \begin{bmatrix} \frac{1}{b_s} f_v(T_s, U_s, K_s - E'_sU_s)E(t_{n+1}) \\ g_u(t_{n+1}, x_{n+1}) \end{bmatrix} \tag{3.5}$$

is boundedly invertible for sufficiently small h. The locally unique solution x_{n+1}^* can be approximated by Newton's method as well.

REMARK 3.1. We note that the first equations of (3.2) and (3.4) are scaled by h. If we do not apply the scaling, then the first block rows of the Jacobians in (3.3) and (3.5) are multiplied by $1/h$, which could increase the condition numbers of the Jacobians, in particular when the stepsize h is very small. On the other hand, the scaling by h is natural since it helps to balance the factor $1/h$ in the first equations of (3.2) and (3.4) as it is done for ODEs. Thus, the formulations (3.2) and (3.4) are consistent with the formulas of the Runge-Kutta methods for ODEs. That is why we suggest the scaling by h to the first equations of (3.2) and (3.4).

3.2. Discretization by implicit Runge-Kutta schemes. The s-stage implicit Runge-Kutta (IRK) scheme applied to the DAEs (2.5) reads as follows:

$$E_iU_i = E(t_n)x_n + h\sum_{j=1}^{s} a_{ij}K_j, \tag{3.6a}$$

$$0 = f\big(T_i, U_i, K_i - E_i'U_i\big), \tag{3.6b}$$

$$0 = g(T_i, U_i), \qquad i = 1, 2, \ldots, s, \tag{3.6c}$$

$$E(t_{n+1})x_{n+1} = E(t_n)x_n + h\sum_{i=1}^{s} b_iK_i, \tag{3.6d}$$

$$0 = g(t_{n+1}, x_{n+1}). \tag{3.6e}$$

By a similar argument as in the case of the HERK methods, it can be shown that if x_n is given sufficiently close to the exact value and h is sufficiently small, then the large system (3.6a), (3.6b), (3.6c) is locally uniquely solvable for U_i and K_i, $i = 1, 2, \ldots, s$.

Now we show that if the IRK is such that its coefficient matrix A is invertible, then the system (3.6a), (3.6b), (3.6c) is reduced by explicitly solving for K_i, $i = 1, 2, \ldots, s$. The set of equations (3.6a) with $i = 1, 2, \ldots, s$ yields the linear system $(A \otimes I_{m_1})K = D$, where $K = [K_1^T\ K_2^T\ \ldots\ K_s^T]^T$ and $D = [D_1^T\ D_2^T\ \ldots\ D_s^T]^T$ with

$$D_i = \frac{E_iU_i - E(t_n)x_n}{h}, \qquad i = 1, 2, \ldots, s.$$

Let $W = [w_{ij}] = A^{-1}$, then we have

$$K_i = \sum_{j=1}^{s} w_{ij}D_j = \sum_{j=1}^{s} w_{ij}\frac{E_jU_j - E(t_n)x_n}{h}, \qquad i = 1, 2, \ldots, s.$$

Inserting these expressions into the equation of (3.6b), for convenience also multiplying both sides by h, (3.6b), (3.6c) yield the nonlinear system $\Phi_n(U) = 0$ of the form

$$\begin{aligned} 0 &= hf\Big(T_i, U_i, \sum_{j=1}^{s} w_{ij}\frac{E_jU_j - E(t_n)x_n}{h} - E_i'U_i\Big), \\ 0 &= g(T_i, U_i), \qquad i = 1, 2, \ldots, s, \end{aligned} \tag{3.7}$$

where $U = [U_1^T\ U_2^T\ \ldots\ U_s^T]^T$. Set

$$f_v^i = f_v\Big(T_i, U_i, \sum_{j=1}^{s} w_{ij}\frac{E_jU_j - E(t_n)x_n}{h} - E_i'U_i\Big),$$

$$f_u^i = f_u\Big(T_i, U_i, \sum_{j=1}^{s} w_{ij}\frac{E_jU_j - E(t_n)x_n}{h} - E_i'U_i\Big),$$

and $g_u^i = g_u(T_i, U_i)$, then the Jacobian matrix of Φ_n with respect to U is

(3.8)
$$\frac{\partial \Phi_n}{\partial U} =$$
$$\left[\begin{array}{c|c|c|c}
\begin{matrix} hf_u^1 + w_{11} f_v^1 E_1 - hf_v^1 E_1' \\ g_u^1 \end{matrix} & \begin{matrix} w_{12} f_v^1 E_2 \\ 0 \end{matrix} & \dots & \begin{matrix} w_{1s} f_v^1 E_s \\ 0 \end{matrix} \\ \hline
\begin{matrix} w_{21} f_v^2 E_1 \\ 0 \end{matrix} & \begin{matrix} hf_u^2 + w_{22} f_v^2 E_2 - hf_v^2 E_2' \\ g_u^2 \end{matrix} & \dots & \begin{matrix} w_{2s} f_v^2 E_s \\ 0 \end{matrix} \\ \hline
\vdots & \vdots & \ddots & \vdots \\ \hline
\begin{matrix} w_{s1} f_v^s E_1 \\ 0 \end{matrix} & \begin{matrix} w_{s2} f_v^s E_2 \\ 0 \end{matrix} & \dots & \begin{matrix} hf_u^s + w_{ss} f_v^s E_s - hf_v^s E_s' \\ g_u^s \end{matrix}
\end{array}\right].$$

We will show that $J = \frac{\partial \Phi_n}{\partial U}$ is nonsingular for sufficiently small h and for x_n in a small neighbourhood of the exact solution.

LEMMA 3.2. *Suppose that the condition (1.2) holds and $A = [a_{ij}]$ is invertible. Then the Jacobian $\frac{\partial \Phi_n}{\partial U}$ given in (3.8) is nonsingular for sufficiently small h and for x_n in a small neighborhood of the exact solution of problem (1.1).*

Proof. By assumption and the definition $W = A^{-1}$, it follows that the matrix

$$\bar{H} = \left[\begin{array}{c|c|c|c}
\begin{matrix} w_{11} f_v E \\ w_{11} g_u \end{matrix} & \begin{matrix} w_{12} f_v E \\ w_{12} g_u \end{matrix} & \dots & \begin{matrix} w_{1s} f_v E \\ w_{1s} g_u \end{matrix} \\ \hline
\begin{matrix} w_{21} f_v E \\ w_{21} g_u \end{matrix} & \begin{matrix} w_{22} f_v E \\ w_{22} g_u \end{matrix} & \dots & \begin{matrix} w_{s2} f_v E \\ w_{2s} g_u \end{matrix} \\ \hline
\vdots & \vdots & \ddots & \vdots \\ \hline
\begin{matrix} w_{s1} f_v E \\ w_{s1} g_u \end{matrix} & \begin{matrix} w_{s2} f_v E \\ w_{s2} g_u \end{matrix} & \dots & \begin{matrix} w_{ss} f_v E \\ w_{ss} g_u \end{matrix}
\end{array}\right]_{t=t_n} = W \otimes \begin{bmatrix} f_v E \\ g_u \end{bmatrix}_{t=t_n}$$

is boundedly invertible for sufficiently small h. Therefore, the matrix

$$\tilde{H} = \left[\begin{array}{c|c|c|c}
\begin{matrix} w_{11} f_v^1 E_1 \\ w_{11} g_u^1 \end{matrix} & \begin{matrix} w_{12} f_v^1 E_2 \\ w_{12} g_u^2 \end{matrix} & \dots & \begin{matrix} w_{1s} f_v^1 E_s \\ w_{1s} g_u^s \end{matrix} \\ \hline
\begin{matrix} w_{21} f_v^2 E_1 \\ w_{21} g_u^1 \end{matrix} & \begin{matrix} w_{22} f_v^2 E_2 \\ w_{22} g_u^2 \end{matrix} & \dots & \begin{matrix} w_{2s} f_v^2 E_s \\ w_{2s} g_u^s \end{matrix} \\ \hline
\vdots & \vdots & \ddots & \vdots \\ \hline
\begin{matrix} w_{s1} f_v^s E_1 \\ w_{s1} g_u^1 \end{matrix} & \begin{matrix} w_{s2} f_v^s E_2 \\ w_{s2} g_u^2 \end{matrix} & \dots & \begin{matrix} w_{ss} f_v^s E_s \\ w_{ss} g_u^s \end{matrix}
\end{array}\right] = B\tilde{J} \tag{3.9}$$

is boundedly invertible for sufficiently small h as well. Here,

$$\tilde{J} := \left[\begin{array}{c|c|c|c}
\begin{matrix} w_{11} f_v^1 E_1 \\ g_u^1 \end{matrix} & \begin{matrix} w_{12} f_v^1 E_2 \\ 0 \end{matrix} & \dots & \begin{matrix} w_{1s} f_v^1 E_s \\ 0 \end{matrix} \\ \hline
\begin{matrix} w_{21} f_v^2 E_1 \\ 0 \end{matrix} & \begin{matrix} w_{22} f_v^2 E_2 \\ g_u^2 \end{matrix} & \dots & \begin{matrix} w_{2s} f_v^2 E_s \\ 0 \end{matrix} \\ \hline
\vdots & \vdots & \ddots & \vdots \\ \hline
\begin{matrix} w_{s1} f_v^s E_1 \\ 0 \end{matrix} & \begin{matrix} w_{s2} f_v^s E_2 \\ 0 \end{matrix} & \dots & \begin{matrix} w_{ss} f_v^s E_s \\ g_u^s \end{matrix}
\end{array}\right]$$

and

$$B := \begin{bmatrix} I_{m_1} & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & w_{11}I_{m_2} & 0 & w_{12}I_{m_2} & \cdots & 0 & w_{1s}I_{m_2} \\ 0 & 0 & I_m & 0 & \cdots & 0 & 0 \\ 0 & w_{21}I_{m_2} & 0 & w_{22}I_{m_2} & \cdots & 0 & w_{2s}I_{m_2} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & I_m & 0 \\ 0 & w_{s1}I_{m_2} & 0 & w_{s2}I_{m_2} & \cdots & 0 & w_{ss}I_{m_2} \end{bmatrix},$$

where I_{m_1}, I_{m_2} are identity matrices. It is not difficult to verify that the matrix B is invertible. Hence, it follows that the matrix $\tilde{J}$ in (3.9) is boundedly invertible for sufficiently small h as well. Since $J = \tilde{J} + O(h)$, we conclude that the Jacobian matrix $J = \frac{\partial \Phi_n}{\partial U}$ in (3.8) is nonsingular for all sufficiently small h and for x_n in a small neighbourhood of the exact solution of problem (1.1). □

Once the unique solution U is numerically determined, e.g., by Newton's method, a numerical approximation of K is immediately obtained. If the given Runge-Kutta method is stiffly accurate, i.e., A is invertible and the last row of A and b^T coincide, then we simply set $x_{n+1} = U_s$. Otherwise, the approximation x_{n+1} will be determined by solving the extra system (3.6d), (3.6e), which is rewritten in the form $L_n(x_{n+1}) = 0$. The associated Jacobian of L_n is

$$\frac{\partial L_n}{\partial x_{n+1}} = \begin{bmatrix} E \\ g_u \end{bmatrix}_{t=t_{n+1}}.$$

Since f_v is invertible and

$$\begin{bmatrix} f_v E \\ g_u \end{bmatrix} = \begin{bmatrix} f_v & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} E \\ g_u \end{bmatrix}$$

is nonsingular in a small neighborhood of the solution $x(t)$, the Jacobian of L_n is boundedly invertible. Therefore, the solution x_{n+1} of (3.6d), (3.6e) exists, and it is locally unique.

When implementing the IRK method (3.6), the numerical values of U_i, $i = 1, 2, \ldots, s$, are approximated from the system (3.7) by Newton's method. However, for an easy implementation, the equation (3.7) is replaced by

$$\begin{aligned} 0 &= hf\big(T_i, U_i, \sum_{j=1}^{s} w_{ij} \frac{E_j U_j - E(t_n)x_n}{h} - E_i' U_i\big), \\ 0 &= \sum_{j=1}^{s} w_{ij} g(T_j, U_j), \qquad\qquad i = 1, 2, \ldots, s. \end{aligned}$$

Here, we recall once again that the coefficients w_{ij} are the entries of $W = A^{-1}$. Therefore, the system defining U is of the form $\bar{\Phi}_n(U) = 0$, and the associated Jacobian matrix is

(3.10)
$$\frac{\partial \bar{\Phi}_n}{\partial U} = \left[\begin{array}{c|c|c|c}
\begin{matrix} hf_u^1 + w_{11} f_v^1 E_1 - hf_v^1 E_1' \\ w_{11} g_u^1 \end{matrix} & \begin{matrix} w_{12} f_v^1 E_2 \\ w_{12} g_u^2 \end{matrix} & \dots & \begin{matrix} w_{1s} f_v^1 E_s \\ w_{1s} g_u^s \end{matrix} \\ \hline
\begin{matrix} w_{21} f_v^2 E_1 \\ w_{21} g_u^1 \end{matrix} & \begin{matrix} hf_u^2 + w_{22} f_v^2 E_2 - hf_v^2 E_2' \\ w_{22} g_u^2 \end{matrix} & \dots & \begin{matrix} w_{2s} f_v^2 E_s \\ w_{2s} g_u^s \end{matrix} \\ \hline
\vdots & \vdots & \ddots & \vdots \\ \hline
\begin{matrix} w_{s1} f_v^s E_1 \\ w_{s1} g_u^1 \end{matrix} & \begin{matrix} w_{s2} f_v^s E_2 \\ w_{s2} g_u^2 \end{matrix} & \dots & \begin{matrix} hf_u^s + w_{ss} f_v^s E_s - hf_v^s E_s' \\ w_{ss} g_u^s \end{matrix}
\end{array}\right].$$

We denote this Jacobian matrix by H. By ignoring all the terms of size $O(h)$ appearing on the right-hand side of (3.10), H can be approximated by

$$\tilde{H} = \left[\begin{array}{c|c|c|c}
\begin{matrix} w_{11} f_v^1 E_1 \\ w_{11} g_u^1 \end{matrix} & \begin{matrix} w_{12} f_v^1 E_2 \\ w_{12} g_u^2 \end{matrix} & \dots & \begin{matrix} w_{1s} f_v^1 E_s \\ w_{1s} g_u^s \end{matrix} \\ \hline
\begin{matrix} w_{21} f_v^2 E_1 \\ w_{21} g_u^1 \end{matrix} & \begin{matrix} w_{22} f_v^2 E_2 \\ w_{22} g_u^2 \end{matrix} & \dots & \begin{matrix} w_{2s} f_v^2 E_s \\ w_{2s} g_u^s \end{matrix} \\ \hline
\vdots & \vdots & \ddots & \vdots \\ \hline
\begin{matrix} w_{s1} f_v^s E_1 \\ w_{s1} g_u^1 \end{matrix} & \begin{matrix} w_{s2} f_v^s E_2 \\ w_{s2} g_u^2 \end{matrix} & \dots & \begin{matrix} w_{ss} f_v^s E_s \\ w_{ss} g_u^s \end{matrix}
\end{array}\right],$$

which is boundedly invertible for sufficiently small h as we have seen in the proof of Lemma 3.2. For simplicity, it can be further approximated by the "frozen" Jacobian

$$\bar{H} = \left[\begin{array}{c|c|c|c}
\begin{matrix} w_{11} f_v E \\ w_{11} g_u \end{matrix} & \begin{matrix} w_{12} f_v E \\ w_{12} g_u \end{matrix} & \dots & \begin{matrix} w_{1s} f_v E \\ w_{1s} g_u \end{matrix} \\ \hline
\begin{matrix} w_{21} f_v E \\ w_{21} g_u \end{matrix} & \begin{matrix} w_{22} f_v E \\ w_{22} g_u \end{matrix} & \dots & \begin{matrix} w_{2s} f_v E \\ w_{2s} g_u \end{matrix} \\ \hline
\vdots & \vdots & \ddots & \vdots \\ \hline
\begin{matrix} w_{s1} f_v E \\ w_{s1} g_u^1 \end{matrix} & \begin{matrix} w_{s2} f_v E \\ w_{s2} g_u \end{matrix} & \dots & \begin{matrix} w_{ss} f_v E \\ w_{ss} g_u \end{matrix}
\end{array}\right]_{t_n} = W \otimes \begin{bmatrix} f_v E \\ g_u \end{bmatrix}_{t=t_n}. \tag{3.11}$$

It is easy to calculate the inverse of $\bar{H}$, namely

$$\bar{H}^{-1} = A \otimes \begin{bmatrix} f_v E \\ g_u \end{bmatrix}_{t=t_n}^{-1}.$$

Thus, at each time step, only one LU factorization of a matrix of size n by n is needed.

REMARK 3.3. In order to determine the approximation x_{n+1} at the time step $t = t_{n+1}$, we have to evaluate the derivatives $E'(T_i)$, $i = 1, 2, \dots, s$. If the derivative of E is not available analytically, then it can be approximated by appropriate finite difference formulas or by using interpolation polynomials based on a set of nearby stage points. It is recommendable that the order of the finite difference formulas should not be less than the order of the Runge-Kutta method that we use; see the result on the analysis of the computational errors in Theorem 3.8 below.

REMARK 3.4. If the matrix A of the IRK method is lower triangular with nonzero diagonal elements, i.e., we deal with a diagonally implicit Runge-Kutta method (DIRK), then the implementation of the IRK scheme (3.6) and that of the HERK scheme (3.1) are almost the same. Indeed, if we take a DIRK method with

$$A = \begin{bmatrix} a_{11} & 0 & 0 & \cdots & 0 & 0 \\ a_{21} & a_{22} & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{s-1,1} & a_{s-1,2} & a_{s-1,3} & \cdots & a_{s-1,s-1} & 0 \\ a_{s,1} & a_{s,2} & a_{s,3} & \cdots & a_{s,s-1} & a_{s,s} \end{bmatrix},$$

then the system (3.6) becomes

$$E_i U_i = E(t_n)x_n + h\sum_{j=1}^{i} a_{ij}K_j, \tag{3.12a}$$

$$0 = f\big(T_i, U_i, K_i - E_i' U_i\big), \tag{3.12b}$$

$$0 = g(T_i, U_i), \qquad i = 1, 2, \ldots, s, \tag{3.12c}$$

$$E(t_{n+1})x_{n+1} = E(t_n)x_n + h\sum_{i=1}^{s} b_i K_i, \tag{3.12d}$$

$$0 = g(t_{n+1}, x_{n+1}). \tag{3.12e}$$

From the equation (3.12a), we find the expression for K_i, and by substituting the result into (3.12b), we obtain a nonlinear system for U_i. Thus, we solve subsequently s nonlinear systems for U_i, $i = 1, 2, \ldots, s$. This procedure is similar to the implementation of the HERK scheme (3.1). Finally, we solve the system (3.12d), (3.12e) for x_{n+1}.

3.3. Convergence analysis.

We now analyze the convergence of the HERK and the IRK methods applied to the reformulated form (2.5). To obtain the convergence results for the discretization schemes presented above, we begin with the following lemma.

LEMMA 3.5. *The reduction of the form* (2.5) *to the form* (2.8) *and the discretization by the HERK/IRK method commute, i.e., the following diagram is commutative:*

$$\begin{tikzcd} \textit{DAE (2.5)} \arrow[r, "x(t) = Q(t)y(t)"] \arrow[d, "\textit{HERK/IRK}"'] & \textit{DAE (2.8)} \arrow[d, "\textit{HERK/IRK}"] \\ \textit{Scheme (3.1)/(3.6)} \arrow[r, "x_n = Q(t_n)y_n"] & \textit{Scheme (2.3)} \end{tikzcd}$$

Proof. Consider the DAE (2.8) on an interval $[t_n, t_{n+1}]$. We assume that $y_{1,n}, y_{2,n}$ are the approximations of $y_1(t_n), y_2(t_n)$, respectively. Let the stage approximations be defined by $[V_i^T \; H_i^T]^T \simeq y(T_i) = y(t_n + c_i h) = [y_1^T(T_i) \; y_2^T(T_i)]^T$. The Runge-Kutta scheme (2.3) applied to the semi-explicit index-1 DAE (2.8) reads

$$\begin{aligned}
V_i &= y_{1,n} + h\sum_{j=1}^{s} a_{ij} F\big(T_j, V_j, H_j\big), \\
0 &= G(T_i, V_i, H_i), \qquad\qquad i = 1, 2, \dots, s, \\
y_{1,n+1} &= y_{1,n} + h\sum_{i=1}^{s} b_i F\big(T_i, V_i, H_i\big), \\
0 &= G(t_{n+1}, y_{1,n+1}, y_{2,n+1}).
\end{aligned} \tag{3.13}$$

Let $P_i = F\big(T_i, V_i, H_i\big)$ be approximations to the derivatives $y_1'(T_i)$, $i = 1, 2, \dots, s$. By the definition of F and G given when introducing (2.8), we have the equivalent system

$$\begin{aligned}
V_i &= y_{1,n} + h\sum_{j=1}^{s} a_{i,j} P_j, \\
0 &= f\big(T_i, Q_i[V_i^T\ H_i^T]^T, P_i - E_i' Q_i[V_i^T\ H_i^T]^T\big), \\
0 &= g(T_i, Q_i[V_i^T\ H_i^T]^T), \qquad\qquad i = 1, 2, \dots, s, \\
y_{1,n+1} &= y_{1,n} + h\sum_{i=1}^{s} b_i P_i, \\
0 &= g(t_{n+1}, Q(t_{n+1}) y_{n+1}).
\end{aligned} \tag{3.14}$$

Here we have set $Q_i = Q(T_i)$. On the other hand, we now show that the RK methods (3.1) and (3.6) for (2.5) lead to the scheme (3.14) by the corresponding change of variables $x_n = Q(t_n) y_n$. Let us define $[M_i^T\ N_i^T]^T = Q_i^{-1} U_i$. Here the partition is done according to the dimensions of the variables y_1 and y_2. By the definition of the matrix Q in (2.6), we have

$$E_i U_i = E_i Q_i (Q_i)^{-1} U_i = [I\ \ 0](Q_i)^{-1} U_i = [I\ \ 0][M_i^T\ N_i^T]^T = M_i.$$

Similarly, we have

$$E(t_n) x_n = E(t_n) Q(t_n) (Q(t_n))^{-1} x_n = [I\ \ 0][y_{1,n}^T\, y_{2,n}^T]^T = y_{1,n}.$$

Then, the RK schemes (3.1) and (3.6) can be rewritten as

$$\begin{aligned}
M_i &= y_{1,n} + h\sum_{j=1}^{s} a_{i,j} K_j, \\
0 &= f\big(T_i, Q_i[M_i^T\ N_i^T]^T, K_i - E_i' Q_i[M_i^T\ N_i^T]^T\big), \\
0 &= g\big(T_i, Q_i[M_i^T\ N_i^T]^T\big), \qquad\qquad i = 1, 2, \dots, s, \\
y_{1,n+1} &= y_{1,n} + h\sum_{i=1}^{s} b_i K_i, \\
0 &= g\big(t_{n+1}, Q(t_{n+1}) y_{n+1}\big).
\end{aligned} \tag{3.15}$$

Clearly, the scheme (3.15) and the scheme (3.14) coincide. □

The convergence of the RK scheme (3.6) immediately follows.

THEOREM 3.6. *Consider the IVP for the DAE* (1.1) *with consistent initial value, i.e.,* $g(t_0, x_0) = 0$*. Suppose that* (1.2) *holds in a neighbourhood of the exact solution* $x(t)$*. Then, the IRK scheme* (3.6) *applied to the equivalent DAE* (2.5) *is convergent of order* p*, i.e.,*

$$\|x_n - x(t_n)\| = \mathcal{O}(h^p) \qquad \text{as } h \to 0$$

($t_n \in [t_0, T]$ *is fixed with* $t_n - t_0 = nh$*).*

Proof. According to Lemma 3.5, the scheme (3.6) applied to the DAE (1.1) leads to the scheme (3.13) for the problem (2.8). Namely, the relation $x_n = Q(t_n)y_n$ holds. By Theorem 2.1, we obtain

$$\|y_n - y(t_n)\| = \mathcal{O}(h^p).$$

It follows that

$$\|x_n - x(t_n)\| = \|Q(t_n)y_n - Q(t_n)y(t_n)\| \le \|Q(t_n)\|\|y_n - y(t_n)\| = \mathcal{O}(h^p). \qquad \square$$

Similarly, we obtain the convergence of the half-explicit Runge-Kutta scheme (3.1).

THEOREM 3.7. *Consider the IVP for the DAE* (1.1) *with consistent initial value, i.e.,* $g(t_0, x_0) = 0$*. Suppose that* (1.2) *holds in a neighbourhood of the exact solution* $x(t)$*. Then, the HERK scheme* (3.1) *applied to the equivalent DAE* (2.5) *is convergent of order* p*, i.e.,*

$$\|x_n - x(t_n)\| = \mathcal{O}(h^p) \qquad \text{as } h \to 0$$

($t_n \in [t_0, T]$ *is fixed with* $t_n - t_0 = nh$*).*

3.4. Absolute stability.

In addition to the convergence analysis, we are also interested in the absolute stability of the numerical methods. For ODEs, the well-known test equation $y' = \lambda y$, where $\Re\lambda \le 0$, is used; see, e.g., [2, 9]. Here we analyze the absolute stability of the RK schemes (3.1) and (3.6) via the following test equation for DAEs; see [14]. Consider the linear DAE

$$\begin{bmatrix} 1 & -\omega t \\ 0 & 0 \end{bmatrix} x' = \begin{bmatrix} \lambda & \omega(1-\lambda t) \\ -1 & (1+\omega t) \end{bmatrix} x, \tag{3.16}$$

where ω and λ are complex parameters, $\Re\lambda \le 0$, and $x = [x_1, x_2]^T$. The system (3.16) is a strangeness-free DAEs of the form (1.1), where

$$E(t) = \begin{bmatrix} 1 & -\omega t \end{bmatrix}.$$

Given initial data $x_1(0) = 1, x_2(0) = 1$, then the system (3.16) has the solution

$$x = \begin{bmatrix} e^{\lambda t}(1+\omega t) \\ e^{\lambda t} \end{bmatrix}.$$

In [14], the concept of Dahlquist's stability function is extended to DAEs by considering the stability function $R(z, w)$ defined for the test DAE (3.16). If we apply the half-explicit Euler method which was proposed in [17] to the test DAE (3.16), then we obtain the DAE stability function

$$R(z, w) = \frac{1 + z + w}{1 + w},$$

where $z = \lambda h, w = \omega h$, and h is the stepsize. For the implicit Euler method (see [14]), we have the stability function

$$R(z, w) = \frac{1-w}{1-z-w}.$$

Next, we determine the DAE stability function for the half-explicit and implicit Runge-Kutta methods presented in this section. The system (3.16) is reformulated as

$$\left(\begin{bmatrix} 1 & -\omega t \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \right)' = \begin{bmatrix} \lambda & -\lambda\omega t \\ -1 & (1+\omega t) \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}. \tag{3.17}$$

Applying the half-explicit Euler method, after some elementary manipulations, we obtain

$$\begin{aligned} x_{2,n+1} &= (1+\lambda h) x_{2,n}, \\ x_{1,n+1} &= (1+\lambda h)(1+\omega t_{n+1}) x_{2,n+1}. \end{aligned}$$

Therefore, we obtain the DAE stability function $R(z, w) = 1 + z$. In a similar way, if we apply the implicit Euler method to the reformulated test equation (3.17), then we obtain the DAE stability function

$$R(z, w) = \frac{1}{1-z}.$$

Note that these stability functions are exactly the stability functions of the Euler methods for the ODE case. We now determine the DAE stability function $R(z, w)$ in the general cases. Applying the scheme (3.1) or (3.6) to the problem (3.16) yields

$$M_i = y_{1,n} + h \sum_{j=1}^{s} a_{ij} K_j, \tag{3.18a}$$

$$K_i = \lambda(U_{1,i} - \omega T_i U_{2,i}), \tag{3.18b}$$

$$0 = -U_{1,i} + U_{2,i} + \omega T_i U_{2,i}, \quad i = 1, 2, \ldots, s, \tag{3.18c}$$

$$y_{1,n+1} = y_{1,n} + h \sum_{i=1}^{s} b_i K_i, \tag{3.18d}$$

$$0 = -x_{1,n+1} + x_{2,n+1} + \omega t_{n+1} x_{2,n+1}, \tag{3.18e}$$

where $y_{1,n} = x_{1,n} - \omega t_n x_{2,n}$, $M_i = U_{1,i} - \omega T_i U_{2,i}$, $i = 1, 2, \ldots, s$. Let us set

$$M = [M_1^T \; M_2^T \; \ldots \; M_s^T]^T, \quad K = [K_1^T \; K_2^T \; \ldots \; K_s^T]^T, \quad \text{and} \quad \mathbf{1} = [1 \; 1 \; \ldots \; 1]^T.$$

Equation (3.18b) leads to $K_i = \lambda M_i$, $i = 1, 2, \ldots, s$, hence it follows that

$$K = \lambda M. \tag{3.19}$$

Moreover, equation (3.18a) implies $M = \mathbf{1} y_{1,n} + hAK$. Replacing K by (3.19), it is easily seen that

$$M = (I - h\lambda A)^{-1} \mathbf{1} y_{1,n}. \tag{3.20}$$

From the system (3.18d)–(3.18e), we obtain

$$\begin{aligned} y_{1,n+1} &= y_{1,n} + hb^T K, \\ x_{2,n+1} &= x_{1,n+1} - \omega t_{n+1} x_{2,n+1} = y_{1,n+1}, \\ x_{1,n+1} &= (1 + \omega t_{n+1}) x_{2,n+1}. \end{aligned} \tag{3.21}$$

Substituting (3.19), (3.20) into the first equation of (3.21) and taking $z = h\lambda$, we have

$$y_{1,n+1} = \Big(1 + zb^T(I - zA)^{-1}\mathbf{1}\Big)y_{1,n}.$$

The second equation of (3.21) yields $y_{1,n} = x_{2,n}$. Therefore, we obtain

$$\begin{aligned} x_{2,n+1} &= \Big(1 + zb^T(I - zA)^{-1}\mathbf{1}\Big)x_{2,n}, \\ x_{1,n+1} &= (1 + \omega t_{n+1})x_{2,n+1}. \end{aligned} \tag{3.22}$$

From equation (3.22) and the definition of the stability function for DAEs (see [14]), we obtain the stability function for both the scheme (3.1) and (3.6)

$$R(z, w) = 1 + zb^T(I - zA)^{-1}\mathbf{1}.$$

We conclude that the methods (3.1) and (3.6) applied to the reformulated test DAE (3.17) preserve the stability property of the original (explicit or implicit) Runge-Kutta methods. This fact will be particularly important when we approximate Lyapunov and Sacker-Sell spectral intervals numerically; see [16, 18].

3.5. The influence of computational errors. When we implement the methods (3.1) and (3.6) for the reformulated DAEs (2.5), certain computational errors arise, namely rounding errors, errors caused by Newton's method for solving nonlinear systems, and approximation errors for the evaluation of E'. The accumulation of these errors will be discussed in this section. For the sake of simplicity, first we consider the half-explicit Euler method and present a rigorous perturbation analysis. Also for simplifying the notations in this part, we set

$$E_n = E(t_n), \quad E_{n+1} = E(t_{n+1}), \quad E'_n = E'(t_n), \qquad \text{for } n = 0, 1, \ldots, N-1.$$

Furthermore, $\widetilde{E}'_n$ is an approximation to $E'(t_n)$.

Theorem 3.8. *Suppose that $x_0, \widetilde{x}_0$ are the exact and perturbed initial values, respectively. Let $\{x_n\}$ be the solution of the half-explicit Euler scheme for the DAEs (2.5) and $\{\widetilde{x}_n\}$ be the perturbed solution defined by the following perturbed scheme*

$$\begin{aligned} \delta_n &= hf\Big(t_n, \widetilde{x}_n, \frac{E_{n+1}\widetilde{x}_{n+1} - E_n\widetilde{x}_n}{h} - \widetilde{E}'_n\widetilde{x}_n\Big), \\ \theta_n &= g(t_{n+1}, \widetilde{x}_{n+1}), \end{aligned} \tag{3.23}$$

where $h = t_{n+1} - t_n$. Let us denote $\theta_{-1} = g(t_0, \widetilde{x}_0)$. We assume that the errors δ_n, θ_n, and $\varepsilon_n = \widetilde{E'}_n - E'_n$ are sufficiently small for $n = 0, 1, \ldots, N-1$. Then there exist constants $\mathcal{C}$, $\mathcal{K}$, $\mathcal{L}$, $\mathcal{M}$, and h_0 such that for any mesh with $h \le h_0$, the perturbed solution $\{\widetilde{x}_n\}$ exists and satisfies

$$\begin{aligned} \|\widetilde{x}_n - x_n\| &\le \mathcal{C}\|\widetilde{x}_0 - x_0\| \\ &\qquad + \mathcal{K}\max_{0\le i\le N-1}\|\varepsilon_i\| + \mathcal{L}\max_{0\le i\le N-1}\|\delta_i/h\| + \mathcal{M}\max_{-1\le i\le N-1}\|\theta_i\| \end{aligned} \tag{3.24}$$

for all $n \ge 0$, provided that the initial error $\widetilde{x}_0 - x_0$ is sufficiently small.

Proof. First, by the same argument used for verifying the feasibility of the HERK methods (3.1), the perturbed system (3.23) has a locally unique solution $\widetilde{x}_{n+1}$, provided that $\widetilde{x}_n$ is sufficiently close to the exact value $x(t_n)$ and h is sufficiently small. By induction, we will first prove the estimate (3.24), then the global existence of the sequence $\{\widetilde{x}_n\}$ follows.

With the matrix Q defined by (2.6), we denote $Q_n = Q(t_n)$ and introduce the transformation

$$x_n = Q_n y_n = Q_n(y_{1,n}^T, y_{2,n}^T)^T, \qquad \widetilde{x}_n = Q_n \widetilde{y}_n = Q_n(\widetilde{y}_{1,n}^T, \widetilde{y}_{2,n}^T)^T,$$

which yields the transformed system

$$\begin{aligned} 0 &= f\Big(t_n, Q_n\widetilde{y}_n, \frac{\widetilde{y}_{1,n+1} - \widetilde{y}_{1,n}}{h} - \widetilde{E}_n' Q_n \widetilde{y}_n\Big) - \frac{\delta_n}{h}, \\ 0 &= g(t_{n+1}, Q_{n+1}\widetilde{y}_{n+1}) - \theta_n. \end{aligned}$$

Due to the invertibility of f_v and invoking the Implicit Function Theorem, there exists a function $\widetilde{F} = \widetilde{F}(t, y_1, y_2, \delta)$ such that the system (3.23) is rewritten as

$$\begin{aligned} \frac{\widetilde{y}_{1,n+1} - \widetilde{y}_{1,n}}{h} - \widetilde{E}_n' Q_n \widetilde{y}_n &= \widetilde{F}\big(t_n, \widetilde{y}_{1,n}, \widetilde{y}_{2,n}, \delta_n/h\big), \\ 0 &= G(t_{n+1}, \widetilde{y}_{1,n+1}, \widetilde{y}_{2,n+1}, \theta_n). \end{aligned}$$

Equivalently, we have

$$\begin{aligned} \widetilde{y}_{1,n+1} &= \widetilde{y}_{1,n} + h\widetilde{E}_n' Q_n \widetilde{y}_n + h\widetilde{F}\big(t_n, \widetilde{y}_{1,n}, \widetilde{y}_{2,n}, \delta_n/h\big), \\ 0 &= G(t_{n+1}, \widetilde{y}_{1,n+1}, \widetilde{y}_{2,n+1}, \theta_n). \end{aligned} \tag{3.25}$$

Here, the Jacobian matrix of $G = G(t, y_1, y_2, \theta)$ with respect to y_2 is $\frac{\partial G}{\partial y_2} = g_u Q^{(2)}$, which is nonsingular in a neighborhood of $\big(t, y_1(t), y_2(t), 0\big)$. According to the Implicit Function Theorem, there exists a function $\chi = \chi(t, y_1, \theta)$ such that

$$\widetilde{y}_{2,n+1} = \chi(t_{n+1}, \widetilde{y}_{1,n+1}, \theta_n).$$

According to the half-explicit Euler method applied to (2.5), x_{n+1} is determined from the unperturbed system

$$\begin{aligned} 0 &= hf\Big(t_n, x_n, \frac{E_{n+1}x_{n+1} - E_n x_n}{h} - E_n' x_n\Big), \\ 0 &= g(t_{n+1}, x_{n+1}). \end{aligned}$$

In a similar way, we derive

$$\begin{aligned} y_{1,n+1} &= y_{1,n} + hE_n' Q_n y_n + h\widetilde{F}\big(t_n, y_{1,n}, y_{2,n}, 0\big), \\ 0 &= G(t_{n+1}, y_{1,n+1}, y_{2,n+1}, 0). \end{aligned} \tag{3.26}$$

The second equation of (3.26) yields

$$y_{2,n+1} = \chi(t_{n+1}, y_{1,n+1}, 0).$$

The main idea is as follows. By substituting the expressions of $\widetilde{y}_{2,n+1}$ and $y_{2,n+1}$ into the first equations, the error estimation problem is equivalent to the stability estimation problem of the standard Euler scheme, which is well-known. Thus, we estimate the difference $\|\widetilde{y}_{1,n+1} - y_{1,n+1}\|$ first, then a bound for $\|\widetilde{y}_{n+1} - y_{n+1}\|$ is obtained. Finally, the estimate for $\|\widetilde{x}_{n+1} - x_{n+1}\|$ follows.

Let us denote $c_n = \widetilde{y}_{1,n} - y_{1,n}$, $d_n = \widetilde{y}_{2,n} - y_{2,n}$. Differentiating the equation $G(t, y_1, \chi(t, y_1, \theta), \theta) = 0$, we obtain $G_{y_1} + G_{y_2}\chi_{y_1} = 0$. Since G_{y_2} is nonsingular in a sufficiently small neighborhood of the exact solution, we have $\chi_{y_1} = -G_{y_2}^{-1}G_{y_1}$. By a similar argument, we obtain $\chi_\theta = -G_{y_2}^{-1}G_\theta$. From the definition of G, we have $G_\theta = -I$. Thus, the equality $\chi_\theta = G_{y_2}^{-1}$ holds. Hence, there exist a constant $\mathcal{C}_1 > 0$ and a sufficiently small h_0 so that we have

$$\|d_{n+1}\| \le \mathcal{C}_1(\|c_{n+1}\| + \|\theta_n\|) \tag{3.27}$$

for all $h \le h_0$. From the first equations of (3.25) and (3.26), we have

$$\begin{aligned}\|c_{n+1}\| \le \|c_n\| + h\|(\widetilde{E}_n' - E_n')Q_n\widetilde{y}_n\| \\ + h\|E_n'Q_n(\widetilde{y}_n - y_n)\| + h\|\widetilde{F}\big(t_n, \widetilde{y}_{1,n}, \widetilde{y}_{2,n}, \delta_n/h\big) - \widetilde{F}\big(t_n, y_{1,n}, y_{2,n}, 0\big)\|.\end{aligned}$$

There exist positive constants $\mathcal{K}_1$ and $\mathcal{C}_2 > 0$ such that $\|(\widetilde{E}_n' - E_n')Q_n\widetilde{y}_n\| \le \mathcal{K}_1\|\varepsilon_n\|$ and $\|E_n'Q_n(\widetilde{y}_n - y_n)\| \le \|E_n'\|\|Q_n(\widetilde{y}_n - y_n)\| \le \mathcal{C}_2(\|c_n\| + \|d_n\|)$. By similar arguments, there exist positive constants $\mathcal{C}_3$, $\mathcal{C}_4$, $\mathcal{L}_1$ such that

$$\begin{aligned}\|\widetilde{F}\big(t_n, \widetilde{y}_{1,n}, \widetilde{y}_{2,n}, \widetilde{\delta}_n\big) - \widetilde{F}\big(t_n, y_{1,n}, y_{2,n}, 0\big)\| &\le \mathcal{C}_3\|c_n\| + \mathcal{C}_4\|d_n\| + \mathcal{L}_1\|\delta_n/h\| \\ &\le \mathcal{C}_5\|c_n\| + \mathcal{M}_1\|\theta_{n-1}\| + \mathcal{L}_1\|\delta_n/h\|,\end{aligned}$$

where $\mathcal{C}_5 = \mathcal{C}_3 + \mathcal{C}_4\mathcal{C}_1$, $\mathcal{M}_1 = \mathcal{C}_4\mathcal{C}_1$. Then, we have

$$\begin{aligned}\|c_{n+1}\| &\le \|c_n\| + h\mathcal{K}_1\|\varepsilon_n\| + h\mathcal{C}_2(\|c_n\| + \|d_n\|) \\ &\qquad + h\mathcal{C}_5\|c_n\| + h\mathcal{M}_1\|\theta_{n-1}\| + h\mathcal{L}_1\|\delta_n/h\| \\ &\le \|c_n\| + h\mathcal{K}_1\|\varepsilon_n\| + h\mathcal{C}_2\big(\|c_n\| + \mathcal{C}_1(\|c_n\| + \|\theta_{n-1}\|)\big) \\ &\qquad + h\mathcal{C}_5\|c_n\| + h\mathcal{M}_1\|\theta_{n-1}\| + h\mathcal{L}_1\|\delta_n/h\| \\ &= (1 + h\widetilde{\mathcal{C}}_0)\|c_n\| + h\big(\mathcal{K}_1\|\varepsilon_n\| + \mathcal{M}_2\|\theta_{n-1}\| + \mathcal{L}_1\|\delta_n/h\|\big)\end{aligned} \tag{3.28}$$

for all $n \ge 0$ and all $h \le h_0$, where $\widetilde{\mathcal{C}}_0 = \mathcal{C}_2(1 + \mathcal{C}_1) + \mathcal{C}_5$, $\mathcal{M}_2 = \mathcal{C}_2\mathcal{C}_1 + \mathcal{M}_1$. Set

$$\eta = \mathcal{K}_1 \max_{0 \le i \le N-1} \|\varepsilon_i\| + \mathcal{M}_2 \max_{-1 \le i \le N-1} \|\theta_i\| + \mathcal{L}_1 \max_{0 \le i \le N-1} \|\delta_i/h\|,$$

where we recall that $\theta_{-1} = g(t_0, \widetilde{x}_0)$. Repeating the estimation (3.28) yields

$$\begin{aligned}\|c_{n+1}\| &\le (1 + h\widetilde{\mathcal{C}}_0)\|c_n\| + h\eta \\ &\le h\|\eta\| + h(1 + h\widetilde{\mathcal{C}}_0)\eta + (1 + h\widetilde{\mathcal{C}}_0)(1 + h\widetilde{\mathcal{C}}_0)\|c_{n-1}\| \\ &\vdots \\ &\le \frac{1}{\widetilde{\mathcal{C}}_0}(e^{\widetilde{\mathcal{C}}_0(t_{n+1}-t_0)} - 1)\eta + e^{\widetilde{\mathcal{C}}_0(t_{n+1}-t_0)}\|c_0\|.\end{aligned} \tag{3.29}$$

The last inequality in (3.29) is obtained from elementary ones that are also used in proving the zero-stability of the Euler method; see [2, Chapter 3]. Let

$$\widetilde{\mathcal{C}}_1 = \max\{\frac{1}{\widetilde{\mathcal{C}}_0}(e^{\widetilde{\mathcal{C}}_0(t_N-t_0)} - 1), e^{\widetilde{\mathcal{C}}_0(t_N-t_0)}\}.$$

Then, we have

$$\|\widetilde{y}_{1,n+1} - y_{1,n+1}\| \le \widetilde{\mathcal{C}}_1\big(\|\widetilde{y}_{1,0} - y_{1,0}\| + \eta\big). \tag{3.30}$$

Combining (3.30) and (3.27) yields

$$\begin{aligned}\|\widetilde{y}_{2,n+1} - y_{2,n+1}\| &\le \mathcal{C}_1\big(\widetilde{\mathcal{C}}_1(\|\widetilde{y}_{1,0} - y_{1,0}\| + \eta) + \|\theta_n\|\big)\\ &\le \widetilde{\mathcal{C}}_2(\|\widetilde{y}_{1,0} - y_{1,0}\| + \eta),\end{aligned}$$

where $\widetilde{\mathcal{C}}_2 = \mathcal{C}_1(\widetilde{\mathcal{C}}_1 + 1/\mathcal{M}_2)$. We now derive

$$\begin{aligned}\|\widetilde{x}_{n+1} - x_{n+1}\| = \|Q_{n+1}(\widetilde{y}_{n+1} - y_{n+1})\| \le \|Q_{n+1}&\begin{pmatrix}\widetilde{y}_{1,n+1} - y_{1,n+1}\\ \widetilde{y}_{2,n+1} - y_{2,n+1}\end{pmatrix}\|\\ &\le \mu_1(\|\widetilde{y}_{1,n+1} - y_{1,n+1}\| + \|\widetilde{y}_{2,n+1} - y_{2,n+1}\|)\\ &\le \mu_1\widetilde{\mathcal{C}}_1(\|\widetilde{y}_{1,0} - y_{1,0}\| + \eta) + \mu_1\widetilde{\mathcal{C}}_2(\|\widetilde{y}_{1,0} - y_{1,0}\| + \eta)\\ &\le \mu_1(\widetilde{\mathcal{C}}_1 + \widetilde{\mathcal{C}}_2)(\|\widetilde{y}_{1,0} - y_{1,0}\| + \eta)\\ &\le \mu_1(\widetilde{\mathcal{C}}_1 + \widetilde{\mathcal{C}}_2)\big(\|Q_0^{-1}(\widetilde{x}_0 - x_0)\| + \eta\big)\\ &\le \mu_1\mu_2(\widetilde{\mathcal{C}}_1 + \widetilde{\mathcal{C}}_2)\|\widetilde{x}_0 - x_0\| + \mu_1(\widetilde{\mathcal{C}}_1 + \widetilde{\mathcal{C}}_2)\eta\\ &\le \mathcal{C}\|\widetilde{x}_0 - x_0\|\\ &\qquad + \mathcal{K}\max_{0\le i\le N-1}\|\varepsilon_i\| + \mathcal{L}\max_{0\le i\le N-1}\|\delta_i/h\| + \mathcal{M}\max_{-1\le i\le N-1}\|\theta_i\|,\end{aligned}$$

where

$$\mathcal{C} = \mu_1\mu_2(\widetilde{\mathcal{C}}_1 + \widetilde{\mathcal{C}}_2),\ \ \mathcal{L} = \mu_1(\widetilde{\mathcal{C}}_1 + \widetilde{\mathcal{C}}_2)\mathcal{L}_1,\ \ \mathcal{M} = \mu_1(\widetilde{\mathcal{C}}_1 + \widetilde{\mathcal{C}}_2)\mathcal{M}_2,\ \ \mathcal{K} = \mu_1(\widetilde{\mathcal{C}}_1 + \widetilde{\mathcal{C}}_2)\mathcal{K}_1.$$

Thus, if h_0 is sufficiently small, then by induction, the sequence $\{\widetilde{x}_n\}_{n=0}^N$ exists for $h \le h_0$, $n = 0, 1, \ldots, N-1$, and the estimate (3.24) holds with the constants $\mathcal{C}$, $\mathcal{K}$, $\mathcal{L}$, and $\mathcal{M}$, which are independent of h. □

REMARK 3.9. The estimate (3.24) also gives us some suggestions for the practical implementation. Actually, δ_n and θ_n come from two sources: from rounding errors of magnitude $\mathcal{O}(\epsilon)$, where ϵ is the machine error, and from approximation errors caused by Newton's method with a given stopping criterion. Due to the term δ_n/h on the right-hand side of (3.24), on one hand, for moderate h, one can solve the nonlinear equations approximately without harming the convergence order. It is easy to see that the tolerance of Newton's method should be prescribed at least as small as $\mathcal{O}(h^2)$. On the other hand, if the step size h becomes very small, then the rounding errors will accumulate and be dominant. If the stepsize is too small, then the rounding errors may make the actual error blow up (and even may make the existence of the actual numerical solution questionable). Similar implications can be stated for the error ε_n as well. Finally, we note that the result of Theorem 3.8 actually generalizes the well-known error analysis of the Euler method for ODEs.

REMARK 3.10. If in an implementation one avoids the scaling by h, then the first equation of the perturbed system (3.23) is replaced by

$$\delta_n = f\Big(t_n, \widetilde{x}_n, \frac{E_{n+1}\widetilde{x}_{n+1} - E_n\widetilde{x}_n}{h} - \widetilde{E}_n'\widetilde{x}_n\Big),$$

which obviously leads to another error bound that looks slightly different than (3.24). Namely, the term δ_i/h is simply replaced by δ_i. However, due to Remark 3.1, for an efficient implementation of the Runge-Kutta methods (3.1) and (3.6), the scaling by h applied to the discretization of the differential part of DAE (1.1) is useful and highly recommended.

The arguments in this proof can be used in a similar way for the analysis of computational errors for the general HERK and IRK schemes (3.1), (3.6). However, in the scope of this paper it is omitted since we wish to avoid lengthy but rather technical estimations.

TABLE 4.1
Errors of the solutions to the IVP (3.16) with $\omega = 100, \lambda = -1$.

$\alpha = 0.5$	Standard HERK2 method		HERK2 method (3.1)	
$h = 0.1$	Actual error in x_1	Error order in x_1	Actual error in x_1	Error order in x_1
h	1.0930e+002	–	9.7922e-002	–
$h/2$	5.3486e+001	1.0311	2.3546e-002	2.0562
$h/4$	2.3607e+001	1.1799	5.7751e-003	2.0276
$h/8$	9.2599e+000	1.3502	1.4302e-003	2.0137
$h/16$	3.2091e+000	1.5288	3.5587e-004	2.0068
$h/32$	9.9293e-001	1.6924	8.8758e-005	2.0034
$h = 0.1$	Actual error in x_2	Error order in x_2	Actual error in x_2	Error order in x_2
h	4.0881e-001	–	6.6154e-004	–
$h/2$	2.5432e-001	0.68479	1.5918e-004	2.0552
$h/4$	1.3317e-001	0.93334	3.9049e-005	2.0273
$h/8$	5.7892e-002	1.2018	9.6706e-006	2.0136
$h/16$	2.1089e-002	1.4568	2.4063e-006	2.0068
$h/32$	6.6539e-003	1.6643	6.0017e-007	2.0034

4. Numerical experiments. As an illustration of the schemes (3.1) and (3.6) for the DAEs (1.1), we present some numerical experiments to demonstrate the convergence order and also make a comparison with the corresponding standard schemes.

EXAMPLE 4.1. First, we consider the IVP for the test DAE (3.16) with some specified values of the parameters λ and ω on the interval $[0, 5]$.

We have solved this initial value problem by the 2-stage half-explicit Runge-Kutta methods (HERK2) on uniform meshes with different stepsizes h. The underlying explicit 2-stage RK methods are given by the following Butcher tableau:

$$\begin{array}{c|cc} 0 & 0 & 0 \\ \alpha & \alpha & 0 \\ \hline & 1-\frac{1}{2\alpha} & \frac{1}{2\alpha} \end{array} \qquad (0 < \alpha \leq 1).$$

This class of methods is well-known to be of second order for strangeness-free DAEs; see [17]. The methods are implemented in Matlab, and we compute the actual errors $\max_{0\leq n\leq N} |x_i(t_n) - x_{i,n}|$, $i = 1, 2$, for various stepsizes. The values of the parameters λ, ω, and α are specified in the tables. We also calculate estimates for the numerical convergence order. The numerical results in Tables 4.1, 4.2, and 4.3 for the components x_1 and x_2 confirm that the HERK2 method (3.1) is convergent of second order, but the numerical solutions by the standard HERK method proposed in [17] are unstable when $w = \omega h$ is not sufficiently small.

EXAMPLE 4.2. We consider the nonlinear DAE

$$\begin{aligned} x_1(x_1' + tx_2') &= x_1x_2e^t + e^{2t} + t\cos te^t - e^{2t}\sin t, \\ 0 &= e^{-t}x_1 - x_2 + \sin t - 1, \end{aligned} \tag{4.1}$$

for $t \in [0, 1]$ with the initial condition $x(0) = [1\ 0]^T$. It is easy to see that the DAE (4.1) is strangeness-free and that the exact unique solution is $x_1 = e^t, x_2 = \sin t$.

First, we carry out numerical experiments for the half-explicit variants of the classical 4-stage Runge-Kutta method (HERK4) on uniform meshes with different stepsizes h. The

TABLE 4.2
Errors of the solutions to the IVP (3.16) with $\omega = 100, \lambda = -1$.

$\alpha = 1$	Standard HERK2 method		HERK2 method (3.1)	
h = 0.1	Actual error in x_1	Error order in x_1	Actual error in x_1	Error order in x_1
h	1.1439e+000	–	2.3546e-002	–
$h/2$	4.8792e-001	1.2293	5.7751e-003	2.0276
$h/4$	1.8920e-001	1.3667	1.4302e-003	2.0137
$h/8$	6.5367e-002	1.5333	3.5587e-004	2.0068
$h/16$	2.0207e-002	1.6937	8.8758e-005	2.0034
$h/32$	5.7300e-003	1.8183	2.2163e-005	2.0017
h = 0.1	Actual error in x_2	Error order in x_2	Actual error in x_2	Error order in x_2
h	7.8179e-003	–	1.5918e-004	–
$h/2$	3.3141e-003	1.2382	3.9049e-005	2.0273
$h/4$	1.2815e-003	1.3707	9.6706e-006	2.0136
$h/8$	4.4226e-004	1.5349	2.4063e-006	2.0068
$h/16$	1.3666e-004	1.6943	6.0017e-007	2.0034
$h/32$	3.8747e-005	1.8184	1.4987e-007	2.0017

4-stage Runge-Kutta method of order 4 is given by the Butcher tableau

$$\begin{array}{c|cccc} 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ \hline & \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{1}{6} \end{array} .$$

The numerical results in Tables 4.4 and 4.5 clearly illustrate that the convergence order of the standard HERK4 method presented in [17] is reduced, but that of the HERK4 in the form (3.1) remains the same as that for ODEs.

Next, we solve the above initial value problem by the implicit midpoint method (IMID) in the form (3.6), which has the following Butcher tableau:

$$\begin{array}{c|c} \frac{1}{2} & \frac{1}{2} \\ \hline & 1 \end{array} .$$

Note that the midpoint Runge-Kutta method is of second order, and it is not stiffly accurate. The numerical results are displayed in Table 4.6 and well illustrate the preservation of the second order. In the last experiments, we implement a 2-stage implicit Runge-Kutta method (IRK2) in the form (3.6), whose coefficients are given in the Butcher tableau

$$\begin{array}{c|cc} \frac{1}{3} & \frac{5}{12} & \frac{-1}{12} \\ 1 & \frac{3}{4} & \frac{1}{4} \\ \hline & \frac{3}{4} & \frac{1}{4} \end{array} .$$

This method is known to be of third order for ODEs. In Table 4.7, the numerical results are displayed in the case that the Jacobian matrix of the Newton iteration is analytically given. Table 4.8 displays the numerical results in the case that the Jacobian arising in the Newton iteration is approximated by formula (3.11). We use as a stopping criterion for the Newton iteration that the difference between two consecutive Newton iterates is smaller than h^4.

TABLE 4.3
Errors of solutions to IVP (3.16) with $\omega = -100, \lambda = -1$.

$\alpha = 0.5$	Standard HERK2 method		HERK2 method (3.1)	
h = 0.1	Actual error in x_1	Error order in x_1	Actual error in x_1	Error order in x_1
h	2.6219e+006	–	2.3312e-002	–
$h/2$	6.2102e+043	-124.16	5.7176e-003	2.0276
$h/4$	4.0796e+006	123.52	1.4159e-003	2.0137
$h/8$	1.4370e+001	18.115	3.5233e-004	2.0068
$h/16$	1.9556e+000	2.8774	8.7875e-005	2.0034
$h/32$	3.9783e-001	2.2974	2.1943e-005	2.0017
h = 0.1	Actual error in x_2	Error order in x_2	Actual error in x_2	Error order in x_2
h	5.2542e+003	–	1.5918e-004	–
$h/2$	1.2445e+041	-124.16	3.9049e-005	2.0273
$h/4$	8.1755e+003	123.52	9.6706e-006	2.0136
$h/8$	8.7111e-002	16.518	2.4063e-006	2.0068
$h/16$	1.3119e-002	2.7312	6.0017e-007	2.0034
$h/32$	2.7071e-003	2.2768	1.4987e-007	2.0017

TABLE 4.4
Errors of the solution x_1 to the IVP (4.1) by standard HERK4 and HERK4 methods.

	Standard HERK4 method		HERK4 method (3.1)	
h = 0.2	Actual error in x_1	Error order in x_1	Actual error in x_1	Error order in x_1
h	1.1600e-004	–	4.1224e-005	–
$h/2$	1.5930e-005	2.8642	2.4838e-006	4.0529
$h/4$	2.0815e-006	2.9361	1.5166e-007	4.0336
$h/8$	2.6583e-007	2.9690	9.3585e-009	4.0185
$h/16$	3.3582e-008	2.9847	5.8102e-010	4.0096
$h/32$	4.2198e-009	2.9924	3.6193e-011	4.0048
$h/64$	5.2886e-010	2.9962	2.2569e-012	4.0033
$h/128$	6.6190e-011	2.9982	1.3634e-013	4.0491

TABLE 4.5
Errors of the solution x_2 to the IVP (4.1) by standard HERK4 and HERK4 methods.

	Standard HERK4 method		HERK4 method (3.1)	
h = 0.2	Actual error in x_2	Error order in x_2	Actual error in x_2	Error order in x_2
h	4.2672e-005	–	1.5571e-005	–
$h/2$	5.8604e-006	2.8642	9.3492e-007	4.0579
$h/4$	7.6573e-007	2.9361	5.6984e-008	4.0362
$h/8$	9.7794e-008	2.9690	3.5129e-009	4.0198
$h/16$	1.2354e-008	2.9847	2.1799e-010	4.0103
$h/32$	1.5524e-009	2.9924	1.3575e-011	4.0052
$h/64$	1.9456e-010	2.9962	8.4865e-013	3.9997
$h/128$	2.4350e-011	2.9982	5.2403e-014	4.0175

TABLE 4.6
Errors of the solution to the IVP (4.1) by the IMID method.

	IMID method		IMID method	
$h = 0.1$	Actual error in x_1	Error order in x_1	Actual error in x_2	Error order in x_2
h	1.1184e-002	–	1.5136e-003	–
$h/2$	2.7900e-003	2.0031	3.7759e-004	2.0031
$h/4$	6.9713e-004	2.0008	9.4347e-005	2.0008
$h/8$	1.7426e-004	2.0002	2.3583e-005	2.0002
$h/16$	4.3563e-005	2.0000	5.8957e-006	2.0000
$h/32$	1.0891e-005	2.0000	1.4739e-006	2.0000
$h/64$	2.7227e-006	2.0000	3.6848e-007	2.0000
$h/128$	6.8067e-007	2.0000	9.2119e-008	2.0000

TABLE 4.7
Errors of the solution to the IVP (4.1) by the IRK2 method.

	IRK2 method (3.6)		IRK2 method (3.6)	
$h = 0.1$	Actual error in x_1	Error order in x_1	Actual error in x_2	Error order in x_2
h	9.0149e-006	–	4.7991e-006	–
$h/2$	1.1346e-006	2.9901	6.0274e-007	2.9931
$h/4$	1.4207e-007	2.9976	7.5353e-008	2.9998
$h/8$	1.7769e-008	2.9991	9.4195e-009	2.9999
$h/16$	2.2216e-009	2.9997	1.1773e-009	3.0002
$h/32$	2.7773e-010	2.9999	1.4714e-010	3.0001
$h/64$	3.4712e-011	3.0002	1.8391e-011	3.0001
$h/128$	4.3379e-012	3.0004	2.2994e-012	2.9997

TABLE 4.8
Errors of the solution to the IVP (4.1) by the IRK2 method in the case of approximate Jacobian.

	IRK2 method (3.6)		IRK2 method (3.6)	
$h = 0.1$	Actual error in x_1	Error order in x_1	Actual error in x_2	Error order in x_2
h	8.6234e-006	–	4.5654e-006	–
$h/2$	1.1388e-006	2.9208	6.0677e-007	2.9115
$h/4$	1.4207e-007	3.0028	7.5292e-008	3.0106
$h/8$	1.7781e-008	2.9982	9.4208e-009	2.9986
$h/16$	2.2267e-009	2.9974	1.1777e-009	2.9998
$h/32$	2.8010e-010	2.9909	1.4735e-010	2.9987
$h/64$	3.6075e-011	2.9569	1.8485e-011	2.9948
$h/128$	5.1550e-012	2.8070	2.3478e-012	2.9770

5. Conclusion. In this paper, we have revisited the Runge-Kutta methods for solving structured strangeness-free DAEs of the form (1.1). It is shown that, instead of discretizing directly the DAEs systems, we apply the half-explicit and implicit Runge-Kutta methods to the reformulated form (2.5). Not only the convergence order and the stability are preserved, but also some conditions on the Runge-Kutta methods are relaxed. In essence, here we have shown that the reformulated form (2.5) behaves under discretization like a semi-explicit

DAE of index 1, while the strangeness-free DAE (1.1) behaves like a semi-explicit DAE of index 2 as explained in [17]. Thus, a wider class of methods can be used for efficiently solving the DAEs (1.1). The numerical solutions by the discretization schemes proposed in this paper also reflect better the stability characteristics (Lyapunov exponents, Sacker-Sell spectral intervals) of the original differential-algebraic equations. Furthermore, integrators with error control and automatic stepsize selection, which are based on popular embedded Runge-Kutta pairs such as Dormand-Prince, and Runge-Kutta-Chebyshev methods, which are well known for efficiently solving stiff problems, can be easily adopted to solving DAEs. In addition, symmetric collocation such as Gauss methods, which are not stiffly accurate but have good stability property in the context of boundary value problems, may also be considered for solving BVPs for DAEs of the form (1.1). It is worth investigating these topics in the future.

Acknowledgment. We would like to thank the referees for their very helpful comments that led to improvements of this paper. We are also grateful to Roswitha März for her valuable comments on an earlier version of the paper.

REFERENCES

[1] M. ARNOLD, K. STREHMEL, AND R. WEINER, *Half-explicit Runge-Kutta methods for semi-explicit differential-algebraic equations of index 1*, Numer. Math., 64 (1993), pp. 409–431.

[2] U. ASCHER AND L. PETZOLD, *Computer Methods for Ordinary Differential Equations and Differential-Algebraic equations*, SIAM, Philadelphia, 1998.

[3] K. E. BRENAN, S. L. CAMPBELL, AND L. R. PETZOLD, *Numerical Solution of Initial-Value Problems in Differential Algebraic Equations*, 2nd ed., SIAM, Philadelphia, 1996.

[4] M. V. BULATOV, V. H. LINH, AND L. S. SOLOVAROVA, *On BDF-based multistep schemes for some classes of linear differential-algebraic equations of index at most 2*, Acta Math. Vietnam., 41 (2016), pp. 715–730.

[5] Y. CAO, S. LI, L. PETZOLD, AND R. SERBAN, *Adjoint sensitivity analysis for differential-algebraic equations: the adjoint DAE system and its numerical solution*, SIAM J. Sci. Comput., 24 (2003), pp. 1076–1089.

[6] L. DIECI AND T. EIROLA, *On smooth decompositions of matrices*, SIAM J. Matrix Anal. Appl., 20 (1999), pp. 800–819.

[7] R. DOKCHAN, *Numerical Integration of Differential-Algebraic Equations with Harmless Critical Point*, PhD. Thesis, Math.-Nat. Fakultät, Humboldt-University of Berlin, Berlin, 2011.

[8] E. HAIRER, CH. LUBICH, AND M. ROCHE, *The Numerical Solution of Differential-Algebraic Systems by Runge-Kutta Methods*, Springer, Berlin, 1989.

[9] E. HAIRER, S. P. NØRSETT, AND G. WANNER, *Solving Ordinary Differential Equations. I. Nonstiff Problems*, 2nd ed., Springer, Berlin, 1993.

[10] E. HAIRER AND G. WANNER, *Solving Ordinary Differential Equations II—Stiff and Differential-Algebraic Problems*, 2nd ed., Springer, Berlin, 1996.

[11] I. HIGUERAS AND B. GARCIA-CELAYETA, *Runge-Kutta methods for DAEs. A new approach*, J. Comput. Appl. Math., 111 (1999), pp. 49–61.

[12] I. HIGUERAS, R. MÄRZ, AND C. TISCHENDORF, *Stability preserving integration of index-1 DAEs*, Appl. Num. Math., 45 (2003), pp. 175–200.

[13] P. KUNKEL AND V. MEHRMANN, *Differential-Algebraic Equations Analysis and Numerical Solution*, European Mathemantical Society, Zürich, 2006.

[14] ———, *Stability properties of differential-algebraic equations and spin-stabilized discretization*, Electron. Trans. Numer. Anal., 26 (2007), pp. 385–420.
http://etna.mcs.kent.edu/vol.26.2007/pp385-420.dir/pp385-420.pdf

[15] R. LAMOUR, R. MÄRZ, AND C. TISCHENDORF, *Differential-Algebraic Equations: A Projector Based Analysis*, Springer, Heidelberg, 2013.

[16] V. H. LINH AND V. MEHRMANN, *Approximation of spectral intervals and associated leading directions for linear differential-algebraic equations via smooth singular value decompositions*, SIAM J. Numer. Anal., 49 (2011), pp. 1810–1835.

[17] ———, *Efficient integration of matrix-valued non-stiff DAEs by half-explicit methods*, J. Comput. Appl. Math., 262 (2014), pp. 346–360.

[18] V. H. LINH, V. MEHRMANN, AND E. VAN VLECK, *QR methods and error analysis for computing Lyapunov and Sacker-Sell spectral intervals for linear differential-algebraic equations*, Adv. Comput. Math., 35 (2011), pp. 281–322.