Pergamon

Computers Math. Applic. Vol. 28, No. 10–12, pp. 17–31, 1994

0898-1221/94 $7.00 + 0.00

0898-1221(94)00183-9

Preconditioning in Parallel Runge-Kutta Methods for Stiff Initial Value Problems

P. J. VAN DER HOUWEN AND B. P. SOMMEIJER
Centre for Mathematics and Computer Science
P.O. Box 94079, 1090 GB Amsterdam, The Netherlands

Abstract—From a theoretical point of view, Runge-Kutta methods of collocation type belong to the most attractive step-by-step methods for integrating stiff problems. These methods combine excellent stability features with the property of superconvergence at the step points. Like the initial-value problem itself, they only need the given initial value without requiring additional starting values, and therefore, are a natural discretization of the initial-value problem. On the other hand, from a practical point of view, these methods have the drawback of requiring in each step the solution of a system of equations of dimension sd, s and d being the number of stages and the dimension of the initial-value problem, respectively. In contrast, linear multistep methods, the main competitor of Runge-Kutta methods, require the solution of systems of dimension d. However, parallel computers have changed the scene and have motivated us to design parallel iteration methods for solving the implicit systems in such a way that the resulting methods become efficient step-by-step methods for integrating stiff initial-value problems.

Keywords—Numerical analysis, Runge-Kutta methods, Stiff problems, Parallelism, Preconditioning.

1. INTRODUCTION

From a theoretical point of view, Runge-Kutta methods of collocation type belong to the most attractive step-by-step methods for integrating the stiff initial-value problem (IVP)

$$\mathbf{y}'(t) = \mathbf{f}(\mathbf{y}(t)), \quad \mathbf{y}(t_0) = \mathbf{y}_0, \qquad \mathbf{y}: \mathbb{R} \to \mathbb{R}^d. \tag{1.1}$$

However, from a practical point of view, these methods have the drawback of requiring in each step the solution of a system of equations of dimension sd, s and d being the number of stages and the dimension of the initial-value problem, respectively. In contrast, linear multistep methods, the main competitor, require the solution of systems of dimension d. This has prevented Runge-Kutta methods from becoming widely-used integration methods for stiff problems.

However, the introduction of parallel computers has changed the scene. In [1] and [2], it has already been shown that solving the implicit Runge-Kutta relations by a suitable parallel iteration process leads to integration methods that are more efficient and much more robust than the best sequential methods such as methods based on the backward differentiation formulas (BDFs). Iterative processes designed for parallel computers have been discussed by several authors. We mention the papers of Bellen [3], Bellen-Vermiglio-Zennaro [4], Jackson-Nørsett [5], Jackson-Kværnø-Nørsett [6], and Burrage [7].

The aim of this paper is to demonstrate that introducing preconditioning into the iteration method results in a further increase of the efficiency.

2. PARALLEL ITERATION METHODS

We shall study parallel iterative methods for solving the stage vector equation in the s-stage Runge-Kutta method

$$\mathbf{Y} = \mathbf{e} \otimes \mathbf{y}_n + h(A \otimes I_d)\mathbf{F}(\mathbf{Y}), \quad \mathbf{y}_{n+1} = \mathbf{y}_n + h\left(\mathbf{b}^\top \otimes I_d\right)\mathbf{F}(\mathbf{Y}). \tag{2.1a}$$

Here, $\mathbf{Y}$ is the sd-dimensional stage vector with s vector components $\mathbf{Y}_i$ of dimensional d, $\mathbf{F}(\mathbf{Y})$ is the sd-dimensional vector $(\mathbf{f}(\mathbf{Y}_i))$, $i = 1, 2, \ldots, s$, $\mathbf{b}$ and $\mathbf{e}$ are s-dimensional vectors, A is an s-by-s matrix, I_d is the d-by-d identity matrix, and $\otimes$ denotes the Kronecker product. The vector $\mathbf{e}$ has unit entries, and $\mathbf{b}$ and A contain the Runge-Kutta parameters. Since we are aiming at stiff IVPs, we assume that (2.1a) represents a stiffly accurate method, that is, $\mathbf{b}^\top = \mathbf{e}_s^\top A$, $\mathbf{e}_s$ denoting the s^{th} unit vector. As a consequence, the step point formula simplifies to

$$\mathbf{y}_{n+1} = \left(\mathbf{e}_s^\top \otimes I_d\right)\mathbf{Y}. \tag{2.1b}$$

The iterative methods studied in the present paper fit into the class

$$\begin{aligned} \mathbf{Y}^{(j+1)} - h(D \otimes I_d)\mathbf{F}\left(\mathbf{Y}^{(j+1)}\right) &= \mathbf{Y}^{(j)} - h(D \otimes I_d)\mathbf{F}\left(\mathbf{Y}^{(j)}\right) \\ &\quad - P_j \mathbf{R}_n\left(h, \mathbf{Y}^{(j)}\right), \qquad j = 0, \ldots, m-1, \\ \mathbf{R}_n(h, \mathbf{Y}) &:= \mathbf{Y} - \mathbf{e} \otimes \mathbf{y}_n - h(A \otimes I_d)\mathbf{F}(\mathbf{Y}), \end{aligned} \tag{2.2a}$$

where $\mathbf{Y}^{(0)}$ is a given initial iterate, D is a diagonal s-by-s matrix with fixed, positive diagonal entries, P_j is an sd-by-sd matrix whose entries may depend on the stepsize h and the Jacobian matrix $J_n = \frac{\partial \mathbf{f}(t_n, \mathbf{y}_n)}{\partial \mathbf{y}}$. The matrix P_j may be considered as a preconditioning matrix for the residual function $\mathbf{R}_n$. It will be assumed that P_j is bounded with respect to h and J_n. Evidently, if (2.2a) converges, then it converges to the stage vector $\mathbf{Y}$. Since D is diagonal, the s stage vector components of $\mathbf{Y}^{(j+1)}$ can be solved in parallel from the equation (2.2a) provided that at least s processors are available. Recursion (2.2a) will be called the *outer* iteration, and the iteration method used for solving $\mathbf{Y}^{(j+1)}$ from (2.2a) is called the *inner* iteration.

Assuming that a Newton-type iteration is used as inner iteration method, we are faced with linear systems whose matrix of coefficients $I_{sd} - h(D \otimes J_n)$ is block diagonal, that is, each processor has to solve linear systems with d-by-d coefficient matrix $I_d - h\delta_i J_n$, where δ_i denotes the i^{th} diagonal entry of D.

After each iteration, we define the step point values

$$\begin{aligned} \mathbf{y}^{(j+1)} &= \left(\mathbf{e}_s^\top \otimes I_d\right)\mathbf{Y}^{(j)}, \qquad j = 0, 1, \ldots, m-1; \\ \mathbf{y}_{n+1} &= \mathbf{y}^{(m)}, \end{aligned} \tag{2.2b}$$

where the step value $\mathbf{y}_{n+1} = \mathbf{y}^{(m)}$ denotes the accepted approximation to the corrector solution at t_{n+1}.

For $P_j = I_{sd}$, we obtain the PDIRK method (Parallel Diagonally Implicit Runge-Kutta method) proposed in [1,2]. In these papers, the matrix D was either used to achieve A-stability or L-stability for a given value of m, or for 'damping at infinity,' that is, the damping of components in the iteration error corresponding to 'infinite' eigenvalues of the Jacobian was optimized by minimizing the spectral radius of the iteration matrix at infinity. Since the latter technique turned out to be superior, the matrix D will again be used for 'damping at infinity,' whereas the matrices P_j will be employed for damping of error components corresponding to (complex) eigenvalues of the Jacobian matrix lying in the neighbourhood of the origin (damping of nonstiff error components).

In order to analyse the convergence of (2.2), we define the stage vector iteration error

$$\varepsilon^{(j)} := \mathbf{Y}^{(j)} - \mathbf{Y},$$

and we write (2.2a) in the form

$$\begin{aligned}\varepsilon^{(j+1)} - h\,(D \otimes I_d)\left[\mathbf{F}\left(\mathbf{Y}^{(j+1)}\right) - \mathbf{F}\left(\mathbf{Y}\right)\right] &= \left[I_{sd} - P_j\right]\varepsilon^{(j)} \\ &- h\,(D \otimes I_d)\left[\mathbf{F}\left(\mathbf{Y}^{(j)}\right) - \mathbf{F}(\mathbf{Y})\right] + hP_j\,(A \otimes I_d)\left[\mathbf{F}\left(\mathbf{Y}^{(j)}\right) - \mathbf{F}(\mathbf{Y})\right]. \quad (2.2\text{a}')\end{aligned}$$

For sufficiently smooth righthand side functions $\mathbf{f}$, we have

$$\mathbf{F}(\mathbf{U} + \delta) - \mathbf{F}(\mathbf{U}) = J(\mathbf{U})\delta + O\left(\delta^2\right),$$

where $J(\mathbf{U})$ is an sd-by-sd block-diagonal matrix whose diagonal blocks consist of the Jacobian matrices $\frac{\partial \mathbf{f}(\mathbf{U}_i)}{\partial \mathbf{y}}$, $\mathbf{U}_i$ being the components of $\mathbf{U}$. On substitution into (2.2a′) and ignoring second order terms of $\varepsilon^{(j)}$, we straightforwardly derive the linear error recursion

$$\left[I_{sd} - h\,(D \otimes I_d)\,J(\mathbf{Y})\right]\varepsilon^{(j+1)} = \left[I_{sd} - h\,(D \otimes I_d)\,J(\mathbf{Y}) - P_j + hP_j\,(A \otimes I_d)\,J(\mathbf{Y})\right]\varepsilon^{(j)}, \quad (2.2\text{a}'')$$

which can be written in the form

$$G\varepsilon^{(j+1)} = (G - P_jC)\,\varepsilon^{(j)}, \quad C := I_{sd} - h\,(A \otimes I_d)\,J(\mathbf{Y}), \quad G := I_{sd} - h\,(D \otimes I_d)\,J(\mathbf{Y}). \quad (2.3)$$

Hence,

$$\varepsilon^{(m)} = H_m(h)\varepsilon^{(0)}, \quad H_m(h) := \prod_{j=m-1}^{0} Z_j(h), \quad Z_j(h) := G^{-1}\left(G - P_jC\right), \qquad m \geq 1. \quad (2.4)$$

Anticipating that for $h \to 0$ the matrix H_m can be written in the form

$$H_m(h) = h^{\theta m}\left[K_m \otimes L_m + \Delta H_m\right], \qquad \theta \geq 0, \quad (2.5)$$

where the s-by-s matrix K_m is determined by the corrector matrix A and the d-by-d matrix L_m by $J(\mathbf{Y})$, we find the iteration error

$$\varepsilon^{(m)} = h^{\theta m}\left[K_m \otimes L_m + \Delta H_m\right]\varepsilon^{(0)}. \quad (2.6)$$

Denoting the (exact) corrector solution by $\mathbf{u}_{n+1} := \left(\mathbf{e}_s^{\mathrm{T}} \otimes I_d\right)\mathbf{Y}$, we find at the step points

$$\mathbf{y}_{n+1} - \mathbf{u}_{n+1} = \left(\mathbf{e}_s^{\mathrm{T}} \otimes I_d\right)H_m(h)\varepsilon^{(0)} = h^{\theta m}\left[\mathbf{e}_s^{\mathrm{T}}K_m \otimes L_m + \left(\mathbf{e}_s^{\mathrm{T}} \otimes I_d\right)\Delta H_m\right]\varepsilon^{(0)}. \quad (2.7)$$

We now assume that the predictor formula is only based on stage values from the preceding step, i.e.,

$$\mathbf{Y}^{(0)} - h\,(D^* \otimes I_d)\,\mathbf{F}\left(\mathbf{Y}^{(0)}\right) = (E \otimes I_d)\,\mathbf{X}, \quad (2.8)$$

where $\mathbf{X}$ is the stage vector computed in the previous step. We distinguish three types of predictors:

BDF predictor	$D^* = D$ and E determined by backward differentiation formulas
EXP predictor	$D^* = O$ and E determined by extrapolation formulas
LSV predictor	$D^* = O$ and $E = \mathbf{e}\mathbf{e}_s^{\mathrm{T}}$ (last-step-value predictor $\mathbf{Y}^{(0)} = \mathbf{e} \otimes \mathbf{y}_n$).

THEOREM 2.1. *Let the error amplification matrix H_m be written in the form (2.5), let the stage order of the corrector (2.1) be r, and define the vectors*

$$\begin{aligned} \mathbf{c} &:= A\mathbf{e} \\ \mathbf{v}_0 &:= E\mathbf{e} - \mathbf{e} \\ \mathbf{v}_j &:= \frac{1}{j!}\left(E(\mathbf{c}-\mathbf{e})^j + jD^*\mathbf{c}^{j-1} - \mathbf{c}^j\right), \qquad 1 \le j \le r, \\ \mathbf{v}_j &:= \frac{1}{j!}\left(E(\mathbf{c}-\mathbf{e})^j - j\left(A - D^*\right)\mathbf{c}^{j-1}\right), \qquad j > r. \end{aligned} \tag{2.9}$$

If the matrices D^* and E are such that $\mathbf{v}_j = \mathbf{0}$ for $j = 0, \dots, q$ with $q \le r-1$, then the predictor is of order q and the iteration error is given by

$$\varepsilon^{(m)} = h^{\theta m+q+1}\left[\mathbf{C}_m \otimes L_m \mathbf{y}^{(q+1)}(t_n) + O(\Delta H_m) + O(h)\right], \tag{2.10}$$

where the principal iteration error vector $\mathbf{C}_m$ is given by $\mathbf{C}_m := K_m \mathbf{v}_{q+1}$ with $K_0 := I_s$.

PROOF. Let $\mathbf{y}(t)$ denote the locally exact solution at the point t_n, and let us impose the localizing assumption, that is, we assume that the components of $\mathbf{X}$ are on $\mathbf{y}(t)$. Suppose that

$$\varepsilon^{(0)} = \mathbf{Y}^{(0)} - \mathbf{Y} = O\left(h^{q+1}\right), \qquad q \le r-1.$$

Then

$$\begin{aligned} \mathbf{Y}^{(0)} &= (E \otimes I_d)\mathbf{X} + h\left(D^* \otimes I_d\right)\mathbf{F}\left(\mathbf{Y}^{(0)}\right) \\ &= (E \otimes I_d)\mathbf{y}(t_{n-1}\mathbf{e} + h\mathbf{c}) + h\left(D^* \otimes I_d\right)\mathbf{F}\left(\mathbf{y}(t_n\mathbf{e} + h\mathbf{c}) + O\left(h^{q+1}\right)\right) \\ &= (E \otimes I_d)\mathbf{y}(t_{n-1}\mathbf{e} + h\mathbf{c}) + h\left(D^* \otimes I_d\right)\mathbf{y}'(t_n\mathbf{e} + h\mathbf{c}) + O\left(h^{q+2}\right), \\ \mathbf{Y} &= \mathbf{e} \otimes \mathbf{y}(t_n) + h(A \otimes I_d)\mathbf{F}(\mathbf{Y}) \\ &= \mathbf{e} \otimes \mathbf{y}(t_n) + h(A \otimes I_d)\mathbf{F}\left(\mathbf{y}(t_n\mathbf{e} + h\mathbf{c}) + O\left(h^{r+1}\right)\right) \\ &= \mathbf{e} \otimes \mathbf{y}(t_n) + h(A \otimes I_d)\mathbf{y}'(t_n\mathbf{e} + h\mathbf{c}) + O\left(h^{r+2}\right), \end{aligned}$$

where $\mathbf{y}(t_n\mathbf{e} + h\mathbf{c})$ is defined by its components $\mathbf{y}(t_n + hc_i)$, $i = 1, 2, \dots, s$ (componentwise notation). Taylor expansion yields

$$\begin{aligned} \mathbf{Y}^{(0)} - \mathbf{Y} = {} & (E \otimes I_d)\Big(\mathbf{e} \otimes \mathbf{y}(t_n) + h(\mathbf{c}-\mathbf{e}) \otimes \mathbf{y}'(t_n) + \frac{1}{2!}h^2(\mathbf{c}-\mathbf{e})^2 \otimes \mathbf{y}''(t_n) \\ & + \frac{1}{3!}h^3(\mathbf{c}-\mathbf{e})^3 \otimes \mathbf{y}'''(t_n) + \cdots\Big) - \mathbf{e} \otimes \mathbf{y}(t_n) \\ & + \left(\left(D^* - A\right) \otimes I_d\right)\Big(h\mathbf{e} \otimes \mathbf{y}'(t_n) + h^2\mathbf{c} \otimes \mathbf{y}''(t_n) + \frac{1}{2!}h^3\mathbf{c}^2 \otimes \mathbf{y}'''(t_n) \\ & + \frac{1}{3!}h^4\mathbf{c}^3 \otimes \mathbf{y}''''(t_n) + \cdots\Big) + O\left(h^{q+2}\right). \end{aligned}$$

Since the corrector satisfies the simplifying condition $C(r)$, i.e., $jA\mathbf{c}^{j-1} = \mathbf{c}^j$, $1 \le j \le r$, we can eliminate the matrix A from the Taylor coefficients up to order r. Finally, by introducing the vectors $\mathbf{v}_j$, the predictor error is given by

$$\varepsilon^{(0)} = \mathbf{Y}^{(0)} - \mathbf{Y} = \sum_{j=0} \mathbf{v}_j h^j \otimes \mathbf{y}^{(j)}(t_n) + O\left(h^{q+2}\right).$$

The proof is completed by substitution of this expression into (2.6). ∎

Although we are primarily interested in the iteration error at the step points, the accuracy of the stage vector $\mathbf{Y}^{(m)}$ itself plays a role in the predictor formula (2.8) for the next step (unless the LSV predictor is used). Therefore, all components of the principal iteration error vector $\mathbf{C}_m$ should be considered and not only its last component.

3. PRECONDITIONING

First, we show that there exists a two-parameter family of preconditioners by which in each iteration the iteration error can be reduced by a factor $O(h^2)$ as $h \to 0$. The parameters occurring in the preconditioners can be used for improving the accuracy of specific solution components. In the case of *linear* or weakly nonlinear IVPs, these parameters can effectively be employed by fitting them to the points in the spectrum of the Jacobian matrix of the IVP that correspond to the solution components we want to approximate with increased accuracy. The family of preconditioners derived here contains the preconditioners constructed in [8] and [9] as special cases.

3.1. The Iteration Error

The following theorem provides the explicit form of our preconditioners.

THEOREM 3.1. *Let S_{2m} be the polynomial of degree $2m$ defined by*

$$S_{2m}(x) = \left(\pi_0 - \sigma_0 x + x^2\right)\left(\pi_1 - \sigma_1 x + x^2\right) \cdot \cdots \cdot \left(\pi_{m-1} - \sigma_{m-1} x + x^2\right), \tag{3.1}$$

where σ_j and π_j are real coefficients, and let the matrices $P_j, j = 0, 1, 2, \ldots, m-1$, be defined by the expressions

$$\begin{aligned} P_j &= \left(I_{sd} - hD \otimes J_n\right)^{-1}\left(I_{sd} - \pi_j h^2 D^2 W_j(h) \otimes I_d + hD^2\left(W_j(h) - I_s\right)A^{-1} \otimes J_n\right), \\ J_n &:= \frac{\partial \mathbf{f}(\mathbf{y}_n)}{\partial \mathbf{y}}, \\ W_j(h) &= \left(I_s - 2D^{-1}A + D^{-2}A^2\right)\left(I_s - \sigma_j hA + \pi_j h^2 A^2\right)^{-1}. \end{aligned} \tag{3.2}$$

Then, for small h, the error amplification matrices Z_j and H_m are given by

$$\begin{aligned} Z_j(h) &= h^2\left(A^2 - 2DA + D^2\right) \otimes \left(\pi_j I_d - \sigma_j J_n + J_n^2\right) - h^2 (A - D) \otimes I_d \Delta J_n + O\left(h^3\right), \\ H_m(h) &= h^{2m}\left(A^2 - 2DA + D^2\right)^m \otimes S_{2m}\left(J_n\right) + O\left(h^{2m}\Delta J_n\right) + O\left(h^{2m+1}\right), \end{aligned} \tag{3.3}$$

where ΔJ_n vanishes if J_n does not depend on $\mathbf{y}_n$.

PROOF. The line of proof is analogous to that given in [9]. It starts with writing the preconditioner in the form

$$P_j = \left(I_{sd} - hD \otimes J_n\right)^{-1}\left(I_{sd} + M_j \otimes I_d + N_j \otimes J_n\right), \quad J_n := \frac{\partial \mathbf{f}(\mathbf{y}_n)}{\partial \mathbf{y}}, \tag{3.2'}$$

where M_j and N_j are matrices to be determined. Next, the matrices C and G defined in (2.3) are written as

$$\begin{aligned} C &= I_{sd} - h\left(A \otimes I_d\right)\left[\left(I_s \otimes J_n\right) + h\Delta J_n\right], \\ G &:= I_{sd} - h\left(D \otimes I_d\right)\left[\left(I_s \otimes J_n\right) + h\Delta J_n\right], \end{aligned} \tag{3.4}$$

where ΔJ_n is the block-diagonal matrix $h^{-1}\left[J(\mathbf{Y}) - \left(I_s \otimes J_n\right)\right]$ which is bounded as $h \to 0$ and vanishes if J_n does not depend on $\mathbf{y}_n$. Finally, P_j is substituted into the matrix Z_j as defined in (2.4) and the coefficient matrices M_j and N_j are determined such that $Z_j = O\left(h^2\right)$. An elementary derivation then leads to the expression (3.2) for P_j containing the free parameters σ_j and π_j.

Given the matrices P_j, the matrices Z_j and H_m can now be derived by substituting (3.2) and (3.4) into (2.4). For Z_j, we find

$$\begin{aligned} Z_j &= I_{sd} - G^{-1}P_jC = I_{sd} - \left(I_{sd} - h\left(D \otimes J_n\right) - h^2\left(D \otimes I_d\right)\Delta J_n\right)^{-1} \\ &\quad \times P_j\left(I_{sd} - h\left(A \otimes J_n\right) - h^2\left(A \otimes I_d\right)\Delta J_n\right) \\ &= I_{sd} - \left(I_{sd} - h\left(D \otimes J_n\right)\right)^{-1}\left(I_{sd} + h^2\left(D \otimes I_d\right)\Delta J_n\right) \\ &\quad \times P_j\left(I_{sd} - h\left(A \otimes J_n\right) - h^2\left(A \otimes I_d\right)\Delta J_n\right) + O\left(h^3\Delta J_n\right). \end{aligned}$$

Using (3.2′), we find

$$\begin{aligned}
Z_j &= I_{sd} - (I_{sd} - h\,(D\otimes J_n))^{-1} P_j\,(I_{sd} - h\,(A\otimes J_n)) - h^2(A-D)\otimes I_d\Delta J_n + O\left(h^3\Delta J_n\right)\\
&= (I_{sd} - h\,(D\otimes J_n))^{-2}\Big[(I_{sd} - h\,(D\otimes J_n))^2\\
&\quad - \left(I_{sd} - \pi_j h^2 D^2 W_j(h)\otimes I_d + hD^2\left(W_j(h) - I_s\right)A^{-1}\otimes J_n\right)(I_{sd} - h\,(A\otimes J_n))\Big]\\
&\quad - h^2(A-D)\otimes I_d\Delta J_n + O\left(h^3\Delta J_n\right)\\
&= (I_{sd} - h\,(D\otimes J_n))^{-2} D^2 W_j(h)\\
&\quad\times\left[\pi_j h^2 I_s\otimes I_d - h\left(I_s - W_j(h)^{-1}\left[I_s - 2D^{-1}A + D^{-2}A^2\right]\right)A^{-1}\otimes J_n + h^2 I_s\otimes J_n^2\right]\\
&\quad - h^2(A-D)\otimes I_d\Delta J_n + O\left(h^3\Delta J_n\right),
\end{aligned}$$

where $W_j(h)$ is defined in (3.2). Elimination of $W_j^{-1}(h)$ yields

$$\begin{aligned}
Z_j = h^2\,(I_{sd} - h\,(D\otimes J_n))^{-2} D^2 W_j(h)&\\
\otimes\left(\pi_j I_d - \sigma_j J_n + J_n^2\right) - h^2(A-D)\otimes I_d\Delta J_n + O\left(h^3\Delta J_n\right),& \qquad (3.3a)
\end{aligned}$$

resulting into the expression given in the theorem as $h\to 0$. On substitution of (3.3a) into (2.4), we obtain for H_m

$$\begin{aligned}
H_m &= h^{2m}\,(I_{sd} - h\,(D\otimes J_n))^{-2m}\\
&\quad\times\prod_{j=m-1}^{0}\left[D^2W_j(h)\otimes\left(\pi_j I_d - \sigma_j J_n + J_n^2\right) - (A-D)\otimes I_d\Delta J_n\right] + O\left(h^{2m+1}\right), \qquad (3.3b)
\end{aligned}$$

which again reduces to the expression given in the theorem as $h\to 0$. ∎

The method defined by (2.2) and (3.2) will be denoted by PDIRKJ$\{2m,\lambda_k\}$ (Parallel Diagonally Implicit Runge-Kutta method using the Jacobian matrix and $2m$ fitting points $\{\lambda_k\}$). From (3.2), it follows that the preconditioners P_j involve Jacobian evaluations and LU-decompositions of $I_{sd} - hD\otimes J_n$. However, these are already available because they are needed in the Newton iteration process, so that per iteration the sequential costs of applying the preconditioner P_j essentially consists of a forward-backward substitution of dimension d and a multiplication by the Jacobian J_n.

Upon substitution of (3.3) into (2.10) and by observing that the order q of the predictor can never exceed the number of interpolated values or the stage order r of the corrector, we find that the stage vector iteration error of the PDIRKJ $\{2m,\lambda_k\}$ method is of the form

$$\varepsilon^{(m)} = h^{2m+q+1}\left[\mathbf{C}_m\otimes S_{2m}\,(J_n)\,\mathbf{y}^{(q+1)}\,(t_n) + O\,(\Delta J_n) + O(h)\right], \qquad (3.5)$$

where the principal iteration error vector takes the form

$$\begin{aligned}
\mathbf{C}_m &= \left(A^2 - 2DA + D^2\right)^m \mathbf{v}_{q+1},\\
\mathbf{v}_{q+1} &:= \frac{1}{(q+1)!}\left(E(\mathbf{c}-\mathbf{e})^{q+1} + (q+1)D^*\mathbf{c}^q - \mathbf{c}^{q+1}\right), \qquad q\le\min\{r,s-1\}.
\end{aligned} \qquad (3.6)$$

For the LSV predictor $\mathbf{Y}^{(0)} = \mathbf{e}\otimes\mathbf{y}_n$, we have $q=0$, so that $\mathbf{v}_{q+1} = -\mathbf{c}$. In the case of the EXP and BDF predictors, we deduce from Theorem 2.1 that we can always achieve $q=\min\{r,s-1\}$ if E satisfies the relations

$$\begin{aligned}
E\mathbf{e} &= \mathbf{e};\\
E(\mathbf{c}-\mathbf{e})^j &= \mathbf{c}^j - jD^*\mathbf{c}^{j-1}, \qquad j=1,\dots,r;\\
E(\mathbf{c}-\mathbf{e})^j &= j\,(A-D^*)\,\mathbf{c}^{j-1}, \qquad j=r+1,\dots,s-1.
\end{aligned}$$

By introducing the vectors

$$\begin{aligned} \mathbf{k}_0 &:= \mathbf{e} \\ \mathbf{k}_j &:= \mathbf{c}^j - jD^*\mathbf{c}^{j-1}, \qquad && j = 1,\dots,r; \\ \mathbf{k}_j &:= j\,(A - D^*)\,\mathbf{c}^{j-1}, \qquad && j = r+1,\dots,s-1, \end{aligned} \tag{3.7}$$

and by defining the s-by-s matrices U and V such that their columns are, respectively, given by the vectors $\{\mathbf{k}_j\}$ and $\{(\mathbf{c}-\mathbf{e})^j\}$, $j = 0,\dots,s-1$, we may write $E = UV^{-1}$, provided that V is nonsingular. The vector $\mathbf{v}_{q+1}$ can now be obtained by formula (3.6). Notice that, in the particular case where the corrector is of collocation type, we have $r = s$.

From the preceding derivations, it follows that the order of PDIRKJ methods is given by $p^* = \min\{p, 2m\}$ for LSV predictors and by $p^* = \min\{p, 2m + \min\{r, s-1\}\}$ for EXP and BDF predictors. The truncation error constants are determined by the truncation error constant of the corrector and the iteration error vector $\mathbf{C}_m$ defined by (3.6) and (3.7).

It is tempting to exploit the free matrix D for the minimization of the magnitude of $\mathbf{C}_m$. However, $\mathbf{C}_m$ characterizes the magnitude of the *nonstiff* iteration error components, and since we are dealing with *stiff* IVPs, we should also consider the *stiff* iteration error components (error components corresponding to eigenvalues of the Jacobian matrix J_n of *large* magnitude).

3.2. Stiff Iteration Error Components

In this section, we investigate the damping of the *stiff* iteration error components. We shall do this for the test equation $\mathbf{y}' = \lambda\mathbf{y} + \mathbf{g}(t)$, where $\mathbf{g}(t)$ is a smooth function of t and λ is a *stiff* eigenvalue of J_n, that is, $z := h\lambda$ is of large magnitude. The following theorem is the stiff analogue of Theorem 3.1 for this test equation.

THEOREM 3.2. *Let S_{2m}, P_j and W_j be defined by (3.1) and (3.2), and define the matrices*

$$K_0 := I_s, \quad K_m(h) := \prod_{j=m-1}^{0} W_j(h), \qquad m \geq 1. \tag{3.8}$$

Then, for $z := h\lambda \to \infty$, the error amplification matrices Z_j and H_m are given by

$$Z_j(h) = W_j(h) \otimes I_d + O\left(z^{-1}\right), \quad H_m(h) = K_m(h) \otimes I_d + O\left(z^{-1}\right). \tag{3.9}$$

PROOF. It is convenient to apply the iteration method (2.2) directly to the test equation $\mathbf{y}' = \lambda\mathbf{y} + \mathbf{g}(t)$, rather than rewriting this equation in autonomous form. It is straightforwardly verified that we again obtain the recursion (2.3) with $J(\mathbf{Y}) = \lambda I_{sd}$. Hence, the matrix Z_j reduces to

$$\begin{aligned} Z_j(h) = &\ \left[I_{sd} - z^{-1}D^{-1} \otimes I_d\right]^{-1} \\ &\times \left[I_{sd} - \left(D^{-1} \otimes I_d\right) P_j \left(A \otimes I_d\right) - z^{-1}\left(D^{-1} \otimes I_d\right) + z^{-1}\left(D^{-1} \otimes I_d\right) P_j\right] \\ = &\ \left[I_{sd} + z^{-1}D^{-1} \otimes I_d\right] \\ &\times \left[I_{sd} - \left(D^{-1} \otimes I_d\right) P_j \left(A \otimes I_d\right) - z^{-1}\left(D^{-1} \otimes I_d\right) + z^{-1}\left(D^{-1} \otimes I_d\right) P_j\right] + O\left(z^{-2}\right) \\ = &\ I_{sd} - \left(D^{-1} \otimes I_d\right) P_j \left(A \otimes I_d\right) + O\left(z^{-1}\right). \end{aligned}$$

Substitution of

$$\begin{aligned} P_j = &\ -\left[I_{sd} - z^{-1}D^{-1} \otimes I_d\right]^{-1}\left[\left(D \otimes I_d\right)\left(W_j(h) - I_s\right)A^{-1} \otimes I_d + z^{-1}\left(D^{-1} \otimes I_d\right) + O\left(z^{-1}h^2\right)\right] \\ = &\ -\left[I_{sd} + z^{-1}D^{-1} \otimes I_d\right] \\ &\times \left[\left(D \otimes I_d\right)\left(W_j(h) - I_s\right)A^{-1} \otimes I_d + z^{-1}\left(D^{-1} \otimes I_d\right) + O\left(z^{-1}h^2\right)\right] + O\left(z^{-2}\right) \\ = &\ -\left(D \otimes I_d\right)\left(W_j(h) - I_s\right)A^{-1} \otimes I_d + O\left(z^{-1}\right) \end{aligned}$$

and using (2.4) yields (3.9). ∎

From this theorem, we conclude that for the *stiff* error components the matrix $H_m(h) = O(1)$ as $h \to 0$, whereas, for the *nonstiff* error components, the matrix $H_m(h) = O\left(h^{2m}\right)$ (see Theorem 3.1). Hence, it is to be expected that the convergence of the *stiff* error components will dominate the overall convergence of the iteration process. This leads us to base the determination of the matrix D on the magnitude of the matrix $K_m(h)$ as defined in (3.8).

4. DETERMINATION OF THE MATRIX D

In this section, the matrix D will be employed for improving the convergence of the *stiff* error components by controlling the magnitude of the matrix $K_m(h)$ defined in (3.8). We shall concentrate on the case $h = 0$ and we write $K_m = K_m(0) = W^m$ where $W = W(0) = I_s - 2D^{-1}A + D^{-2}A^2$ (cf. (3.2)). A similar situation is discussed in [1] for the PDIRK methods. We recall that these methods are obtained from (2.2) by dropping the preconditioner. For the PDIRK methods, the matrix W is given by $I - D^{-1}A$. In [1], the matrix D is chosen such that D minimizes the spectral radius of W. This minimal-spectral-radius iteration strategy is based on the assumption that the reduction factor ρ_m in the formula

$$\|W^m\|_\infty = [\rho_m]^m, \tag{4.1}$$

converges sufficiently fast to the spectral radius $\rho(W)$ of W. Clearly, if the reduction factor $\rho_m \approx \rho(W)$, then the best we can do seems to be the minimization of $\rho(W)$. However, this relation is only asymptotically guaranteed, that is, $\rho_\infty = \rho(W)$, provided $\rho(W) \leq 1$. Hence, it is not evident that the minimal-spectral-radius approach leads to matrices D such that ρ_m is also sufficiently small for small values of m. We investigate this for the PDIRKJ methods based on Radau IIA correctors of orders 3, 5 and 7. The first 5 significant digits of the entries of the matrices D minimizing ρ_m are given in Table 4.1, and Table 4.2 lists for $m = 4$ and $m = 5$ the matrices D minimizing ρ_m (the minimal-reduction-factor iteration strategy). Furthermore, Table 4.3 presents, for various values of m, the ρ_m-values for these three strategies. These results give rise to the following observations:

(i) in all strategies, the factors ρ_m strongly vary with m,
(ii) in all strategies, the first two iterations may lead to amplification of the stiff error components,
(iii) ignoring the first two iterations, the minimal ρ_4 and ρ_∞ strategies seem to be preferable.

Table 4.1. Matrices $D = \mathbf{diag}(d_1, \ldots, d_s)$ minimizing $\rho_\infty = \rho(W)$.

Corrector	s	$\rho_\infty = \rho(W)$	d_1	d_2	d_3	d_4
Radau IIA	2	0	0.97266	0.39661		
Radau IIA	3	0.013	0.49336	0.25710	0.39656	
Radau IIA	4	0.0041	0.46239	0.29118	0.15770	0.24121

Table 4.2. Matrices D minimizing ρ_m for the four-point Radau IIA corrector.

m	ρ_m	d_1	d_2	d_3	d_4
4	0.13	0.46151	0.29070	0.15757	0.24088
5	0.14	0.26698	0.15915	0.29987	0.35116

Table 4.3. Values of ρ_m for the four-point Radau IIA corrector.

Iteration strategy	$m=1$	$m=2$	$m=3$	$m=4$	$m=5$	$m=6$	$m=7$	$m=8$
Minimal ρ_∞	5.16	2.33	0.79	0.21	0.15	0.14	0.13	0.09
Minimal ρ_4	5.17	2.33	0.78	0.13	0.14	0.14	0.13	0.08
Minimal ρ_5	2.67	1.67	1.07	0.38	0.14	0.15	0.14	0.12

However, it should be remarked that the computed reduction factors are based on the norm of the matrix W^m, and therefore correspond to a "worst case" situation and not necessarily to the actual situation. For example, if the stiff components of the initial error $\varepsilon^{(0)}$ are in a particular subspace of the stiff eigenspace of the Jacobian J_n, then the actual ρ_m factors may be much smaller. In order to get some insight into the initial error $\varepsilon^{(0)}$, we again consider the test equation $\mathbf{y}' = \lambda \mathbf{y} + \mathbf{g}(t)$. Let us assume that the stage vector $\mathbf{X}$ occurring in the predictor formula (2.8) is sufficiently close to the corrector stage vector solution corresponding to the preceding step, that is, $\mathbf{y}_{n-1}, \mathbf{y}_n$ and $\mathbf{X}$ approximately satisfy (2.1). In such a model situation, we can derive an explicit expression for $\varepsilon^{(0)}$:

THEOREM 4.1. *Let $\mathbf{y}_{n-1}, \mathbf{y}_n$ and $\mathbf{X}$ approximately satisfy (2.1). Then, for the BDF predictors with $D^* = D$ and the explicit predictors with $D^* = O$, the stiff part of the initial iteration error can, respectively, be approximated by*

$$\begin{aligned} \varepsilon^{(0)} &= z^{-1}\mathbf{v} \otimes \mathbf{y}_n + O\left(z^{-2}\right) + O\left(hz^{-2} g\left(t_n\right)\right), \\ \mathbf{v} &= \left(I_s - \left[\mathbf{e}_s^{\mathsf{T}} A^{-1}\mathbf{e}\right]^{-1} D^{-1} E\right) A^{-1}\mathbf{e}, \end{aligned} \tag{4.2a}$$

$$\begin{aligned} \varepsilon^{(0)} &= \mathbf{v} \otimes \mathbf{y}_n + O\left(z^{-1}\right) + O\left(hz^{-1} g\left(t_n\right)\right), \\ \mathbf{v} &= \left[\mathbf{e}_s^{\mathsf{T}} A^{-1}\mathbf{e}\right]^{-1} E A^{-1}\mathbf{e}. \end{aligned} \tag{4.2b}$$

PROOF. It is easily verified that

$$\begin{aligned} \mathbf{Y}^{(0)} &= \left(\left(I_s - zD^*\right)^{-1} E \otimes I_d\right) \mathbf{X} + h\left(\left(I_s - zD^*\right)^{-1} D^* \otimes I_d\right) \mathbf{g}\left(t_n\mathbf{e} + h\mathbf{c}\right), \\ \mathbf{Y} &= \left(I_s - zA\right)^{-1} \mathbf{e} \otimes \mathbf{y}_n + h\left(\left(I_s - zA\right)^{-1} A \otimes I_d\right) \mathbf{g}\left(t_n\mathbf{e} + h\mathbf{c}\right). \end{aligned}$$

Since $\mathbf{y}_{n-1}, \mathbf{y}_n$ and $\mathbf{X}$ are assumed to approximately satisfy (2.1), we have for our test equation

$$\begin{aligned} \mathbf{X} &= \left(I_s - zA\right)^{-1} \mathbf{e} \otimes \mathbf{y}_{n-1} + h\left(\left(I_s - zA\right)^{-1} A \otimes I_d\right) \mathbf{g}\left(t_{n-1}\mathbf{e} + h\mathbf{c}\right), \\ \mathbf{y}_{n-1} &= R(z)^{-1}\mathbf{y}_n - hR(z)^{-1}\left(\mathbf{e}_s^{\mathsf{T}}\left(I_s - zA\right)^{-1} A \otimes I_d\right) \mathbf{g}\left(t_{n-1}\mathbf{e} + h\mathbf{c}\right), \\ R(z) &:= \mathbf{e}_s^{\mathsf{T}}\left(I_s - zA\right)^{-1}\mathbf{e}. \end{aligned}$$

Thus,

$$\begin{aligned} \mathbf{X} = &\left(I_s - zA\right)^{-1} \mathbf{e} \otimes \left[R(z)^{-1}\mathbf{y}_n - hR(z)^{-1}\left(\mathbf{e}_s^{\mathsf{T}}\left(I_s - zA\right)^{-1} A \otimes I_d\right) \mathbf{g}\left(t_{n-1}\mathbf{e} + h\mathbf{c}\right)\right] \\ &+ h\left(\left(I_s - zA\right)^{-1} A \otimes I_d\right) \mathbf{g}\left(t_{n-1}\mathbf{e} + h\mathbf{c}\right) \\ = &\; R(z)^{-1}\left(I_s - zA\right)^{-1} \mathbf{e} \otimes \mathbf{y}_n - h\left[\left(R(z)^{-1}\left(I_s - zA\right)^{-1}\mathbf{e}\mathbf{e}_s^{\mathsf{T}} - I_s\right)\left(I_s - zA\right)^{-1} A \otimes I_d\right] \\ &\times \mathbf{g}\left(t_{n-1}\mathbf{e} + h\mathbf{c}\right). \end{aligned}$$

The initial stage vector error takes the form

$$\begin{aligned} \varepsilon^{(0)} &= \mathbf{q}(z) \otimes \mathbf{y}_n - h\left(M(z) \otimes I_d\right) \mathbf{g}\left(t_{n-1}\mathbf{e} + h\mathbf{c}\right) + h\left(N(z) \otimes I_d\right) \mathbf{g}\left(t_n\mathbf{e} + h\mathbf{c}\right), \\ \mathbf{q}(z) &:= \left(R(z)^{-1}\left(I_s - zD^*\right)^{-1} E - I_s\right)\left(I_s - zA\right)^{-1}\mathbf{e}, \\ M(z) &:= \left(I_s - zD^*\right)^{-1} E\left[R(z)^{-1}\left(I_s - zA\right)^{-1}\mathbf{e}\mathbf{e}_s^{\mathsf{T}} - I_s\right]\left(I_s - zA\right)^{-1} A, \\ N(z) &:= \left(I_s - zD^*\right)^{-1} D^* - \left(I_s - zA\right)^{-1} A. \end{aligned}$$

For $|z| \to \infty$, the choice $D^* = D$ yields

$$\begin{aligned} \mathbf{q}(z) &= z^{-1}\left(I_s - \frac{1}{\mathbf{e}_s^{\mathrm{T}} A^{-1}\mathbf{e}} D^{-1}E\right) A^{-1}\mathbf{e} + O\left(z^{-2}\right), \\ M(z) &= z^{-2} D^{-1} E \left(\frac{1}{\mathbf{e}_s^{\mathrm{T}} A^{-1}\mathbf{e}} A^{-1}\mathbf{e}\mathbf{e}_s^{\mathrm{T}} - I_s\right) + O\left(z^{-3}\right), \\ N(z) &= z^{-2}\left(A^{-1} - D^{-1}\right) + O\left(z^{-3}\right). \end{aligned} \tag{4.3a}$$

For $D^* = O$, we find

$$\begin{aligned} \mathbf{q}(z) &= \frac{1}{\mathbf{e}_s^{\mathrm{T}} A^{-1}\mathbf{e}} E A^{-1}\mathbf{e} + O\left(z^{-1}\right), \\ M(z) &= -z^{-1} E \left(\frac{1}{\mathbf{e}_s^{\mathrm{T}} A^{-1}\mathbf{e}} A^{-1}\mathbf{e}\mathbf{e}_s^{\mathrm{T}} - I_s\right), \\ N(z) &= z^{-1} I_s + O\left(z^{-2}\right). \end{aligned} \tag{4.3b}$$

From (4.3), the assertion of the theorem readily follows. ∎

From (2.4) and the Theorems 3.2 and 4.1, we deduce that, in the model situation, the final iteration error reads

$$\begin{aligned} \varepsilon^{(m)} &= \left[K_m(h) \otimes I_d + O\left(z^{-1}\right)\right] \varepsilon^{(0)} \\ &= z^{-\sigma}\Gamma_m(h) \otimes \mathbf{y}_n + O\left(z^{-1-\sigma}\right) + O\left(h z^{-1-\sigma} g\left(t_n\right)\right), \\ \Gamma_m(h) &:= K_m(h)\mathbf{v}, \end{aligned} \tag{4.4}$$

where $\sigma = 0$ if $D^* = O$, and $\sigma = 1$ if $D^* = D$. The vector $\Gamma_m(h)$ will be called the *stiff* iteration error vector. We define the *actual* reduction factor γ_m by

$$\gamma_m := \sqrt[m]{\|\Gamma_m\|_\infty \|\Gamma_0\|_\infty^{-1}}, \qquad \Gamma_m := \Gamma_m(0). \tag{4.5}$$

Table 4.4 presents the analogue of Table 4.3 for the quantities γ_m. Table 4.4 indicates that on the basis of the actual reduction factors, the three iteration strategies will show a much more equal behaviour than Table 4.3 suggests. However, also note that the minimal ρ_5 strategy has an initial vector Γ_0 of much smaller magnitude.

Table 4.4. Values of γ_m for the four-point Radau IIA corrector.

Iteration strategy	$\|\Gamma_0\|_\infty$	$m=1$	$m=2$	$m=3$	$m=4$	$m=5$	$m=6$	$m=7$	$m=8$
Minimal ρ_∞	213.8	1.31	0.99	**0.30**	0.10	0.11	0.11	0.08	**0.05**
Minimal ρ_4	214.1	1.31	0.99	0.31	**0.09**	0.11	0.11	0.09	0.06
Minimal ρ_5	**67.4**	**0.68**	**0.84**	0.31	0.13	**0.09**	**0.10**	0.09	0.07

Table 4.5. Values of c_m for the four-point Radau IIA corrector.

Iteration strategy	$\|\mathbf{C}_0\|_\infty$	$m=1$	$m=2$	$m=3$	$m=4$	$m=5$	$m=6$	$m=7$	$m=8$
Minimal ρ_∞	0.043	0.037	0.25	0.22	0.27	0.28	0.29	0.30	0.31
Minimal ρ_4	0.044	**0.036**	0.25	0.22	0.27	0.28	0.29	0.30	0.31
Minimal ρ_5	**0.007**	0.101	**0.17**	**0.18**	**0.19**	**0.19**	**0.19**	**0.20**	**0.20**

Finally, we compare the actual nonstiff reduction factors based on the nonstiff iteration error vector $\mathbf{C}_m$ and defined by

$$c_m := \sqrt[m]{\|\mathbf{C}_m\|_\infty \|\mathbf{C}_0\|_\infty^{-1}}. \tag{4.6}$$

This leads to the values listed in Table 4.5. Evidently, it is now the minimal ρ_5 approach that is clearly superior to the minimal ρ_∞ and minimal ρ_4 strategies.

Summarizing, we conclude that the three iteration strategies are expected to perform similarly in cases where the stiff components in the iteration error dominate the rate of convergence, and that the minimal ρ_5 strategy should become superior if the nonstiff components dominate the rate of convergence.

5. NUMERICAL EXPERIMENTS

In this section, we compare the PDIRKJ$\{2m, \lambda_k\}$ methods, using various iteration strategies, with the PDIRK methods developed in [1] which are obtained from (2.2) by setting $P_j = I_{sd}$. The PDIRK methods are applied with the iteration strategy used in [1], that is, the initial iterate is provided by the LSV predictor, the outer iteration strategy is based on the minimal ρ_∞ approach, and the inner iteration uses modified Newton, iterated to convergence. The PDIRKJ$\{2m, \lambda_k\}$ methods are applied with the BDF predictor (unless stated otherwise), the same inner iteration strategy as in PDIRK, and with an outer iteration strategy based on either the minimal ρ_∞ approach or the minimal ρ_5 approach. Both methods use Jacobian matrices at step points that are updated in each step.

The accuracy of the numerical solution is given by the number of correct digits Δ, obtained by writing the maximum norm of the absolute error or relative error at the endpoint in the form $10^{-\Delta_{abs}}$ or $10^{-\Delta_{rel}}$, respectively. The sequential computational effort is estimated by the total number of nonlinear systems that have to be solved per processor (it is assumed that at least s processors are available). This number is given by NM, where N is the total number of steps, and $M = m$ when using the LSV predictor and $M = m + 1$ when using the BDF predictor.

5.1. Convergence of Stiff and Nonstiff Iteration Error Components

We start with a comparison of the convergence of the stiff and nonstiff iteration error components for the PDIRKJ$\{2m, \lambda_k\}$ methods with zero fitting points ($\lambda_k = 0$). As a first test problem, we choose the problem of Kaps [10]:

$$\begin{gathered} \frac{dy_1}{dt} = -\left(2+\varepsilon^{-1}\right) y_1 + \varepsilon^{-1} (y_2)^2, \quad \frac{dy_2}{dt} = y_1 - y_2 (1 + y_2), \\ y_1(0) = y_2(0) = 1, \quad \varepsilon = 10^{-3}, \qquad 0 \le t \le 1, \end{gathered} \tag{5.1}$$

with the exact solution $y_1 = \exp(-2t)$ and $y_2 = \exp(-t)$ for all values of the parameter ε. This system consists of a stiff and nonstiff equation. The first and second vector component of the numerical solution may be considered as the stiff and nonstiff solution components. Both methods are applied with the four-point Radau IIA corrector.

The Tables 5.1a and 5.1b present accuracies for the nonstiff and stiff component in the Kaps problem (5.1). These results clearly show that the accuracy of both the PDIRK, PDIRKJ and of the corrector solution is dominated by the accuracy of the *stiff* solution component. Furthermore, we see that for both the BDF and LSV predictor the PDIRKJ method is more accurate than PDIRK, particularly for low numbers of iterations. This behaviour was confirmed for almost all other test problems we tried, so that we shall omit further comparisons with the PDIRK method.

Table 5.1a. Values of Δ_{abs} for the nonstiff component in the Kaps problem (5.1).

Method	h	$M=2$	$M=3$	$M=4$	$M=5$	$M=6$	4-stage Radau IIA
PDIRKJ $\{2m, 0\}$	1/2	3.6	5.4	7.7	8.6	8.8	8.8
PDIRKJ $\{2m, 0\}^*$	1/2	4.9	6.9	8.3	8.6	8.8	
PDIRK	1/2	2.5	4.2	5.7	6.4	7.4	
PDIRKJ $\{2m, 0\}$	1/4	4.1	6.8	9.4	10.4	11.7	11.8
PDIRKJ $\{2m, 0\}^*$	1/4	4.5	7.8	9.3	9.5	10.8	
PDIRK	1/4	3.4	4.2	7.1	7.9	9.2	

*BDF predictor replaced by LSV predictor

Table 5.1b. Values of Δ_{abs} for the stiff component in the Kaps problem (5.1).

Method	h	$M=2$	$M=3$	$M=4$	$M=5$	$M=6$	4-stage Radau IIA
PDIRKJ $\{2m,0\}$	1/2	3.7	3.8	5.4	7.8	6.4	6.4
PDIRKJ $\{2m,0\}^*$	1/2	2.4	4.0	5.6	6.4	6.4	
PDIRK	1/2	1.9	2.3	5.5	5.7	6.8	
PDIRKJ $\{2m,0\}$	1/4	4.4	4.9	6.5	7.8	7.8	7.8
PDIRKJ $\{2m,0\}^*$	1/4	1.6	4.8	6.3	6.5	7.5	
PDIRK	1/4	0.6	1.1	5.6	6.2	7.4	

* BDF predictor replaced by LSV predictor

5.2. Comparison of Outer Iteration Strategies

Next we compare the minimal ρ_∞ and the minimal ρ_5 outer iteration strategy for the PDIRKJ $\{2m,0\}$ method with the four-point Radau IIA corrector. The first test problem is the nonlinear Prothero and Robinson problem (cf. [1]):

$$\begin{gathered}\frac{dy(t)}{dt} = -\varepsilon^{-1}\left(y^3(t) - g(t)^3\right) + g'(t),\\ y(t_0) = g(t_0), \quad g(t) := \cos(t), \quad \varepsilon := 10^{-3}, \qquad 0 \le t \le 1,\end{gathered} \tag{5.2}$$

with exact solution $y(t) = g(t)$ for all values of the parameter ε. The results of Table 5.2 indicate a better performance of the minimal ρ_5 iteration strategy.

Table 5.2. Values of Δ_{abs} for the Prothero and Robinson problem (5.2).

Iteration strategy	h	$M=2$	$M=3$	$M=4$	$M=5$	$M=6$	4-stage Radau IIA
Minimal ρ_∞	1/2	2.3	3.4	3.7	6.0	7.0	7.3
Minimal ρ_5		2.9	4.4	4.6	6.7	7.2	
Minimal ρ_∞	1/4	5.4	5.2	7.8	8.1	8.4	8.5
Minimal ρ_5		7.2	6.5	7.8	8.4	8.4	

The test set of Enright *et al.* [11] contains the following system of ODEs describing a chemical reaction:

$$\begin{gathered}\frac{d\mathbf{y}}{dt} = -\begin{pmatrix} .013 + 1000y_3 & 0 & 0 \\ 0 & 2500y_3 & 0 \\ .013 & 0 & 1000y_1 + 2500y_2 \end{pmatrix}\mathbf{y}, \quad \mathbf{y}(0) = (1,1,0)^{\mathrm{T}},\\ 0 \le t \le T := 51.\end{gathered} \tag{5.3a}$$

Since we use fixed step sizes in our experiments, we avoided the initial phase by choosing the starting point at $t_0 = 1$. The corresponding initial and end point values are given by

$$\mathbf{y}(1) \approx \begin{pmatrix} 0.990731920827 \\ 1.009264413846 \\ -.366532612659\ 10^{-5} \end{pmatrix}, \quad \mathbf{y}(T) = \begin{pmatrix} 0.591045966680 \\ 1.408952165382 \\ -.186793736719\ 10^{-5} \end{pmatrix} \tag{5.3b}$$

Table 5.3 shows a more or less comparable performance of the two iteration strategies.

Table 5.3. Values of Δ_{abs} for the chemical reaction problem (5.3).

Iteration strategy	h	$M=2$	$M=3$	$M=4$	$M=5$	$M=6$	4-stage Radau IIA
Minimal ρ_∞	T/2	3.1	5.7	7.5	8.9	10.2	9.8
Minimal ρ_5		3.8	5.4	7.2	8.8	9.6	
Minimal ρ_∞	T/4	4.1	7.2	9.1	9.6	10.6	11.8
Minimal ρ_5		5.1	7.0	9.2	10.4	11.4	

Finally, we consider the circuit analysis problem of Horneber [12] consisting of 15 highly nonlinear, stiff equations describing a ring modulator. For specifications of this problem, we refer to [13]. We solved this problem on the interval $0 \le t \le 10^{-3}$. Table 5.4 presents results obtained by PDIRKJ$\{2m, 0\}$ using the minimal ρ_∞ and minimal ρ_5 iteration strategies, and by PDIRK using the minimal ρ_∞ strategy. In this difficult problem, the inner/outer iteration process did not always converge (indicated by *). Evidently, the minimal ρ_5 iteration strategy is less robust than the minimal ρ_∞ strategy.

Table 5.4. Values of Δ_{abs} for the Horneber problem.

Method	Iteration strategy	h	$M=2$	$M=3$	$M=4$	$M=5$	4-stage Radau IIA
PDIRKJ	Minimal ρ_∞	$4_{10}-7$	*	4.4	4.4	4.4	4.5
	Minimal ρ_5		*	*	*	*	
PDIRK	Minimal ρ_∞		*	*	4.4	4.9	
PDIRKJ	Minimal ρ_∞	$2_{10}-7$	4.8	6.6	7.4	8.4	8.4
	Minimal ρ_5		6.0	6.7	7.5	8.4	
PDIRK	Minimal ρ_∞		*	*	5.5	6.1	

Our conclusion from the experiments of this subsection is that the minimal ρ_5 iteration strategy is often more accurate than the minimal ρ_∞ strategy, but the greater robustness of the minimal ρ_∞ strategy leads us to adopt this strategy as the most recommendable one.

5.3. Comparison of Correctors of Different Orders

In an actual implementation where the desired accuracy is controlled by a user-specified tolerance parameter, it is desirable that the method performs well in a range of stepsizes. A four-point Radau IIA corrector is expected to be suitable for producing high accuracy results because of its relatively high stiff order $s = 4$ and nonstiff order $p = 7$, but how does it perform for larger stepsizes when compared with lower order correctors. Table 5.5 compares s-point Radau IIA correctors for $s = 2, 3$ and 4. In all cases, we used the minimal ρ_∞ strategy for which the matrices D are listed in Table 4.1. Evidently, the PDIRKJ$\{2m, 0\}$ using the four-point Radau IIA corrector is more robust and considerably more accurate than when using lower-order correctors.

Table 5.5. Values of Δ_{abs} for the Horneber problem.

s	h	$M=2$	$M=3$	$M=4$	$M=5$
2	$4_{10}-7$	*	*	*	*
3		*	*	*	*
4		*	4.4	4.4	4.4
2	$2_{10}-7$	2.4	2.9	2.9	2.9
3		4.3	5.9	6.0	6.0
4		4.8	6.6	7.4	8.4
2	$1_{10}-7$	3.3	3.8	3.8	3.8
3		5.5	7.3	7.4	
4		6.2	8.5	8.5	

5.4. Spectral Fitting

Finally, we demonstrate that the parameters occurring in the preconditioners can be used for improving the accuracy of specific solution components. This facility may be useful in problems where we not only have stiff and nonstiff components, but also "stiff/nonstiff" components. For example, the IVP

$$\frac{dy}{dt} = -\begin{pmatrix} 10 & -\alpha & 0 & 0 & 0 & 0 \\ \alpha & 10 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{10} \end{pmatrix} \mathbf{y} + \mathbf{e}\sin(t), \tag{5.4}$$
$$\alpha = 10^4, \quad \mathbf{y}(0) = \mathbf{e}, \qquad 0 \le t \le 20, \quad \mathbf{e} := (1,1,1,1,1,1)^{\mathrm{T}},$$

has two extremely stiff components y_1 and y_2, one stiff/nonstiff component y_3, and three nonstiff components y_4, y_5 and y_6 (this problem differs from Problem B2 in [11] by the additional inhomogeneous term $\mathbf{e}\sin(t)$ which makes the solution less trivial). The PDIRK method with all fitting points at the origin has a strong damping effect on the stiff and nonstiff error components, but does not pay much attention to the stiff/nonstiff error components. Table 5.6 lists minimal accuracies for the three types of solution components obtained by PDIRKJ$\{2m, \lambda_k\}$ using three fitting strategies:

A all fitting points λ_k are at the origin,

B the fitting points coincide with the zeros of the Chebyshev polynomial shifted to the interval $[a, b] = [-4, 0]$,

C two fitting points at the origin and the remaining fitting points as in strategy B.

The results in Table 5.6 clearly show that strategy A "neglects" the stiff/nonstiff component y_3. Stategy B improves the accuracy of this middle component considerably, but at the cost of the nonstiff components. Strategy C seems to be an effective compromise; already after three iterations, the stiff/nonstiff as well as the nonstiff components have reached the corrector solution.

Table 5.6. Values of Δ_{rel} for problem (5.4) obtained by PDIRKJ$\{2m, \lambda_k\}$ with fitting strategies A, B and C.

Component	h	Strategy	$m = 1$	$m = 2$	$m = 3$	$m = 4$	4-stage Radau IIA
stiff	1	A	5.6	5.7	8.0	7.1	7.1
stiff/nonstiff			1.6	3.5	4.2	4.8	5.0
nonstiff			1.9	4.5	5.5	5.6	5.6
stiff	1	B	5.7	5.9	6.4	6.6	7.1
stiff/nonstiff			3.5	4.9	5.0	5.0	5.0
nonstiff			1.4	3.0	5.1	5.5	5.6
stiff	1	C	5.6	5.8	6.4	6.6	7.1
stiff/nonstiff			1.6	4.6	5.0	5.0	5.0
nonstiff			1.9	4.8	5.6	5.6	5.6

REFERENCES

1. P.J. van der Houwen and B.P. Sommeijer, Iterated Runge-Kutta methods on parallel computers, *SIAM J. Sci. Stat. Comput.* **12**, 1000–1028 (1991).
2. P.J. van der Houwen, B.P. Sommeijer, and W. Couzy, Embedded diagonally implicit Runge-Kutta algorithms on parallel computers, *Math. Comp.* **58**, 135–159 (1992).
3. A. Bellen, Parallelism across the steps for difference and differential equations, *Lecture Notes in Mathematics 1386*, pp. 22–35, Springer-Verlag, (1987).
4. A. Bellen, R. Vermiglio, and M. Zennaro, Parallel ODE solvers with step-size-control, *J. Comp. Appl. Math.* **31**, 277–293 (1990).
5. K.R. Jackson, and S.P. Nørsett, The potential for parallelism in Runge-Kutta methods, Part I: RK formulas in standard form, Technical Report No. 239/90, Department of Computer Science, University of Toronto, (1990).

6. K.R. Jackson, A. Kværnø, and S.P. Nørsett, Order of Runge-Kutta methods when using Newton-type iteration, Technical Report No. 1/91, Devision of Math. Sciences, University of Trondheim, (1992).
7. K. Burrage, The search for the Holy Grail, or Predictor-Corrector methods for solving ODEIVPs, *Appl. Numer. Math.* **11**, 125–141 (1993).
8. P.J. van der Houwen, Preconditioning in implicit initial value problem methods on parallel computers, *Advances in Computational Mathematics* (1993) (to appear).
9. P.J. van der Houwen, and B.P. Sommeijer, Butcher-Kuntzmann methods for nonstiff problems on parallel computers, Report NM-R9305, CWI, Amsterdam, *APNUM 15* (to appear).
10. P. Kaps, Rosenbrock-type methods, In *Numerical Methods for Stiff Initial Value Problems*, Bericht Nr. 9, (Edited by G. Dahlquist and R. Jeltsch), Inst. für Geometrie und Praktische Mathematik der RWTH, Aachen, (1981).
11. W.H. Enright, T.E. Hull, and B. Lindberg, Comparing numerical methods for stiff systems of ODEs, *BIT* **15**, 10–48 (1975).
12. E.H. Horneber, Analysis of nonlinear RLCÜ circuits by means of mixed potential functions with a systematic representation of nonlinear dynamic circuit analysis (in German), Thesis defended at the University of Kaiserslautern, Department of Electronics, (1976).
13. B.P. Sommeijer, Parallelism in the numerical integration of initial value problems, Thesis defended at the University of Amsterdam, (1992).