Journal of Computational Mathematics
Vol.39, No.4, 2021, 599–620.
http://www.global-sci.org/jcm
doi:10.4208/jcm.2005-m2019-0238

IMPLICIT-EXPLICIT RUNGE-KUTTA-ROSENBROCK METHODS WITH ERROR ANALYSIS FOR NONLINEAR STIFF DIFFERENTIAL EQUATIONS*

Bin Huang and Aiguo Xiao
School of Mathematics and Computational Science & Hunan Key Laboratory for Computation and Simulation in Science and Engineering, Xiangtan University, Xiangtan 411105, China
Email: huangbin_xtu@sina.com, xag@xtu.edu.cn
Gengen Zhang[1)]
South China Research Center for Applied Mathematics and Interdisciplinary Studies, South China Normal University, Guangzhou 510631, China
Email: zhanggen036@163.com

Abstract

Implicit-explicit Runge-Kutta-Rosenbrock methods are proposed to solve nonlinear stiff ordinary differential equations by combining linearly implicit Rosenbrock methods with explicit Runge-Kutta methods. First, the general order conditions up to order 3 are obtained. Then, for the nonlinear stiff initial-value problems satisfying the one-sided Lipschitz condition and a class of singularly perturbed initial-value problems, the corresponding errors of the implicit-explicit methods are analysed. At last, some numerical examples are given to verify the validity of the obtained theoretical results and the effectiveness of the methods.

Mathematics subject classification: 65L04, 65L20, 65L06.
Key words: Stiff differential equations, Implicit-explicit Runge-Kutta-Rosenbrock method, Order conditions, Convergence.

1. Introduction

The systems of stiff ordinary differential equations (ODEs) are important mathematical models. They often appear in many fields of science and engineering and come also from space discretization of some initial-boundary value problems of partial differential equations (see, e.g., [1–6, 10–15, 21–23, 28–31]. In particular, many stiff ODEs can be rewritten in additive form whose terms have different stiffness properties, for example, their functions on the right side can be split to stiff part and nonstiff part.

For solving numerically stiff ODEs, Runge-Kutta (RK) methods are a common choice. On the one hand, explicit RK methods require less computing efforts and can be implemented easily, but the demand for their stability leads to the strict time step constraint. On the other hand, many implicit RK methods are often of good stability property, but need to solve the systems of nonlinear algebraic equations.

A good compromise is to apply linearly implicit Rosenbrock-type RK schemes, so-called W-methods (see, e.g., [2, 12, 13, 15, 16, 19, 24, 25, 27, 32]), which can avoid the solution of the systems of nonlinear algebraic equations and only require the solution of linear algebraic systems at each time step. Bassi et al. [2] focused on the implementation and assessment of linearly

* Received October 25, 2019 / Revised version received February 12, 2020 / Accepted May 6, 2020 / Published online May 25, 2021 /
1) Corresponding author

implicit Rosenbrock schemes as time integrators for the high-order discontinuous Galerkin space discretization of the compressible and incompressible Navier-Stokes equations. González-Pinto et al. [12] considered a new class of AMF-type W-methods for the time integration of a large system of ODEs and linear parabolic problems with mixed derivatives and constant coefficients.

For combining the advantages of explicit and implicit RK methods, implicit-explicit (IMEX) RK methods have been proposed and become a very active area of research (see, e.g., [3-11, 17, 18, 20–22, 30, 31, 34–37]). For instance, Ascher et al. [1] constructed the efficient IMEX RK methods with better stability regions than the best known IMEX multistep schemes over a wide parameter range. Boscarino et al. [3–7] focused on IMEX RK methods for differential-algebraic systems, hyperbolic systems, kinetic equations, fully nonlinear relaxation problems and stiff problems. Izzo et al. [20] investigated the construction of highly stable IMEX RK methods up to the order $p = 4$, where the implicit schemes are A-stable and the explicit schemes has strong stability property. Yue et al. [35] were devoted to the nonlinear stability and B-convergence of additive RK (ARK) methods for nonlinear stiff problems with multiple stiffness. Other efficient IMEX RK methods and their applicitions can be found in [11, 31] etc.

For reducing further computing efforts of IMEX RK methods, linearized implicit RK methods (such as Rosenbrock methods) can replace the implicit RK methods in IMEX RK methods. Ullrich et al. [28] introduced an operator-split Runge-Kutta-Rosenbrock (RKR) time discretization strategy for nonhydrostatic atmospheric models, and its temporal accuracy up to order 3 is achieved. Higueras et al. [18] considered additive semi-implicit RK (ASIRK) methods for stiff additive differential systems by combining diagonally implicit RK methods or linearized implicit RK methods with explicit RK methods, and constructed two 2-order 3-stage ASIRK schemes with low-storage requirements. Other efficient IMEX linearized RK methods can refer to [37]. But these works only considered the construction of efficient algorithms and the analysis of the linear stability, and no rigorous global error analysis was made.

Motivated by the above discussion, we consider further rigorous global error analysis and the construction of efficient algorithms for IMEX RKR methods for nonlinear stiff ODEs. The order conditions up to order 3 are obtained. For the nonlinear stiff initial-value problems satisfying the one-sided Lipschitz condition and a class of singularly perturbed initial-value problems, the errors of these methods are analysed rigorously. Moreover, 2-order 3-stage IMEX RKR methods are proposed and shown to be efficient, and they can overcome the severe time step restriction of explicit schemes and require just one Jacobian matrix evaluation per time step, thus the overall computational efforts are reduced obviously.

The rest of the paper is organized as follows. In Section 2, we present the IMEX RKR methods by combining implicit Rosenbrock methods with explicit RK methods for stiff ODEs. The order conditions of the IMEX RKR methods are given in Section 3. In Section 4, we give the convergence results of the IMEX RKR methods for the nonlinear stiff initial-value problems of ODEs with one-sided Lipschitz condition. In Section 5, the convergence results of the IMEX RKR methods for a class of singularly perturbed initial-value problems are obtained. In Section 6, some numerical examples are given to verify the obtained theoretical results.

2. IMEX RKR Methods

Consider the initial-value problems of nonlinear stiff ODEs

$$\begin{cases} y'(t) = F(t, y(t)) = f(t, y(t)) + g(t, y(t)), & t \in [0, T_e], \\ y(0) = y_0, & y_0 \in R^m, \end{cases} \tag{2.1}$$

where R^m denotes the m-dimensional Euclidean space, $f, g : D = [0, T_e] \times R^m \to R^m$ are the functions corresponding to stiff and non-stiff processes respectively. Suppose that the problems (2.1) have unique exact solution $y(t)$ and satisfy the following conditions (cf. [34]):

(c1) f satisfies the one-sided *Lipschitz* condition

$$\langle f(t,y) - f(t,z), y - z\rangle \le L_\mu \|y - z\|^2, \quad \forall (t,y),(t,z) \in D, \tag{2.2}$$

and g satisfies the classical *Lipschitz* condition

$$\|g(t,y) - g(t,z)\| \le L\|y - z\|, \quad \forall (t,y),(t,z) \in D, \tag{2.3}$$

here L_μ and L are one-sided *Lipschitz* constant and classical *Lipschitz* constant respectively, the norm $\|\cdot\|$ is induced by the standard inner product $\langle\cdot,\cdot\rangle$ on R^m.

(c2) f is sufficiently smooth on D, and its partial derivatives except for $\partial f/\partial y$ are bounded by moderately sized constants. g and its derivatives have the smoothness and boundedness required below.

(c3) The exact solution $y(t)$ and its derivatives have the smoothness and boundedness required below.

Problem (2.1) involves the stiff term f and the nonstiff term g. A natural approach to numerical solutions is to employ IMEX time discretization. In this paper, an explicit RK method is used for the non-stiff part g, and the implicit Rosenbrock method is used for the stiff part f. Then s-stage IMEX RKR methods are given by the formula

$$\begin{cases} y_{n+1} = y_n + h\sum\limits_{i=1}^{s} b_i k_i, \\ (I - h\gamma\mathcal{J})k_i = f(t_n + c_i h, Y_{n+1,i}) + g(t_n + c_i h, \widetilde{Y}_{n+1,i}) + h\mathcal{J}\sum\limits_{j=1}^{i-1}\gamma_{ij}k_j, \\ Y_{n+1,i} = y_n + h\sum\limits_{j=1}^{i-1} a_{ij}k_j, \\ \widetilde{Y}_{n+1,i} = y_n + h\sum\limits_{j=1}^{i-1} \widetilde{a}_{ij}k_j, \end{cases} \tag{2.4}$$

where $h > 0$ is a time stepsize, $y_n \approx y(t_n)$, I is identity matrix, $\mathcal{J} \in R^{m\times m}$ is a matrix, it is chosen usually as an approximation to the Jacobian matrix $\mathcal{J}_f(t_n) = \frac{\partial f}{\partial y}(t_n, y(t_n))$, and the coefficients $\gamma_{ii} \equiv \gamma > 0$, a_{ij}, $\widetilde{a}_{ij}$, b_i and c_i ($c_1 = 0$) satisfy

$$\sum_{j=1}^{i-1} a_{ij} = \sum_{j=1}^{i-1}\widetilde{a}_{ij} = c_i \quad (1 \le i \le s), \qquad \sum_{i=1}^{s} b_i = 1.$$

For convenience, we use the following abbreviations $\beta = (\beta_{ij})_{s\times s}$, $\widetilde{\beta} = (\widetilde{\beta}_{ij})_{s\times s}$ with $\beta_{ij} = a_{ij} + \gamma_{ij}$, $\widetilde{\beta}_{ij} = \widetilde{a}_{ij} + \gamma_{ij}$, $\beta_{ij} = \widetilde{\beta}_{ij} = 0$ ($j \ge i$), and

$$\begin{aligned} &A = (a_{ij})_{s\times s} = (a_1, a_2, \cdots, a_s)^T, \qquad a_i^T = (a_{i1}, a_{i2}, \cdots, a_{i,i-1}, 0, \cdots, 0), \\ &\widetilde{A} = (\widetilde{a}_{ij})_{s\times s} = (\widetilde{a}_1, \widetilde{a}_2, \cdots, \widetilde{a}_s)^T, \qquad \widetilde{a}_i^T = (\widetilde{a}_{i1}, \widetilde{a}_{i2}, \cdots, \widetilde{a}_{i,i-1}, 0, \cdots, 0), \\ &e = (1, 1, \cdots, 1)^T, \qquad b^T = (b_1, b_2, \cdots, b_s). \end{aligned}$$

For Method (2.4), we need to solve respectively s systems of m linear algebraic equations with unknown k_i and the same coefficient matrix $I - h\gamma\mathcal{J}$ per time step, thus the overall computational effort is reduced obviously.

3. Order Conditions of IMEX RKR Methods

In this section, the order conditions is obtained for the autonomous differential equation and the error at one step (local error). Suppose that all elementary differentials are evaluated at $y_0 = y(t_0)$. Applying the IMEX RKR methods (2.4) to the autonomous differential equation

$$y'(t) = f(y) + g(y) \tag{3.1}$$

yields

$$y_1 = y_0 + \sum_{i=1}^{s} b_i k_i, \tag{3.2a}$$

$$k_i = h\left(f(y_0 + \sum_{j=1}^{i-1} a_{ij}k_j) + g(y_0 + \sum_{j=1}^{i-1} \widetilde{a}_{ij}k_j) + \mathcal{J}\sum_{j=1}^{i} \gamma_{ij}k_j\right), \tag{3.2b}$$

where $\mathcal{J} = \frac{\partial f}{\partial y}(y(t_0))$. Let w^J is the J-component of w and $w^J_j = \frac{\partial w^J}{\partial y^j}$. As in tensor notation, we denote the components of vectors by superscript indices. Then the system (3.1) can be written in tensor notation, and the method (3.2) reads

$$\begin{aligned}
y_1^J &= y_0^J + \sum_j b_j k_i^J,\\
k_i^J &= hf^J(u_i) + hg^J(\widetilde{u}_i) + h\sum_K f_K^J(y_0)\sum_j \gamma_{ij}k_j^K,\\
u_i^J &= y_0^J + \sum_j a_{ij}k_j^J,\\
\widetilde{u}_i^J &= y_0^J + \sum_j \widetilde{a}_{ij}k_j^J.
\end{aligned}$$

In order to obtain the order conditions for IMEX RKR methods, as indicated in [14], we have to compare the Taylor series of y_1^J with that of the exact solution. Now, we compute the derivatives of y_1^J and k_i^J with respect to h at $h = 0$. Inserting the chain rule

$$(f^J(u_j) + g^J(\widetilde{u}_j))' = \sum_K f_K^J(u_j)(u_j^K)' + \sum_K g_K^J(\widetilde{u}_j)(\widetilde{u}_j^K)', \tag{3.3a}$$

$$\begin{aligned}
(f^J(u_j) + g^J(\widetilde{u}_j))'' = &\sum_{K,L} f_{KL}^J(u_j)(u_j^K)'(u_j^L)' + \sum_{K,L} g_{KL}^J(\widetilde{u}_j)(\widetilde{u}_j^K)'(\widetilde{u}_j^L)'\\
&+ \sum_K (f_K^J(u_j)(u_j^K)'' + g_K^J(\widetilde{u}_j)(\widetilde{u}_j^K)'')
\end{aligned} \tag{3.3b}$$

into the Leibniz's rule

$$(k_j^J)^{(q)}\,|_{h=0} = q\left((f^J(u_j))^{(q-1)} + (g^J(\widetilde{u_j}))^{(q-1)}\right)\Big|_{h=0} + q\sum_K f_K^J(y_0)\sum_l \gamma_{jl}(k_l^K)^{(q-1)}\Big|_{h=0}, \tag{3.4}$$

we obtain recursively

$$\begin{aligned}
&(k_j^J)^{(0)}\mid_{h=0}=0,\\
&(k_j^J)^{(1)}\mid_{h=0}=f^J+g^J,\\
&(k_j^J)^{(2)}\mid_{h=0}=2\sum_K f_K^J F^K\sum_l \beta_{jl}+2\sum_K g_K^J F^K \sum_l \widetilde{a}_{jl},\\
&(k_j^J)^{(3)}\mid_{h=0}=3\sum_{K,L} f_{K,L}^J F^K F^L\sum_{i,l} a_{ji}a_{jl}+6\sum_{K,L} f_K^J f_L^K F^L\sum_{i,l}\beta_{ji}\beta_{il}\\
&\qquad\qquad +6\sum_{K,L} f_K^J g_L^K F^L\sum_{i,l}\beta_{ji}\widetilde{a}_{il}+3\sum_{K,L} g_{K,L}^J F^K F^L\sum_{i,l}\widetilde{a}_{ji}\widetilde{a}_{jl}\\
&\qquad\qquad +6\sum_{K,L} g_K^J f_L^K F^L\sum_{i,l}\widetilde{a}_{ji}\beta_{il}+6\sum_{K,L} g_K^J g_L^K F^L\sum_{i,l}\widetilde{a}_{ji}\widetilde{a}_{il},
\end{aligned}$$

where $F^K = f^K + g^K$. Comparing the derivatives of the numerical solution ($q \geq 1$)

$$(y_1^J)^{(q)}\mid_{h=0}=\sum_j b_j (k_j^J)^{(q)}\mid_{h=0} \tag{3.5}$$

with those of the exact solution ($q \geq 1$)

$$\begin{aligned}
&(y^J)'\mid_{t=0}=f^J+g^J,\\
&(y^J)^{(2)}\mid_{t=0}=\sum_K f_K^J F^K+\sum_K g_K^J F^K,\\
&(y^J)^{(3)}\mid_{t=0}=\sum_{K,L} f_{KL}^J F^K F^L+\sum_{K,L} f_K^J f_L^K F^L+\sum_{K,L} f_K^J g_L^K F^L\\
&\qquad\qquad +\sum_{K,L} g_{KL}^J F^K F^L+\sum_{K,L} g_K^J f_L^K F^L+\sum_{K,L} g_K^J g_L^K F^L,
\end{aligned}$$

we obtain the following conditions for order p up to 3

$$p=1:\quad \sum b_j=1, \tag{3.6a}$$

$$p=2:\quad \sum b_j(a_{ji}+\gamma_{ji})=\frac{1}{2},\qquad \sum b_j\widetilde{a}_{ji}=\frac{1}{2}, \tag{3.6b}$$

$$\begin{aligned}
p=3:\quad &\sum b_j a_{ji}a_{jl}=\frac{1}{3}, &&\sum b_j\widetilde{a}_{ji}\widetilde{a}_{jl}=\frac{1}{3},\\
&\sum b_j(a_{ji}+\gamma_{ji})(a_{il}+\gamma_{il})=\frac{1}{6}, &&\sum b_j\widetilde{a}_{ji}(a_{il}+\gamma_{il})=\frac{1}{6},\\
&\sum b_j(a_{ji}+\gamma_{ji})\widetilde{a}_{il}=\frac{1}{6}, &&\sum b_j\widetilde{a}_{ji}\widetilde{a}_{il}=\frac{1}{6}.
\end{aligned} \tag{3.6c}$$

4. Convergence Analysis

We now turn to studying the convergence of the IMEX RKR methods for two classes of nonlinear stiff ODEs. In this section, we consider further rigorous global error analysis of IMEX RKR methods for the nonlinear problem (2.1), and the error estimate results along with their proofs are given. For convenience, we denote $w = (I - h\gamma\mathcal{J})^{-1}h\mathcal{J}$ with

$$\mathcal{J}=\begin{cases}\mathcal{J}_f(t_n)+I_O(h), & \mu(\mathcal{J}_f(t_n))\leq-\mu_0<0,\\ \mathcal{J}_f(t_n), & \mu(\mathcal{J}_f(t_n))\leq 0,\end{cases} \tag{4.1}$$

where $\mathcal{J}_f(t_n) = \frac{\partial f}{\partial y}(t_n, y(t_n))$, the matrix $I_O(h)$ has the same number of rows and columns as $\mathcal{J}$ and its entries are all $O(h)$, $\mu(\cdot)$ denotes the logarithmic norm of matrix, μ_0 is a constant, and there exist constants h^*, d_j such that for $\forall t, t+h \in [0, T_e]$, $|h| \leq h^*$,

$$\mathcal{J}_f(t+h) - \mathcal{J}_f(t) = h(\mathcal{J}_f(t)E_1 + E_2) \tag{4.2}$$

holds, here $E_j = E_j(t,h) \in R^{m\times m}$, and $\| E_j \| \leq d_j, j = 1, 2$.

Firstly, we give a lemma to illustrate some properties, which will be used in the following.

Lemma 4.1 ([13]). *Let $R(z)$ be a rational function and the complex matrix U satisfy*

$$Re\langle x, Ux\rangle \leq L_\alpha \|x\|^2, \quad \forall x \in C^m, L_\alpha \in R$$

for some inner product and the corresponding norm. Then we have $\|R(\mathcal{J})\| \leq \phi_R(L_\alpha)$ in the corresponding operator-norm, here $\phi_R(L_\alpha) = \sup_{Re\ z\ \leq\ L_\alpha} |R(z)|$.

Based on Lemma 4.1 and the related results in [27, 32], for $0 < h \leq h_0$, we have $\mu(\mathcal{J}) \leq 0$, and

$$I + \gamma w = (I - h\gamma\mathcal{J})^{-1} = I_O(1), \tag{4.3a}$$
$$w^k = ((I - h\gamma\mathcal{J})^{-1} h\mathcal{J})^k = I_O(1), \qquad k = 1, \cdots, s+1, \tag{4.3b}$$
$$\|w\| = \| (I - h\gamma\mathcal{J})^{-1} h\mathcal{J} \| \leq \frac{1}{\gamma}, \tag{4.3c}$$

where the matrix $I_O(1)$ has the same number of rows and columns as $\mathcal{J}$ and its entries are all $O(1)$.

In order to obtain the error estimate for IMEX RKR methods, we state the inner stage values $Y_{n+1,i}$, $\widetilde{Y}_{n+1,i}$ and the numerical solution y_{n+1} in the following theorem.

Theorem 4.2. *For the inner stage values $Y_{n+1,i}$, $\widetilde{Y}_{n+1,i}$ and the numerical solution y_{n+1}, we have*

$$Y_{n+1,i} = R_{0,i}(w)y_n + h\sum_{k=0}^{i-2}\sum_{l=1}^{i-1} a_i^T \beta^k e_l w^k (I + \gamma w)\Phi_l, \tag{4.4a}$$
$$\widetilde{Y}_{n+1,i} = \widetilde{R}_{0,i}(w)y_n + h\sum_{k=0}^{i-2}\sum_{l=1}^{i-1} \widetilde{a}_i^T \beta^k e_l w^k (I + \gamma w)\Phi_l, \tag{4.4b}$$
$$y_{n+1} = R_{0,s+1}(w)y_n + h\sum_{k=0}^{s-1}\sum_{l=1}^{s} b^T \beta^k e_l w^k (I + \gamma w)\Phi_l, \tag{4.4c}$$

where

$$\Phi_l = f_l + g_l - \mathcal{J}Y_{n+1,l}, \qquad f_l = f(t_n + c_l h, Y_{n+1,l}), \qquad g_l = g(t_n + c_l h, \widetilde{Y}_{n+1,l}), \tag{4.5a}$$
$$R_{0,i}(w) = I + \sum_{k=0}^{i-2} a_i^T \beta^k e w^{k+1}, \quad \widetilde{R}_{0,i}(w) = I + \sum_{k=0}^{i-2} \widetilde{a}_i^T \beta^k e w^{k+1}, \quad i = 1, \cdots, s, \tag{4.5b}$$
$$R_{0,s+1}(w) = I + \sum_{k=0}^{s-1} b^T \beta^k e w^{k+1}. \tag{4.5c}$$

Proof. Let $\mathbf{K} = (k_1^T, \cdots, k_s^T)^T$, $\mathbf{Y}_{n+1} = (Y_{n+1,1}^T, \cdots, Y_{n+1,s}^T)^T$, $\Phi = (\Phi_1^T, \cdots, \Phi_s^T)^T$. Then the method (2.4) can be rewritten into the vector form

$$\begin{aligned} Y_{n+1,i} &= y_n + h(a_i^T \otimes I_m)\mathbf{K}, \\ \mathbf{Y}_{n+1} &= e \otimes y_n + h(A \otimes I_m)\mathbf{K}, \end{aligned}$$

where $\otimes$ is the matrix *Kronecker* product, and

$$k_i = (I - h\gamma\mathcal{J})^{-1}\Phi_i + (I - h\gamma\mathcal{J})^{-1}\mathcal{J}y_n + (I - h\gamma\mathcal{J})^{-1}h\mathcal{J}\sum_{j=1}^{i-1}\beta_{ij}k_j,$$

where Φ_i is defined by (4.5). Using $(I - \gamma h\mathcal{J})^{-1} = I + \gamma w$, we obtain

$$h(I - \beta \otimes w)\mathbf{K} = h(I_s \otimes (I + \gamma w))\Phi + (I_s \otimes w)(e \otimes y_n). \tag{4.6}$$

It follows from the nilpotent matrix A with $A^k = 0$ that $(I - A)^{-1} = I + A + \cdots + A^{k-1}$. Then based on the formula (4.6), it holds that

$$\begin{aligned} h\mathbf{K} =&(I - \beta \otimes w)^{-1}\Big(h(I_s \otimes (I + \gamma w))\Phi + e \otimes (wy_n)\Big) \\ =&\sum_{k=0}^{s-1}(\beta \otimes w)^k\Big(h(I_s \otimes (I + \gamma w))\Phi + e \otimes (wy_n)\Big) \\ =&\sum_{k=0}^{s-1}(\beta^k \otimes w^k)\Big(h(I_s \otimes (I + \gamma w))\Phi + e \otimes (wy_n)\Big), \end{aligned}$$

and

$$\begin{aligned} Y_{n+1,i} &= y_n + \sum_{k=0}^{s-1}\Big((a_i^T\beta^k) \otimes w^k)(hI_s \otimes (I + \gamma w)\Phi + e \otimes (wy_n)\Big) \\ &= y_n + \sum_{k=0}^{s-1}\Big(h(a_i^T\beta^k) \otimes (w^k(I + \gamma w))\Phi + (a_i^T\beta^k e) \otimes (w^{k+1}y_n)\Big) \\ &= \Big(I + \sum_{k=0}^{s-1}(a_i^T\beta^k e) \otimes w^{k+1}\Big)y_n + \sum_{k=0}^{s-1}\Big(h(a_i^T\beta^k) \otimes (w^k(I + \gamma w))\Big)\Phi. \end{aligned}$$

It follows from $a_i^T\beta^k = 0$ $(k \geq i-1)$ that

$$Y_{n+1,i} = \Big(I + \sum_{k=0}^{i-2}a_i^T\beta^k e w^{k+1}\Big)y_n + \sum_{k=0}^{i-2}\Big(h(a_i^T\beta^k) \otimes (w^k(I + \gamma w))\Big)\Phi.$$

Furthermore,

$$(a_i^T\beta^k) \otimes (w^k(I + \gamma w)) = (a_i^T\widehat{\beta}_1, \cdots, a_i^T\widehat{\beta}_s) \otimes (w^k(I + \gamma w)),$$

where $\widehat{\beta}_l = \beta^k e_l$, $l = 1, \cdots, s$, thus (4.4a) holds. By substituting $\widetilde{a}_i^T$ and b^T for a_i^T respectively, (4.4b) and (4.4c) can be shown similarly. □

Theorem 4.3. Suppose that the computation is evaluated at $y(t_n)$, i.e. $y_n = y(t_n)$, and the simplifying conditions

$$a_i^T c^{l-1} = \frac{1}{l} c_i^l, \quad \widetilde{a}_i^T c^{l-1} = \frac{1}{l} c_i^l, \qquad l = 1, \cdots, q_i, \qquad i = 1, \cdots, s, \tag{4.7a}$$

$$a_i^T \beta^k \Big((\beta + \gamma I) l c^{l-1} - c^l \Big) = 0, \qquad k = 0, 1, \cdots, i-2, \qquad l = 1, \cdots, q_i, \tag{4.7b}$$

$$\widetilde{a}_i^T \beta^k \Big((\beta + \gamma I) l c^{l-1} - c^l \Big) = 0, \qquad k = 0, 1, \cdots, i-2, \qquad l = 1, \cdots, q_i, \tag{4.7c}$$

$$a_i^T \beta^k e_j = 0, \qquad \widetilde{a}_i^T \beta^k e_j = 0, \qquad k = 1, \cdots, i-2, \qquad \text{if} \ \ q_j < q_i - 1 \tag{4.7d}$$

holds, where $c^k = (c_1^k, \cdots, c_s^k)^T$, and e_j denotes the j-th unit vector. Then there exists a constant $h_1 > 0$ such that for $h \le h_1$, $i = 1, \cdots, s+1$, we have the local error estimates

$$\| Y_{n+1,i} - y(t_n + c_i h) \| = O(h^{q_i + 1}), \tag{4.8a}$$

$$\| \widetilde{Y}_{n+1,i} - y(t_n + c_i h) \| = O(h^{q_i + 1}). \tag{4.8b}$$

Proof. According to the Taylor expansion

$$\begin{aligned}
& f_l + g_l - y'(t_n + c_l h) \\
=& f(t_n + c_l h, Y_{n+1,l}) + g(t_n + c_l h, \widetilde{Y}_{n+1,l}) \\
& \quad - \big(f(t_n + c_l h, y(t_n + c_l h)) + g(t_n + c_l h, y(t_n + c_l h)) \big) \\
=& \Big(\mathcal{J}_f(t_n + c_l h) + M_l \Big) \triangle_{n+1,l} + \Big(g_y(t_n, y(t_n)) + I_O(h) \Big) \widetilde{\triangle}_{n+1,l},
\end{aligned}$$

where $\triangle_{n+1,l} = Y_{n+1,l} - y(t_n + c_l h)$, $\widetilde{\triangle}_{n+1,l} = \widetilde{Y}_{n+1,l} - y(t_n + c_l h)$, M_l and $g_y(t, y)$ are bounded, and

$$\begin{aligned}
\mathcal{J}_f(t_n + c_l h) + M_l &= \int_0^1 f_y(t_n + c_l h, y(t_n + c_l h) + \theta \triangle_{n+1,l}) d\theta, \\
g_y(t_n, y(t_n)) + I_O(h) &= \int_0^1 g_y(t_n + c_l h, y(t_n + c_l h) + \theta \widetilde{\triangle}_{n+1,l}) d\theta,
\end{aligned}$$

we can obtain

$$\begin{aligned}
& f_l + g_l - y'(t_n + c_l h) + \mathcal{J} y(t_n + c_l h) - \mathcal{J} Y_{n+1,l} \\
=& f_l + g_l - y'(t_n + c_l h) - \mathcal{J} \triangle_{n+1,l} \\
=& \Big(\mathcal{J}_f(t_n + c_l h) + M_l - \mathcal{J} \Big) \triangle_{n+1,l} + \Big(g_y(t_n, y(t_n)) + I_O(h) \Big) \widetilde{\triangle}_{n+1,l} \\
=& \Big(c_l h \mathcal{J}_f(t_n) E_1 + M_l + I_O(h) \Big) \triangle_{n+1,l} + \Big(g_y(t_n, y(t_n)) + I_O(h) \Big) \widetilde{\triangle}_{n+1,l} \\
=& \Big(c_l h \mathcal{J} E_1 + M_l + I_O(h) \Big) \triangle_{n+1,l} + \Big(g_y(t_n, y(t_n)) + I_O(h) \Big) \widetilde{\triangle}_{n+1,l},
\end{aligned}$$

where $\| E_1 \| \le d_1$. By inserting $f_l + g_l - \mathcal{J} Y_{n+1,l}$ into (4.4a), it holds that

$$Y_{n+1,i} - y(t_n + c_i h) = \overline{A} + \overline{B}, \tag{4.9a}$$

$$\widetilde{Y}_{n+1,i} - y(t_n + c_i h) = \widetilde{A} + \widetilde{B}, \tag{4.9b}$$

where

$$\overline{A} = -y(t_n + c_i h) + \Big(I + \sum_{k=0}^{i-2} a_i^T \beta^k e w^{k+1}\Big) y_n + h \sum_{k=0}^{i-2} \sum_{j=1}^{i-1} a_i^T \beta^k A_1, \tag{4.10a}$$

$$\widetilde{A} = -y(t_n + c_i h) + \Big(I + \sum_{k=0}^{i-2} \widetilde{a}_i^T \beta^k e w^{k+1}\Big) y_n + h \sum_{k=0}^{i-2} \sum_{j=1}^{i-1} \widetilde{a}_i^T \beta^k A_1, \tag{4.10b}$$

$$A_1 = e_j w^k (I + \gamma w)\Big(y'(t_n + c_j h) - \mathcal{J} y(t_n + c_j h)\Big), \tag{4.10c}$$

$$\overline{B} = h \sum_{k=0}^{i-2} \sum_{j=1}^{i-1} a_i^T \beta^k e_j w^k (I + \gamma w) B_1, \tag{4.10d}$$

$$\widetilde{B} = h \sum_{k=0}^{i-2} \sum_{j=1}^{i-1} \widetilde{a}_i^T \beta^k e_j w^k (I + \gamma w) B_1, \tag{4.10e}$$

$$B_1 = \Big(c_l h \mathcal{J} E_1 + M_l + I_O(h)\Big) \triangle_{n+1,l} + \Big(g_y(t_n, y(t_n)) + I_O(h)\Big) \widetilde{\triangle}_{n+1,l}. \tag{4.10f}$$

Now we use the Taylor expansion for $y'(t_n + c_i h), y(t_n + c_i h)$. Then it holds that

$$\begin{aligned}\overline{A} =& - y(t_n + c_i h) + \Big(I + \sum_{k=0}^{i-2} a_i^T \beta^k e w^{k+1}\Big) y_n \\ &+ h \sum_{l=0}^{q_i+1} \sum_{k=0}^{i-2} \sum_{j=1}^{i-1} a_i^T \beta^k e_j c_j^l \frac{h^l}{l!} w^k (I + \gamma w)\Big(y^{(l+1)}(t_n) - \mathcal{J} y^{(l)}(t_n)\Big) \\ &+ h \sum_{k=0}^{i-2} \sum_{j=1}^{i-1} a_i^T \beta^k e_j w^k (I + \gamma w)\Big(O(h^{q_i+2}) - \mathcal{J} O(h^{q_i+2})\Big).\end{aligned}$$

Based on (4.3), (4.7) and the fact that

$$\sum_{j=1}^{i-1} a_i^T \beta^k e_j c_j^l = a_i^T \beta^k c^l, \qquad (I + \gamma w)^{-1} w = h \mathcal{J},$$

it follows that

$$\begin{aligned}\overline{A} =& - y(t_n + c_i h) + y(t_n) + \sum_{l=1}^{q_i} \bigg(\sum_{k=0}^{i-2} a_i^T \beta^k (l c^{l-1} w^k (I + \gamma w) \\ &- c^l w^{k+1})\bigg) \frac{h^l}{l!} y^{(l)}(t_n) + \mathbb{A} + O(h^{q_i+2}) \\ =& - y(t_n + c_i h) + y(t_n) + \sum_{l=1}^{q_i} \frac{c_i^l h^l}{l!} y^{(l)}(t_n) + \mathbb{A} + O(h^{q_i+1}) = O(h^{q_i+1}),\end{aligned}$$

where

$$\mathbb{A} = \frac{1}{(q_i+1)!} \sum_{k=1}^{i-2} a_i^T \beta^k \Big((q_i+1) c^{q_i} w^k (I + \gamma w) - c^{q_i+1} w^{k+1}\Big) h^{q_i+1} y^{(q_i+1)}(t_n) = O(h^{q_i+1}).$$

From (2.4), (4.3), (4.9) and Lemma 4.1, there exist the integer $q_j > 0$ such that for $h \le h_1$,

$$\overline{B} = \sum_{k=0}^{i-2} \sum_{j=1}^{i-1} a_i^T \beta^k e_j w^k O(h^{q_j+2})$$

holds. Moreover, from (4.7), we find (4.8a) holds. In the same way, (4.8b) can be obtained. □

The following result can be carried out similarly as Theorem 4.3, and we skip the proof and give only the result here.

Corollary 4.4. *Suppose that the computation is evaluated at* $y(t_n)$, *i.e.* $y_n = y(t_n)$, *and*

$$b^Tc^{l-1} = \frac{1}{l}, \qquad l = 1, \cdots, q, \tag{4.11a}$$

$$b^T\beta^k((\beta+\gamma I)lc^{l-1} - c^l) = 0, \qquad k = 0, \cdots, s-1, \quad l = 1, \cdots, q, \tag{4.11b}$$

$$b^T\beta^k e_j = 0, \quad b^T\widetilde{\beta}^k e_j = 0, \qquad k = 0, 1, \cdots, i-2, \quad \text{if } q_j < q-1. \tag{4.11c}$$

Then $\|y_{n+1} - y(t_{n+1})\| = O(h^{q+1})$.

With the help of the preceding Theorems 4.2, 4.3 and Corollary 4.4, we can obtain the following global error estimate of the IMEX RKR methods (2.4) applied to (2.1).

Theorem 4.5. *Let* $a_i^T\beta^j = \widetilde{a}_i^T\widetilde{\beta}^j, i = 1, \cdots, s,\ j = 0, 1, \cdots, s-2$, *the stability function* $R_{0,s+1}(\xi)$ *be strongly A-acceptable (i.e.* $|R_{0,s+1}(\infty)| < 1$ *and A-acceptable), and* (4.11) *hold. Then there exists a constant* $h_2 > 0$ *such that for* $h \le h_2$, *it follows from* $\| y_0 - y(t_0) \| = O(h^q)$ *that we have the global error estimate*

$$\|y_n - y(t_n)\| = O(h^q). \tag{4.12}$$

Proof. Let $\widehat{y}_{n+1}$ be the numerical solution by the method (2.4) with the value $y_n = y(t_n)$. Denote

$$\widehat{y}_{n+1} = R_{0,s+1}(w)y_n + h\sum_{k=0}^{s-1}\sum_{l=1}^{s} b^T\beta^k e_l w^k(I+\gamma w)(\widehat{f}_l + \overline{g}_l - \mathcal{J}\widehat{Y}_{n+1,l}), \tag{4.13a}$$

$$\widehat{Y}_{n+1,i} = R_{0,i}(w)y_n + h\sum_{k=0}^{i-2}\sum_{l=1}^{i-1} a_i^T\beta^k e_l w^k(I+\gamma w)(\widehat{f}_l + \overline{g}_l - \mathcal{J}\widehat{Y}_{n+1,l}), \tag{4.13b}$$

$$\overline{Y}_{n+1,i} = \widetilde{R}_{0,i}(w)y_n + h\sum_{k=0}^{i-2}\sum_{l=1}^{i-1} \widetilde{a}_i^T\beta^k e_l w^k(I+\gamma w)(\widehat{f}_l + \overline{g}_l - \mathcal{J}\widehat{Y}_{n+1,l}), \tag{4.13c}$$

$$\nabla_{n+1,i} = Y_{n+1,i} - \widehat{Y}_{n+1,i}, \quad \widetilde{\nabla}_{n+1,i} = \widehat{Y}_{n+1,i} - \overline{Y}_{n+1,i}, \tag{4.13d}$$

where $\widehat{f}_l = f(t_n + c_lh, \widehat{Y}_{n+1,l})$, $\overline{g}_l = g(t_n + c_lh, \overline{Y}_{n+1,l})$. Based on the fact that

$$\begin{aligned}
&f_l + g_l - \mathcal{J}Y_{n+1,l} - \widehat{f}_l - \overline{g}_l + \mathcal{J}\widehat{Y}_{n+1,l}\\
=&f_l - \widehat{f}_l + g_l - \overline{g}_l - \mathcal{J}(Y_{n+1,l} - \widehat{Y}_{n+1,l})\\
=&\int_0^1 f_y(t_n + c_lh, \widehat{Y}_{n+1,l} + \theta(Y_{n+1,l} - \widehat{Y}_{n+1,l}))d\theta(Y_{n+1,l} - \widehat{Y}_{n+1,l})\\
&\quad + \int_0^1 g_y(t_n + c_lh, \overline{Y}_{n+1,l} + \theta(\widetilde{Y}_{n+1,l} - \overline{Y}_{n+1,l}))d\theta\widetilde{\nabla}_{n+1,l} - \mathcal{J}\nabla_{n+1,l}\\
=&\Big(\mathcal{J}_f(t_n + c_lh) - \mathcal{J} + M_l\Big)\nabla_{n+1,l} + \Big(g_y(t_n, y(t_n)) + I_O(h)\Big)\widetilde{\nabla}_{n+1,l}\\
=&G_l\nabla_{n+1,l} + \widetilde{G}_l\widetilde{\nabla}_{n+1,l},
\end{aligned}$$

where $G_l = c_l h\mathcal{J}E_1 + I_O(h) + M_l$, $\widetilde{G}_l = g_y(t_n, y(t_n)) + I_O(h)$, and g_y, M_l are bounded matrixes, we obtain

$$\begin{aligned}&y_{n+1} - \widehat{y}_{n+1}\\ =&R_{0,s+1}(w)(y_n - y(t_n)) + h\sum_{k=0}^{s-1}\sum_{l=1}^{s} b^T\beta^k e_l w^k (I+\gamma w)(G_l\nabla_{n+1,l} + \widetilde{G}_l\widetilde{\nabla}_{n+1,l}).\end{aligned} \tag{4.14}$$

It follows from $a_i^T\beta^k = \widetilde{a}_i^T\widetilde{\beta}^k$ that

$$\begin{aligned}\nabla_{n+1,i} =&R_{0,i}(w)(y_n - y(t_n))\\ &+ h\sum_{k=0}^{s-1}\sum_{l=1}^{s} a_i^T\beta^k e_l w^k (I+\gamma w)\Big(f_l + g_l - \mathcal{J}Y_{n+1,l} - \widehat{f}_l - \overline{g}_l + \mathcal{J}\widehat{Y}_{n+1,l}\Big)\\ =&R_{0,i}(w)(y_n - y(t_n)) + h\sum_{k=0}^{s-1}\sum_{l=1}^{s} a_i^T\beta^k e_l w^k (I+\gamma w)\Big(G_i\nabla_{n+1,i} + \widetilde{G}_i\widetilde{\nabla}_{n+1,i}\Big).\end{aligned}$$

Let $P = (P_1, \cdots, P_s)^T$, $\widetilde{P} = (\widetilde{P}_1, \cdots, \widetilde{P}_s)^T$, and

$$\begin{aligned}&P_j = \sum_{k=0}^{s-1}\sum_{l=1}^{s} a_j^T\beta^k e_l w^k (I+\gamma w)G_l, \qquad \widetilde{P}_j = \sum_{k=0}^{s-1}\sum_{l=1}^{s} a_j^T\beta^k e_l w^k (I+\gamma w)\widetilde{G}_l,\\ &R = diag(R_{0,1}(w), \cdots, R_{0,s}(w))^T,\\ &\nabla = (\nabla_{n+1,1}, \cdots, \nabla_{n+1,s})^T, \quad \widetilde{\nabla} = (\widetilde{\nabla}_{n+1,1}, \cdots, \widetilde{\nabla}_{n+1,s})^T.\end{aligned}$$

Then we can rewrite the above formula into the vector form

$$\nabla = R((y_n - y(t_n)) \otimes e) + h(P\nabla + \widetilde{P}\widetilde{\nabla}).$$

Moreover,

$$\begin{aligned}w^k(I+\gamma w)G_l &= c_l w^{k+1} + w^k(I+\gamma w)(M_l + I_O(h)),\\ w^k(I+\gamma w)\widetilde{G}_l &= w^k(I+\gamma w)\Big(g_y(t_n, y(t_n)) + I_O(h)\Big).\end{aligned}$$

It follows from (4.3) that $\| R_{0,i} \|$ is bounded, $w^k(I+\gamma w)G_l = I_O(1)$ and $w^k(I+\gamma w)\widetilde{G}_l = I_O(1)$. Then

$$\nabla = (I + I_O(h))R((y_n - y(t_n)) \otimes e) + O(h)\widetilde{\nabla}.$$

In the same way,

$$\widetilde{\nabla} = (I + I_O(h))\widetilde{R}((y_n - y(t_n)) \otimes e) + O(h)\nabla.$$

Consequently,

$$\|\nabla_{n+1,i}\| \le C\|y_n - y(t_n)\|, \quad \|\widetilde{\nabla}_{n+1,i}\| \le \widetilde{C}\|y_n - y(t_n)\|, \tag{4.15}$$

where $C, \widetilde{C}$ are constants. Because $R_{0,s+1}(\xi)$ are strongly A-acceptable, it follows from (4.14) and (4.15) that

$$\begin{aligned}&\|y_{n+1} - \widehat{y}_{n+1}\|\\ \le&\|R_{0,s+1}(\omega)\|\|y_n - y(t_n)\| + hc_1\|y_n - y(t_n)\|\\ \le&(1 + c_1 h)\|y_n - y(t_n)\|.\end{aligned}$$

Therefore we can obtain $\|\widehat{y}_{n+1} - y(t_{n+1})\| \le dh^{q+1}$ by using Corollary 4.4, where the constant d doesn't depend on the stiffness of the problem. Moreover, it follows that

$$
\begin{aligned}
&\|y_{n+1} - y(t_{n+1})\| \\
\le&\|y_{n+1} - \widehat{y}_{n+1}\| + \|\widehat{y}_{n+1} - y(t_{n+1})\| \\
\le&(1 + c_1 h)\|y_n - y(t_n)\| + dh^{q+1} \\
\le&(1 + c_1 h)^{n+1}\|y_0 - y(t_0)\| + \sum_{l=0}^{n}(1 + c_1 h)^l dh^{q+1} \\
\le&(1 + c_1 h)^{n+1}\|y_0 - y(t_0)\| + \frac{(1 + c_1 h)^{n+1} - 1}{c_1 h} dh^{q+1} \\
\le&e^{(n+1)c_1 h}\|y_0 - y(t_0)\| + \frac{e^{(n+1)c_1 h} - 1}{c_1} dh^q \\
\le&e^{c_1 T}\|y_0 - y(t_0)\| + dh^q \frac{e^{c_1 T - 1}}{c_1}.
\end{aligned}
$$

Taking $\|y_0 - y(t_0)\| = O(h^q)$ into consideration, we find that $\|y_{n+1} - y(t_{n+1})\| \le c_2 h^q$, where the constant c_2 doesn't depend on the stiffness of the problem. □

We conclude this section with constructing the 3-stage IMEX RKR method which is B-consistent of order $q = 2$. It follows from $Y_{n+1,1} = y_n$ that the first stage $q_1 = \infty$ and the second stage $q_2 = 0$ only. Furthermore, based on the conditions (4.7) and (4.11), we obtain the 2-order 3-stage IMEX RKR method with one parameter c_2 $(0 < c_2 < 1)$:

$$
b_1 = \frac{1}{2}, \quad b_2 = 0, \quad b_3 = \frac{1}{2}, \quad c_1 = 0, \quad c_3 = 1 \tag{4.16a}
$$

$$
a_{21} = \widetilde{a}_{21} = c_2, \quad a_{31} = \widetilde{a}_{31} = 1 - \frac{1}{2c_2}, \quad a_{32} = \widetilde{a}_{32} = \frac{1}{2c_2}, \tag{4.16b}
$$

$$
\gamma_{21} = -c_2, \quad \gamma_{31} = \frac{1}{2c_2} - 1, \quad \gamma_{32} = -\frac{1}{2c_2}, \quad \gamma_{11} = \gamma_{22} = \gamma_{33} = \frac{1}{2}. \tag{4.16c}
$$

We also observe that the coefficient $a_{ij} = \widetilde{a}_{ij}$ $(1 \le i, j \le 3)$ in the scheme (4.16), but the discretization scheme (4.16) is different from the linearly implicit Rosenbrock scheme presented in [27], where the Jacobian matrix

$$
\mathcal{J}_F(t_n) = \frac{\partial F}{\partial y}(t_n, y(t_n)) = \frac{\partial f}{\partial y}(t_n, y(t_n)) + \frac{\partial g}{\partial y}(t_n, y(t_n)).
$$

In order to reduce the overall computational efforts, we choose the Jacobian matrix $\mathcal{J}_f(t_n) = \frac{\partial f}{\partial y}(t_n, y(t_n))$. Additionally, Strehmel et al. [27] is only concerned with the B-convergence results of linearly implicit Rosenbrock methods when applied to stiff semi-linear systems and a special case of nonlinear singular perturbation systems.

5. Extension to a Class of Singular Perturbed Problems

In this section, the IMEX RKR method (2.4) are applied to a class of singular perturbed initial value problems, i.e. problems (2.1) satisfying the conditions $(c2)$, $(c3)$ and the condition

(c4) $f(t, y) = \tilde{f}(t, y)/\varepsilon$, $0 < \varepsilon \ll 1$, $\mu(\frac{\partial \tilde{f}}{\partial y}) \le -\mu_0 < 0$, and $\tilde{f}$ satisfies the *Lipschitz* condition.

The one-sided *Lipschitz* constant of the function $\tilde{f}(t,y)/\varepsilon$ is very large and reaches $O(\varepsilon^{-1})$, thus the classical convergence and B-convergence theories are not suitable for this problem. For this problems, Xiao et al. [34] presented two classes of the IMEX multistep methods by combining implicit one-leg methods with explicit linear multistep methods and explicit one-leg methods respectively, and the order conditions, efficient schemes and convergence results of these methods were obtained.

Applying the IMEX RKR methods (2.4) to the above problem yields

$$\begin{cases} y_{n+1} = y_n + h\sum\limits_{i=1}^{s} b_i k_i, \\ (I - h\gamma\mathcal{J})k_i = \frac{1}{\epsilon}\widetilde{f}(t_n + c_i h, Y_{n+1,i}) + g(t_n + c_i h, \widetilde{Y}_{n+1,i})) + h\mathcal{J}\sum\limits_{j=1}^{i-1}\gamma_{ij}k_j, \end{cases} \tag{5.1}$$

where

$$\mathcal{J} = \frac{1}{\epsilon}\mathcal{J}_{\widetilde{f}}(t_n, y(t_n)) + I_O(h), \qquad \mathcal{J}_{\widetilde{f}}(t_n, y(t_n)) = \frac{\partial\widetilde{f}}{\partial y}(t_n, y(t_n)). \tag{5.2}$$

Based on Lemma 4.1 and the related results in [27,32], for $0 < h \le h_0$, we have $\mu(\mathcal{J}) \le \mu_1 < 0$, and

$$\left\| \frac{h}{\epsilon}(I - \frac{h}{\epsilon}\gamma\mathcal{J})^{-1} \right\| \le D_1, \tag{5.3}$$

where $\gamma > 0$, and the constants h, D_1 are independent of γ, μ.

For the method (5.1), we can obtain the conclusion being similar to Theorems 4.2 with $f_l = \frac{1}{\epsilon}\widetilde{f}_l$, and the corresponding proof is similar. Firstly, we give the lemma presented in [27,32] to illustrate some properties, which will be used in the following.

Lemma 5.1 ([27, 32]). *Suppose that $\epsilon \le c_0 h, h \le h_1$. Then*

$$I + \gamma w = (I - h\gamma\mathcal{J})^{-1} = I_O(\frac{\epsilon}{h}), \tag{5.4a}$$

$$w^k = ((I - h\gamma\mathcal{J})^{-1}h\mathcal{J})^k = I_O(1), \tag{5.4b}$$

where the matrix $I_O(\frac{\epsilon}{h})$ has the same number of rows and columns as $\mathcal{J}$ and its entries are all $O(\frac{\epsilon}{h})$, c_0, h_1 don't depend on $\frac{1}{\epsilon}$.

Theorem 5.2. *Suppose that the computation is evaluated at $y(t_n)$, i.e. $y_n = y(t_n)$, and the simplifying assumption*

$$a_i^T c^{l-1} = \frac{1}{l}c_i^l, \quad \widetilde{a}_i^T c^{l-1} = \frac{1}{l}c_i^l, \qquad l = 1, \cdots, q_i + 1, \tag{5.5a}$$

$$a_i^T\beta^k((\beta + \gamma I)lc^{l-1} - c^l) = 0, \qquad k = 0, 1, \cdots, i-2, \quad l = 1, \cdots, q_i, \tag{5.5b}$$

$$\widetilde{a}_i^T\beta^k((\beta + \gamma I)lc^{l-1} - c^l) = 0, \qquad k = 0, 1, \cdots, i-2, \quad l = 1, \cdots, q_i, \tag{5.5c}$$

$$a_i^T\beta^k e_j = 0, \quad \widetilde{a}_i^T\beta^k e_j = 0, \qquad k = 1, \cdots, i-2, \quad \text{if} \;\; q_j < q_i - 1 \tag{5.5d}$$

holds. Then there exists $h_1 > 0$ such that for $h \le h_1$,

$$\| Y_{n+1,j} - y(t_n + c_j h) \| = O(h^{q_j+1}), \tag{5.6a}$$

$$\| \widetilde{Y}_{n+1,j} - y(t_n + c_j h) \| = O(h^{q_j+1}), \tag{5.6b}$$

where $j = 1, 2, \cdots, s+1$, h_1 doesn't depend on $\frac{1}{\epsilon}$.

Proof. According to the Taylor expansion

$$\begin{aligned}
&\frac{h}{\epsilon}\widetilde{f}_l + hg_l - hy'(t_n + c_l h)\\
=&\frac{h}{\epsilon}\int_0^1 \widetilde{f}_y(t_n + c_l h, y(t_n + c_l h) + \theta(Y_{n+1,l} - y(t_n + c_l h)))d\theta\triangle_{n+1,l}\\
&+ h\int_0^1 g_y(t_n + c_l h, y(t_n + c_l h) + \theta(\widetilde{Y}_{n+1,l} - y(t_n + c_l h)))d\theta\widetilde{\triangle}_{n+1,l}\\
=&h\Big(\frac{1}{\epsilon}\mathcal{J}_{\widetilde{f}}(t_n, y(t_n)) + I_O(h) + M_l\Big)\triangle_{n+1,l} + h\Big(g_y(t_n, y(t_n)) + I_O(h)\Big)\widetilde{\triangle}_{n+1,l},
\end{aligned}$$

where $M_l,\ g_y$ are bounded matrixes, we find

$$\begin{aligned}
&\frac{h}{\epsilon}\widetilde{f}_l + hg_l - hy'(t_n + c_l h) + h\mathcal{J}y(t_n + c_l h) - h\mathcal{J}Y_{n+1,l}\\
=&\frac{h}{\epsilon}\widetilde{f}_l + hg_l - hy'(t_n + c_l h) - h\mathcal{J}\triangle_{n+1,l}\\
=&h(M_l + I_O(h))\triangle_{n+1,l} + h(g_y(t_n, y(t_n)) + I_O(h))\widetilde{\triangle}_{n+1,l},
\end{aligned}$$

and

$$\triangle_{n+1,l} = Y_{n+1,l} - y(t_n + c_l h) = O(h^{q_l+1}), \qquad \widetilde{\triangle}_{n+1,l} = O(h^{q_l+1}).$$

Inserting it into (4.4) with $f_l = \frac{1}{\epsilon}\widetilde{f}_l$ leads to the local error estimates of the i-th stage

$$Y_{n+1,i} - y(t_n + c_i h) = \overline{A} + \overline{B}, \quad \widetilde{Y}_{n+1,i} - y(t_n + c_i h) = \widetilde{A} + \widetilde{B},$$

where $\overline{A}$, A_1, $\triangle_{n+1,l}$, $\widetilde{\triangle}_{n+1,l}$ are defined in (4.10), and

$$\begin{aligned}
\overline{B} &= \sum_{k=0}^{i-2}\sum_{j=1}^{i-1} a_i^T\beta^k e_j w^k (I + \gamma w) O(\frac{h}{\epsilon}) O(h^{q_j+2}),\\
\widetilde{B} &= \sum_{k=0}^{i-2}\sum_{j=1}^{i-1} \widetilde{a}_i^T\beta^k e_j w^k (I + \gamma w) O(\frac{h}{\epsilon}) O(h^{q_j+2}).
\end{aligned}$$

Similar to the proof of Theorems 4.3, based on (5.4) and Lemma 5.1, there exists integer $q_j > 0$ such that for $h \le h_1$, (5.6) holds. □

Corollary 5.3. *Suppose that the computation is evaluated at* $y(t_n)$, *i.e.* $y_n = y(t_n)$, *and*

$$b^T c^{l-1} = \frac{1}{l}, \qquad l = 1, \cdots, q+1, \tag{5.7a}$$

$$b^T\beta^k((\beta + \gamma I)lc^{l-1} - c^l) = 0, \qquad k = 0, \cdots, s-1, \quad l = 1, \cdots, q, \tag{5.7b}$$

$$b^T\beta^k e_j = 0, \quad b^T\widetilde{\beta}^k e_j = 0, \qquad k = 0, 1, \cdots, i-2, \quad \text{if } \ q_j < q-1 \tag{5.7c}$$

holds, then $\|y_n - y(t_{n+1})\| = O(h^{q+1})$.

Theorem 5.4. *Let* $a_i^T\beta^j = \widetilde{a}_i^T\widetilde{\beta}^j$, $i = 1, \cdots, s$, $j = 0, 1, \cdots, s-2$, *the stability function* $R_{0,s+1}(\xi)$ *be strongly A-acceptable, and* (5.7) *hold. Then there exists the constant* $h_2 > 0$ *such that for* $\epsilon \le c_0 h$, $h \le h_2$, $\| y_0 - y(t_0) \| = O(h^q)$, *we have the global error estimate*

$$\| y_n - y(t_n) \| = O(h^q), \tag{5.8}$$

where c_0 *and* h_2 *do not depend on* $\frac{1}{\epsilon}$.

Proof. Let $\widehat{y}_{n+1}$ be the numerical solution by the methods (2.4) with the initial value $y_n = y(t_n)$. Define the same symbols as (4.13) with $f_l = \frac{1}{\epsilon}\widetilde{f}_l$, then

$$\begin{aligned}
&\frac{1}{\epsilon}\widetilde{f}_l + g_l - \mathcal{J}Y_{n+1,l} - \widehat{f}_l - \overline{g}_l + \mathcal{J}\widehat{Y}_{n+1,l}\\
=&\frac{1}{\epsilon}\int_0^1 \widetilde{f}_y(t_n + c_l h, \widehat{Y}_{n+1,l} + \theta(Y_{n+1,l} - \widehat{Y}_{n+1,l}))d\theta(Y_{n+1,l} - \widehat{Y}_{n+1,l})\\
&\quad + \int_0^1 g_y(t_n + c_l h, \overline{Y}_{n+1,l} + \theta(\widehat{Y}_{n+1,l} - \overline{Y}_{n+1,l}))d\theta(\widehat{Y}_{n+1,l} - \overline{Y}_{n+1,l}) - \mathcal{J}(Y_{n+1,l} - \widehat{Y}_{n+1,l})\\
=&\Big(\frac{1}{\epsilon}\mathcal{J}_{\widetilde{f}}(t_n + c_l h) - \mathcal{J} + M_l\Big)(Y_{n+1,l} - \widehat{Y}_{n+1,l}) + \Big(g_y(t_n, y(t_n)) + I_O(h)\Big)(\widehat{Y}_{n+1,l} - \overline{Y}_{n+1,l})\\
=&Q_l\nabla_{n+1,l} + \widetilde{Q}_l\widetilde{\nabla}_{n+1,l},
\end{aligned}$$

where $Q_l = M_l + I_O(h)$, $\widetilde{Q}_l = g_y(t_n, y(t_n)) + I_O(h)$, and g_y, M_l are bounded matrixes. Thanks to $a_i^T\beta^k = \widetilde{a}_i^T\widetilde{\beta}^k$, we obtain

$$\begin{aligned}
\nabla_{n+1,i} &= Y_{n+1,i} - \widehat{Y}_{n+1,i}\\
&=R_{0,i}(w)(y_n - y(t_n)) + h\sum_{k=0}^{s-1}\sum_{l=1}^{s} b^T\beta^k e_l w^k(I + \gamma w)\Big(Q_l\nabla_{n+1,l} + \widetilde{Q}_l\widetilde{\nabla}_{n+1,l}\Big).
\end{aligned} \tag{5.9}$$

Denote

$$K_j = \sum_{k=0}^{s-1}\sum_{l=1}^{s} a_j^T\beta^k e_l w^k(I + \gamma w)Q_l, \qquad \widetilde{K}_j = \sum_{k=0}^{s-1}\sum_{l=1}^{s} a_j^T\beta^k e_l w^k(I + \gamma w)\widetilde{Q}_l, \tag{5.10}$$

$$R = diag(R_0^{(1)}(w), \cdots, R_0^{(s)}(w))^T, \tag{5.11}$$

$$\nabla = (\nabla_{n+1,1}, \cdots, \nabla_{n+1,s})^T, \qquad \widetilde{\nabla} = (\widetilde{\nabla}_{n+1,1}, \cdots, \widetilde{\nabla}_{n+1,s})^T, \tag{5.12}$$

then (5.9) can be rewritten into the vector form

$$\nabla = R((y_n - y(t_n)) \otimes e) + h(K\nabla + \widetilde{K}\widetilde{\nabla}).$$

Moreover,

$$\begin{aligned}
w^k(I + \gamma w)Q_l &= c_l w^{k+1} + w^k(I + \gamma w)\Big(M_l + I_O(h)\Big),\\
w^k(I + \gamma w)\widetilde{Q}_l &= c_l w^{k+1} + w^k(I + \gamma w)\Big(g_y(t_n, y(t_n)) + I_O(h)\Big).
\end{aligned}$$

It follows from (5.2) and (5.4) that $\| R_{0,i} \|$ bounded, and

$$w^k(I + \gamma w)Q_l = I_O(1), \quad w^k(I + \gamma w)\widetilde{Q}_l = I_O(1).$$

Furthermore,

$$\nabla = (I + I_O(h))R((y_n - y(t_n)) \otimes e) + O(h)\widetilde{\nabla}.$$

Similarly,

$$\widetilde{\nabla} = (I + I_O(h))\widetilde{R}((y_n - y(t_n)) \otimes e) + O(h)\nabla.$$

Thus we have

$$\|\nabla_{n+1,i}\| \le C_2\|y_n - y(t_n)\|, \quad \|\widetilde{\nabla}_{n+1,i}\| \le C_3\|y_n - y(t_n)\|,$$

where C_2, C_3 are constants. Taking the above formulas and (5.9) into consideration, it holds that

$$\begin{aligned}&y_{n+1}-\widehat{y}_{n+1}\\=&R_{0,s+1}(w)(y_n-y(t_n))+h\sum_{k=0}^{s-1}\sum_{l=1}^{s}b^T\beta^k e_l w^k(I+\gamma w)(Q_l\nabla_{n+1,l}+\widetilde{Q}_l\widetilde{\nabla}_{n+1,l}).\end{aligned}$$

Due to strong A-acceptability of $R_{0,s+1}(\xi)$,

$$\begin{aligned}&\|y_{n+1}-\widehat{y}_{n+1}\|\\\le&\|R_{0,s+1}(\omega)\|y_n-y(t_n)\|+hc_4\|y_n-y(t_n)\|\\\le&(1+c_4h)\|y_n-y(t_n)\|.\end{aligned}$$

It follows from Corollary 5.3 that $\|\widehat{y}_{n+1}-y(t_{n+1})\|\le dh^{q+1}$, where d doesn't depend on ϵ, thus

$$\begin{aligned}&\|y_{n+1}-y(t_{n+1})\|\\\le&\|y_{n+1}-\widehat{y}_{n+1}\|+\|\widehat{y}_{n+1}-y(t_{n+1})\|\\\le&(1+c_4h)\|y_n-y(t_n)\|+dh^{q+1}\\\le&(1+c_4h)^{n+1}\|y_0-y(t_0)\|+\sum_{l=0}^{n}(1+c_4h)^l dh^{q+1}\\\le&(1+c_4h)^{n+1}\|y_0-y(t_0)\|+\frac{(1+c_4h)^{n+1}-1}{c_4h}dh^{q+1}\\\le&e^{(n+1)c_4h}\|y_0-y(t_0)\|+\frac{e^{(n+1)c_4h}-1}{c_4}dh^q\\\le&e^{c_4T}\|y_0-y(t_0)\|+dh^q\frac{e^{c_4T-1}}{c_4}.\end{aligned}$$

If $\|y_0-y(t_0)\|=O(h^q)$, then $\|y_{n+1}-y(t_{n+1})\|=c_5h^q$, where the constant c_5 is independent of ϵ. □

6. Numerical Examples

The purpose of this section is to illustrate the computational efficiency and theoretical results of the developed method (4.16) and to compare the famous implicit 3-stage Radau IIA method of B-convergence order 2

$$\begin{array}{c|ccc}\frac{4-\sqrt{6}}{10}&\frac{88-7\sqrt{6}}{360}&\frac{296-169\sqrt{6}}{1800}&\frac{-2+3\sqrt{6}}{225}\\\frac{4+\sqrt{6}}{10}&\frac{296+169\sqrt{6}}{1800}&\frac{88+7\sqrt{6}}{360}&\frac{-2-3\sqrt{6}}{225}\\1&\frac{16-\sqrt{6}}{36}&\frac{16+\sqrt{6}}{36}&\frac{1}{9}\\\hline&\frac{16-\sqrt{6}}{36}&\frac{16+\sqrt{6}}{36}&\frac{1}{9}\end{array}\tag{6.1}$$

and the corresponding numerical results will be shown in Tables 6.1-6.10. Moreover, the error at the end point of the interval and the convergence order of the used methods are denoted by

$$err(h)=\|y(t_n)-y_h(t_n)\|,\quad order(h)=\log_2\frac{err(2h)}{err(h)},$$

Table 6.1: Errors and orders of the method (4.16) for (6.2) at $T = 1$.

	$c_2 = \frac{1}{4}$		$c_2 = \frac{1}{2}$	
h	$err(h)$	$order(h)$	$err(h)$	$order(h)$
h	6.3341e-04	–	6.3323e-04	–
$\frac{h}{2}$	1.5398e-04	2.0404	1.5396e-04	2.0402
$\frac{h}{4}$	3.4065e-05	2.1764	3.4062e-05	2.1763
$\frac{h}{8}$	4.0764e-06	3.0629	4.0762e-06	3.0629

Table 6.2: The CPU times of the methods (4.16) and (6.1) for (6.2) with the stepsize $\frac{h}{8}$.

Methods	(4.16) & $c_2 = \frac{1}{4}$	(4.16) & $c_2 = \frac{1}{2}$	(6.1)
CPU times	16.957832 s	16.493799 s	1324.961468 s

respectively, where $y(t_n)$, $y_h(t_n)$ are the exact solution and the numerical solution, respectively.

Example 6.1 ([34]). Consider the following reaction-diffusion problem

$$\begin{cases} \frac{\partial u}{\partial t} = d\frac{\partial^2 u}{\partial x^2} + ru(1-u) + w(t,x), & 0 \le t \le 1, \quad x \in \Omega = (0,\pi), \\ u(t,0) = u(t,\pi) = 0, & 0 \le t \le 1, \\ u(0,x) = \sin x, & x \in \Omega, \end{cases}$$

where $w(t,x) = (d-2-r)e^{-2t}\sin(x) + re^{-4t}\sin^2(x)$ is a given function such that the exact solution $u(t,x) = e^{-2t}\sin(x)$.

Let the space stepsize $h_x = \pi/M$ and M be a positive integer. Taking the equidistant nodes on Ω: $x_j = jh_x$ $(j = 1, \cdots, M-1)$ and using the central difference to discretize $\frac{\partial^2 u}{\partial x^2}$ yield

$$\frac{dy}{dt} = Ly + G(t,y), \tag{6.2}$$

where $y(t) = (u_1(t), u_2(t), \cdots, u_{M-1}(t))^T$, $u_i(t) \approx u(t,x_i)$,

$$L = \frac{d}{h_x^2}\begin{pmatrix} -2 & 1 & 0 & \cdots & 0 \\ 1 & -2 & 1 & \cdots & 0 \\ 0 & 1 & -2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & -2 \end{pmatrix}, \qquad G(t,y) = \begin{pmatrix} g_1(t,y) \\ g_2(t,y) \\ \vdots \\ g_{M-1}(t,y) \end{pmatrix},$$

and $g_i(t,y) = ru_i(t)(1-u_i(t)) + (d-2-r)e^{-2t}\sin(x_i) + re^{-4t}\sin^2(x_i)$. It is easy to show that

$$\mu(L) = -4M^2\sin^2(\frac{\pi}{2M}) < 0 \quad \text{and} \quad \|\frac{\partial g_{\tilde{h}}}{\partial y}\| = \max_i\{|r(1-2u_i(t))|\} \le |r|(1+2e),$$

which implies that the conditions (2.2) and (2.3) are satisfied.

Taking $d = 3$, $r = 1$, $M = 1000$ and the time stepsize $h = 0.02$. The errors $err(h)$ and the convergence orders $order(h)$ of the method (4.16) for (6.2) are presented in Table 6.1, and we can find that the orders of the method (4.16) are at least 2. This confirms the theoretical result given in Theorem 4.5.

From Table 6.2, we show that the IMEX RKR methods (4.16) take much less CPU time than the 3-stage Radau IIA method (6.1) for (6.2), thus the IMEX RKR methods can improve obviously the computation efficiency and avoid solving nonlinear algebraic equations.

Example 6.2 ([23]). Consider the initial value problem

$$\begin{cases} y'(t) = Ay + g(t, y(t)), & 0 \le t \le 5, \\ y_1(0) = 0, \quad y_2(0) = -1, \end{cases} \tag{6.3}$$

where $y(t) = (y_1(t), y_2(t))^T$,

$$A = \begin{pmatrix} -1001 & 999 \\ 999 & -1001 \end{pmatrix}, \qquad g(t, y(t)) = \begin{pmatrix} 2y_1y_2 \\ y_1^2 + y_2^2 \end{pmatrix}.$$

Table 6.3: Errors and orders of the method (4.16) for (6.3) at $T = 5$.

	$c_2 = \frac{1}{4}$		$c_2 = \frac{1}{2}$	
h	$err(h)$	$order(h)$	$err(h)$	$order(h)$
h	9.2861e-009	–	9.2834e-009	–
$\frac{h}{2}$	2.3201e-009	2.0009	2.3198e-009	2.0007
$\frac{h}{4}$	5.7984e-010	2.0005	5.7980e-010	2.0004
$\frac{h}{8}$	1.4494e-010	2.0002	1.4493e-010	2.0002

The exact solution is

$$y_1(t) = \frac{1000}{2001e^{2000t} - 1} - \frac{1}{3e^{2t} - 1}, \qquad y_2(t) = -\frac{1000}{2001e^{2000t} - 1} - \frac{1}{3e^{2t} - 1}.$$

It is easy to verify that $\mu(A) = -2$ and $\|\frac{\partial g}{\partial y}\| = 2(|y_1| + |y_2|) \le 4$, which implies that the conditions (2.2) and (2.3) are satisfied.

Let the stepsize $h = 0.01$. The errors $err(h)$ and the convergence orders $order(h)$ of the method (4.16) for (6.3) are presented in Table 6.3, and we can clearly observe that the orders of the method (4.16) for (6.3) are in accord with the theoretical result given in Theorem 4.5.

From Table 6.4, we show that the IMEX RKR methods (4.16) take much less CPU time than the 3-stage Radau IIA method (6.1) for (6.3), thus the IMEX RKR methods can improve obviously the computation efficiency and avoid solving nonlinear algebraic equations.

Example 6.3 ([15, 34]). Consider the famous Oregonator model

$$\begin{cases} y'(t) = f(t, y) + g(t, y), & 0 \le t \le 5, \\ y_1(0) = 1, \quad y_2(0) = 2, \quad y_3(0) = 3, \end{cases} \tag{6.4}$$

where $y(t) = (y_1(t), y_2(t), y_3(t))^T$,

$$f = \begin{pmatrix} 77.27(y_2 + y_1(1 - y_2)) \\ 0 \\ 0 \end{pmatrix}, \qquad g = \begin{pmatrix} -77.27 \times 8.375 \times 10^{-6} y_1^2 \\ \frac{1}{77.27}(y_3 - (1 + y_1)y_2) \\ 0.161(y_1 - y_3) \end{pmatrix}.$$

The reference exact solution of (6.4) is computed by using the classical fourth-order Runge-Kutta method with the small stepsize $h_1 = 10^{-6}$.

Based on the reference exact solution, we can obtain the eigenvalues of the matrix $\frac{1}{2}((\frac{\partial f}{\partial y})^T + \frac{\partial f}{\partial y})$ and show that

$$\mu(\frac{\partial f}{\partial y}) = 77.27 \frac{1 - y_2 + \sqrt{(1 - y_1)^2 + (1 - y_2)^2}}{2} \le \frac{77.27|1 - y_1|}{2},$$

Table 6.4: The CPU times of the methods (4.16) and (6.1) for (6.3) with the stepsize $\frac{h}{8}$.

Methods	(4.16) & $c_2 = \frac{1}{4}$	(4.16) & $c_2 = \frac{1}{2}$	(6.1)
CPU times	0.103207 s	0.107457 s	0.742482 s

and $\|\frac{\partial g}{\partial y}\| \leq 0.3$, which implies that the conditions (2.2) and (2.3) are satisfied.

Let $h = 0.01$. The errors $err(h)$ and the convergence orders $order(h)$ of the method (4.16) for (6.4) are presented in Table 6.5, and we show that the orders of the method (4.16) are at least 2. This confirms the theoretical result stated in Theorem 4.5.

Table 6.5: Errors and orders of the method (4.16) for (6.4) at $T = 5$.

	$c_2 = \frac{1}{4}$		$c_2 = \frac{1}{2}$	
h	$err(h)$	$order(h)$	$err(h)$	$order(h)$
h	7.3445e-08	–	6.1783e-08	–
$\frac{h}{2}$	1.3814e-08	2.4106	1.2017e-08	2.3622
$\frac{h}{4}$	2.7675e-09	2.3194	2.5165e-09	2.2555
$\frac{h}{8}$	5.9659e-10	2.2138	5.6336e-10	2.1593

Table 6.6: The CPU times of the methods (4.16) and (6.1) for (6.4) with the stepsize $\frac{h}{8}$.

Methods	(4.16) & $c_2 = \frac{1}{4}$	(4.16) & $c_2 = \frac{1}{2}$	(6.1)
CPU times	0.114110 s	0.116880 s	0.813169 s

From Table 6.6, we show that the IMEX RKR methods (4.16) take much less CPU time than the 3-stage Radau IIA method (6.1) for (6.4), thus the IMEX RKR methods can improve obviously the computation efficiency and avoid solving nonlinear algebraic equations.

Example 6.4. Consider the singular perturbed initial value problem

$$\begin{cases} \frac{dy}{dt} = y - \frac{1}{\varepsilon}y^3, \qquad 0 \leq t \leq 1, \\ y(0) = 2. \end{cases} \tag{6.5}$$

The reference exact solution of (6.5) is computed by using the MATLAB $RADAU5$ with the small stepsize $h_1 = 10^{-6}$.

For the initial interval [0,0.01], the numerical solutions are also calculated by the method (4.16) with the small stepsize $h_1 = 10^{-6}$. Let $h = 0.01$. The errors $err(h)$ at the end point of the interval and the convergence orders $order(h)$ of the method (4.16) for (6.5) are presented in Table 6.7, which show that the method (4.16) achieve the expected order of accuracy and that there exists no order reduction for the IMEX RKR schemes.

From Table 6.8, we show that the IMEX RKR methods (4.16) take much less CPU time than the 3-stage Radau IIA method (6.1) for (6.5), thus the IMEX RKR methods can improve obviously the computation efficiency and avoid solving nonlinear algebraic equations.

Example 6.5. Consider the singular perturbed initial value problem

$$\begin{cases} y_1'(t) = y_1 y_2, \\ y_2'(t) = \frac{1}{\varepsilon}(y_2 - y_2^3), \\ y_1(0) = 1, \quad y_2(0) = 2, \end{cases} \tag{6.6}$$

Table 6.7: Errors and orders of the method (4.16) for (6.5) at $T = 1$.

ε	h	$c_2 = \frac{1}{4}$		$c_2 = \frac{1}{2}$	
		$err(h)$	$order(h)$	$err(h)$	$order(h)$
$\varepsilon = 10^{-3}$	h	1.5891e-05	–	1.6527e-05	–
	$\frac{h}{2}$	4.0748e-06	1.9635	4.1310e-06	2.0003
	$\frac{h}{4}$	1.0322e-06	1.9810	1.0366e-06	1.9946
	$\frac{h}{8}$	2.5930e-07	1.9931	2.5962e-07	1.9974
$\varepsilon = 10^{-4}$	h	5.0808e-06	–	5.2872e-06	–
	$\frac{h}{2}$	1.3029e-06	1.9633	1.3212e-06	2.0006
	$\frac{h}{4}$	3.3015e-07	1.9806	3.3157e-07	1.9945
	$\frac{h}{8}$	8.2969e-08	1.9925	8.3073e-08	1.9969
$\varepsilon = 10^{-5}$	h	1.6085e-06	–	1.6740e-06	–
	$\frac{h}{2}$	4.1258e-07	1.9630	4.1839e-07	2.0004
	$\frac{h}{4}$	1.0463e-07	1.9794	1.0509e-07	1.9933
	$\frac{h}{8}$	2.6382e-08	1.9877	2.6419e-08	1.9920

Table 6.8: The CPU times of (4.16) and (6.1) for (6.5) with the stepsize $\frac{h}{8}$ and $\varepsilon = 10^{-5}$.

Methods	(4.16) & $c_2 = \frac{1}{4}$	(4.16) & $c_2 = \frac{1}{2}$	(6.1)
CPU times	0.013130 s	0.013318 s	0.062098 s

Table 6.9: Errors and orders of the method (4.16) for (6.6) at $T = 1$.

ε	h	$c_2 = \frac{1}{4}$		$c_2 = \frac{1}{2}$	
		$err(h)$	$order(h)$	$err(h)$	$order(h)$
$\varepsilon = 10^{-3}$	h	2.2096e-05	–	2.2095e-05	–
	$\frac{h}{2}$	5.5642e-06	1.9895	5.5642e-06	1.9895
	$\frac{h}{4}$	1.3948e-06	1.9961	1.3948e-06	1.9961
	$\frac{h}{8}$	3.4791e-07	2.0033	3 .4790e-07	2.0033
$\varepsilon = 10^{-4}$	h	2.2073e-05	–	2.2073e-05	–
	$\frac{h}{2}$	5.5459e-06	1.9928	5.5457e-06	1.9929
	$\frac{h}{4}$	1.3776e-06	2.0093	1.3774e-06	2.0094
	$\frac{h}{8}$	3.3094e-07	2.0575	3.3074e-07	2.0582
$\varepsilon = 10^{-5}$	h	2.1957e-05	–	2.1937e-05	–
	$\frac{h}{2}$	5.4299e-06	2.0157	5.4103e-06	2.0196
	$\frac{h}{4}$	1.2617e-06	2.1055	1.2421e-06	2.1229
	$\frac{h}{8}$	2.1510e-07	2.5523	1.9547e-07	2.6678

where $0 \le t \le 1$, $y(t) = (y_1(t), y_2(t))^T$, the second component is stiff for small values of the parameter ε, and the first component is non-stiff. This problem can be abbreviated as

$$f(t,y) = \frac{1}{\varepsilon}\begin{pmatrix} 0 \\ y_2 - y_2^3 \end{pmatrix}, \qquad g(t,y) = \begin{pmatrix} y_1 y_2 \\ 0 \end{pmatrix}.$$

The reference exact solution of (6.6) is computed by using the MATLAB *RADAU*5 with the small stepsize $h_1 = 10^{-6}$.

For the initial interval [0,0.01], the numerical solutions are also calculated by the method (4.16) with the small stepsize $h_1 = 10^{-6}$. Let $h = 0.01$. The errors $err(h)$ at the end point of the interval and the convergence orders $order(h)$ of the method (4.16) for (6.6) are presented in Table 6.9. We can observe again that the method (4.16) achieve approximately the expected

Table 6.10: The CPU times of (4.16) and (6.1) for (6.6) with the stepsize $\frac{h}{8}$ and $\varepsilon = 10^{-5}$.

Methods	(4.16) & $c_2 = \frac{1}{4}$	(4.16) & $c_2 = \frac{1}{2}$	(6.1)
CPU times	0.022696 s	0.022267 s	0.148376 s

order of convergence, and that there is no order reduction for some range of stepsizes.

From Table 6.10, we show that the IMEX RKR methods (4.16) take much less CPU time than the 3-stage Radau IIA method (6.1) for (6.6), thus the IMEX RKR methods can improve obviously the computation efficiency and avoid solving nonlinear algebraic equations.

Acknowledgments. The authors wish to thank the anonymous referees for their valuable comments and suggestions. The work is supported by the National Natural Science Foundation of China (Grant Nos. 11671343, 11701110), the Foundation for the Key Laboratory of Computational Physics, China (No. 6142A05180103) as well as the Scientific Research Fund of Science and Technology Department of Hunan Province in China (Grant No. 2018WK4006).

References

[1] U.M. Ascher, S.J. Ruuth and R.J. Spiteri, Implicit-explicit Runge-Kutta methods for time dependent partial differential equations, *Appl. Numer. Math.*, **25** (1997), 151–167.

[2] F. Bassi, L. Botti, A. Colombo, A. Ghidoni and F. Massa, Linearly implicit Rosenbrock-type Runge-Kutta schemes applied to the discontinuous Galerkin solution of compressible and incompressible unsteady flows, *Comput. Fluids*, **118** (2015), 305–320.

[3] S. Boscarino, Error analysis of IMEX Runge-Kutta methods derived from differential-algebraic systems, *SIAM J. Numer. Anal.*, **45** (2007), 1600–1621.

[4] S. Boscarino, L. Pareschi and G. Russo, Implicit-explicit Runge-Kutta schemes for hyperbolic systems and kinetic equations in the diffusion limit, *SIAM J. Sci. Comput.*, **35** (2013), A22–A51.

[5] S. Boscarino, P.G. Lefloch and G. Russo, High-order asymptotic-preserving methods for fully nonlinear relaxation problems, *SIAM J. Sci. Comput.*, **36** (2014), A377–A395.

[6] S. Boscarino and L. Pareschi, On the asymptotic properties of IMEX Runge-Kutta schemes for hyperbolic balance laws, *J. Comput. Appl. Math.*, **316** (2017), 60–73.

[7] S. Boscarino, J. Qiu and G. Russo, Implicit-explicit integral deferred correction methods for stiff problems, *SIAM J. Sci. Comput.*, **40** (2018), A787–A816.

[8] J.H. Chaudhry, J.B. Collins and J.N. Shadid, A posteriori error estimation for multi-stage Runge-Kutta IMEX schemes, *Appl. Numer. Math.*, **117** (2017), 36–49.

[9] S. Conde, S. Gottlieb, Z.J. Grant and J.N. Shadid, Implicit and implicit-explicit strong stability preserving Runge-Kutta methods with high linear order, *J. Sci. Comput.*, **73** (2017), 667–690.

[10] M.S. Espedal and K.H. Karlsen, Numerical solution of reservoir flow models based on large time step operator splitting algorithms, In Filtration in Porous Media and Industrial Applications, *Lect. Notes Math.*, **1734** (2000), 9–77.

[11] D.J. Gardner, J.E. Guerra, F.P. Hamon, D.R. Reynolds, P.A. Ullrich and C.S. Woodward, Implicit-explicit (IMEX) Runge-Kutta methods for non-hydrostatic atmospheric models, *Geosci. Model Dev.*, **11** (2018), 1497–1515.

[12] S. González-Pinto, E. Hairer, D. Hernández-Abreu and S. Pérez-Rodríguez, AMF-type W-methods for parabolic problems with mixed derivatives, *SIAM J. Sci. Comput.*, **40** (2018), A2905–A2929.

[13] E. Hairer, G. Bader and C.H. Lubich, On the stability of semi-implicit methods for ordinary differential equations, *BIT Numer. Math.*, **22** (1982), 211–232.

[14] E. Hairer, S. P. Nørsett and G. Wanner, Solving Ordinary Differential Equations I. Nonstiff Problems, Springer, Berlin 1993.

[15] E. Hairer and G. Wanner, Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems, Springer, Berlin, 1996.

[16] E. Hairer, C.H. Lubich and M. Roche, Error of Rosenbrock methods for stiff problems studied via differential algebraic equations, *BIT Numer. Math.*, **29** (1989), 77–90.

[17] I. Higueras, N. Happenhofer, O. Koch and F. Kupka, Optimized strong stability preserving IMEX Runge-Kutta methods, *J. Comput. Appl. Math.*, **272** (2014), 116–140.

[18] I. Higueras and T. Roldán, Construction of additive semi-implicit Runge-Kutta methods with low-storage requirements, *J. Sci. Comput.*, **67** (2016), 1019–1042.

[19] M. Hochbruck, A. Ostermann and J. Schweitzer, Exponential Rosenbrock-type methods, *SIAM J. Numer. Anal.*, **47** (2009), 786–803.

[20] G. Izzo and Z. Jackiewicz, Highly stable implicit-explicit Runge-Kutta methods, *Appl. Numer. Math.*, **113** (2017), 71–92.

[21] A. Kanevsky, M.H. Carpenter, D. Gottlieb and J.S. Hesthaven, Application of implicit-explicit high order Runge-Kutta methods to discontinuous-Galerkin schemes, *J. Comput. Phys.*, **225** (2007), 1753–1781.

[22] O. Knoth and R. Wolke, Implicit-explicit Runge-Kutta methods for computing atmospheric reactive flows, *Appl. Numer. Math.*, **28** (1998), 327–341.

[23] S. Li, Theory of Computational Methods for Stiff Differential Equation, Hunan Science and Technology Press, Changsha, 1997 (in Chinese).

[24] J.L. Liu and A.G. Xiao, Convergence results of two-step W-methods for two-parameter singular perturbation problems, *Appl. Math. Comput.*, **189** (2007), 669–681.

[25] H.N. Najm, P.S. Wyckoff and O.M. Knio, A semi-implicit numerical scheme for reacting flow I. Stiff chemistry, *J. Comput. Phys.*, **143** (1998), 381-402.

[26] J. Schütz and K. Kaiser, A new stable splitting for singularly perturbed ODEs, *Appl. Numer. Math.*, **107** (2016), 18–33.

[27] K. Strehmel, R. Weiner and M. Büttner, Order results for Rosenbrock type methods on classes of stiff equations, *Numer. Math.*, **59**(1991), 723–737.

[28] P. Ullrich and C. Jablonowski, Operator-split Runge-Kutta-Rosenbrock methods for nonhydrostatic atmospheric models, *Mon. Weather Rev.*, **140** (2012), 1257–1284.

[29] J.G. Verwer, W.H. Hundsdorfer and J.G. Bloom, Numerical time integration for air poliution models, *Surveys Math. Indust*, **10** (2002), 107–174.

[30] J.G. Verwer and B.P. Sommeijer, An implicit-explicit Runge-Kutta-Chebyshev scheme for diffusion-reaction equations, *SIAM J. Sci. Comput.*, **25** (2004), 1824–1835.

[31] C.J. Vogl, A. Steyer, D.R. Reynolds, P.A. Ullrich and C.S. Woodward, Evaluation of implicit-explicit additive Runge-Kutta integrators for the HOMME-NH dynamical core, (2019). https://arXiv:1904.10115v1[math.NA].

[32] Z. Wen and A.G. Xiao, Quantitative error analysis of Rosenbrock methods for multiply-stiff singular perturbation problems, *Mathematica Numerica Sinica*, **28** (2006), 419–432.

[33] A.G. Xiao, Convergence results of Runge-Kutta methods for multiply-stiff singular perturbation problems, *J. Comput. Math.*, **20** (2002), 325–336.

[34] A.G. Xiao, G.G. Zhang and X. Yi, Two classes of implicit-explicit multistep methods for nonlinear stiff initial-value problems, *Appl. Math. Comput.*, **247** (2014), 47–60.

[35] C. Yue, A.G. Xiao and H. Liu, Nonlinear stability and B-convergence of additive Runge-Kutta methods for nonlinear stiff problems, *Adv. Appl. Math. Mech.*, **7** (2015), 472–495.

[36] E. Zharovsky, A. Sandu and H. Zhang, A class of implicit-explicit two-step Runge-Kutta methods, *SIAM J. Numer. Anal.*, **53** (2015), 321–341.

[37] X. Zhong, Additive semi-implicit Runge-Kutta methods for computing high-speed nonequilibrium reactive flows, *J. Comput. Phys.*, **128** (1996), 19–31.